[0317] The procedure of MakePAT, MergePAT and TransPAT is similar to that described above with reference to Fig. 21, except that the AID should be replaced by the link information of the AID and the AID list should be replaced by the link specifying AID list. Also, the procedure of SplitPAT is similar to that described above with reference to Fig. 22, except that the AID should be replaced by the link information of the AID and the AID list should be replaced by the link specifying AID list.

[0318] Here, in the procedures of Fig. 21 and Fig. 22, the link specifying AID list generation is carried out according to Fig. 49 as follows. Namely, a buffer length is determined first (step S9011) and a buffer is generated (step S9012). Then, the link information of the holder AID is copied to a vacant region of the generated buffer (step S9017). Then, the link information of the member AID is copied to a vacant region of the resulting buffer (step S9018), and if the next member AID exists (step S9015 YES), the step S9018 is repeated.

[0319] Next, the determination of the link information of the holder AID will be described. Each of the MakePAT, the MergePAT, the SplitPAT, and the TransPAT commands is defined to have two or more arguments, where AID, PAT, or Enabler can be specified as an argument. In this case, the PAT processing device specifies the link information of the holder AID of the PAT to be outputted after executing each command according to the following rules.

* Case of the MakePAT:

  For the MakePAT command, it is defined that AIDs are to be specified for the first argument to the N-th argument(N = 2, 3, ·····) and Enablers are to be specified for the N+1-th and subsequent arguments. For example, they can be specified as follows.

  MakePAT AID_1, AID_2, ·····, AID_N, Enabler of AID_1, Enabler of AID_2, ·····, Enabler of AID_N

  The PAT processing device interprets the link information of AID of the first argument of the MakePAT command as the link information the holder AID.

  Only when one of the Enablers of the N+1-th and subsequent arguments corresponds to the AID of the first argument, the PAT processing device specifies the link information of this AID (that is the link information of the AID of the first argument) as the link information of the holder AID of the PAT to be outputted after executing the MakePAT command.

* Case of the MergePAT:

  For the MergePAT command, it is defined that PATs are to be specified for the first argument to the N-th argument (N = 2, 3, ·····) and Enabler is to be specified for the N+1-th argument. Namely, they can be specified as follows.

  MergePAT PAT_1 PAT_2 ····· PAT_N Enabler of AID

  The PAT processing device interprets the link information of the holder AID of the PAT of the first argument of the MergePAT command as the link information of the holder AID of the PAT to be outputted after executing the MergePAT command.

  Only when the Enabler of the N+1-th argument corresponds to the holder AID of the PAT of the first argument, the PAT processing device specifies the link information of this AID (that is the link information of the holder AID of the PAT of the first argument) as the link information of the holder AID of the PAT to be outputted after executing the MergePAT command.

* Case of the SplitPAT:

  For the SplitPAT command, it is defined that PAT is to be specified for the first argument, a set of one or more AIDs grouped together by some prescribed symbols (assumed to be parentheses () in this example) are to be specified for the second argument to the N-th argument (N = 3, 4, ·····), and Enabler is to be specified for the N+1-th argument. Namely, they can be specified as follows.

  SplitPAT PAT_1 (AID_{11}) (AID_{21} AID_{22}) ····· (AID_{N1} AID_{N2} ····· AID_{NM}) Enabler of AID

  The PAT processing device interprets the link information of the holder AID of the PAT of the first argument of the SplitPAT command as the link information of the holder AID of the PAT to be outputted after executing the SplitPAT command.

  Only when the Enabler of the N+1-th argument corresponds to the holder AID of the PAT of the first argument, the PAT processing device specifies the link information of this AID (that is the link information of the holder AID of the PAT of the first argument) as the link information of the holder AID of the PAT to be outputted after executing the SplitPAT command.

* Case of the TransPAT:

  For the TransPAT command, it is defined that PATs are to be specified for the first argument and the second argument, an AID is to be specified for the third argument, and Enablers are to be specified for the fourth argument and the fifth argument. Namely, they can be specified as follows.

  TransPAT PAT_1 PAT_2 AID Enabler of AID_1 Enabler of AID_2

  The PAT processing device interprets the link

information of AID of the third argument as the link information of the holder AID of the PAT to be outputted after executing the TransPAT command provided that the link information of AID of the third argument of the TransPAT command is contained in the PAT of the second argument.

Only when the Enabler of the fourth argument corresponds to both the PAT of the first argument and the PAT of the second argument and the Enabler of the fifth argument corresponds to the AID of the third argument, the PAT processing device specifies the link information of the AID of the third argument as the link information of the holder AID of the PAT to be outputted after executing the TransPAT command.

Next, the determination of the link informations of the member AIDs will be described. The definitions of the MakePAT, the MergePAT, the SplitPAT, and the TransPAT commands are as described above. The PAT processing device specifies the link informations of the member AIDs of the PAT to be outputted after executing each command according to the following rules.

* Case of the MakePAT:

Only when the link information of the holder AID of the PAT to be outputted after executing the MakePAT command is formally determined, the PAT processing device interprets all the link informations of the AIDs of the second and subsequent arguments of the MakePAT command as the link informations of the member AIDs of the PAT to be outputted after executing the MakePAT command.

The PAT processing device specifies only the link informations of those AIDs among all the AIDs of the second and subsequent arguments which correspond to the Enablers specified by the N+1-th and subsequent arguments as the link informations of the member AIDs of the PAT to be outputted after executing the MakePAT command.

* Case of the MergePAT:

Only when the link information of the holder AID of the PAT to be outputted after executing the MergePAT command is formally determined, the PAT processing device specifies the link informations of the member AIDs of all the PATs specified by the first to N-th arguments of the MergePAT as the link informations of the member AIDs of the PAT to be outputted after executing the MergePAT command.

* Case of the SplitPAT:

Only when the link information of the holder AID of the PAT to be outputted after executing the SplitPAT command is formally determined, the PAT processing device specifies the link information of the member AID of the PAT specified by the first argument of the SplitPAT command as the link information of the member AID of the PAT to be outputted after executing the SplitPAT command. At this point, the link informations of the member AIDs are distributed into different PATs in units of parentheses (). For example, in the case of:

SplitPAT PAT (AID_{11}) $(AID_{21}$ $AID_{22})$ ・・・・・・・・ $(AID_{N1}$ AID_{N2} ・・・・・・・・ $AID_{NM})$ Enabler of AID

the link informations of (AID_{11}), $(AID_{21}$ $AID_{22})$ and $(AID_{N1}$ AID_{N2} ・・・・・・・・ $AID_{NM})$ will be the link informations of the member AIDs of different PATs having a common link information of holder AID.

* Case of TransPAT:

Only when the link information of the holder AID of the PAT to be outputted after executing the TransPAT command is formally determined, the PAT processing device specifies all the link informations of the member AIDs remaining after excluding the link information of the member AID that is scheduled to be a new holder AID from all the link informations of the member AIDs of the PAT specified by the first argument of the TransPAT command and the link informations of the member AIDs of the PAT specified by the second argument as the link informations of the member AIDs of the PAT to be outputted after executing the TransPAT command.

The verification of the properness of the Enabler in this seventh embodiment is the same as described above with reference to Fig. 24. Also, this verification of the properness of the Enabler is common to the MakePAT, the MergePAT, the SplitPAT and the TransPAT.

[0320] Next, the eighth embodiment of the email access control scheme according to the present invention will be described in detail.

[0321] In this eighth embodiment, the OID is given by a real email address.

[0322] The PAT is an information comprising two or more real email addresses, the holder index, the validity period, the transfer control flag and the PAT processing device identifier (or the identifier of the PAT processing object on the network), which is signed using a secret key of the PAT processing device (or the PAT processing object on the network).

[0323] Here, one of the real email addresses is a holder email address of this PAT, where the change of the information contained in the PAT such as an addition of email address to the PAT, a deletion of email address from the PAT, a change of the validity period in the PAT, a change of the transfer control flag value in the PAT, etc., can be made by presenting the holder email address and an Enabler containing the holder email address to the PAT processing device (or the PAT processing object on the network).

[0324] On the other hand, the email addresses other than the holder email address that are contained in the PAT are all member email addresses, where a change

of the information contained in the PAT cannot be made even when the member email address and an Enabler containing the member email address are presented to the PAT processing device (or the PAT processing object on the network).

[0325] The holder index is a numerical data for identifying the holder email address, which is defined to take a value 1 when the holder email address is a top email address in the email address list formed from the holder email address and the member email addresses, a value 2 when the holder email address is a second email address from the top of the email address list, or a value n when the holder email address is an n-th email address from the top of the email address list.

[0326] The transfer control flag value is defined to take either 0 or 1.

[0327] The holder email address is defined to be a real email address which is written at a position specified by the holder index in the email address list. The member email addresses are defined to be all the email addresses other than the holder email address.

[0328] The validity period is defined by any one or combination of the number of times for which the PAT is available, the absolute time (UTC) by which the PAT becomes unavailable, the absolute time (UTC) by which the PAT becomes available, and the relative time (lifetime) since the PAT becomes available until it becomes unavailable.

[0329] The identifier of the PAT processing device (or the PAT processing object on the network) is defined as a serial number of the PAT processing device (or an distinguished name of the PAT processing object on the network). The secret key of the PAT processing device (or the PAT processing object on the network) is defined to be uniquely corresponding to the identifier.

[0330] Also, in this eighth embodiment, an Enabler is defined as an identifier corresponding to the real email address. The Enabler is an information comprising a character string uniquely indicating that it is an Enabler and a real email address itself, which is signed using the secret key of the PAT processing device or the PAT processing object on the network.

[0331] The generation of the PAT in this eighth embodiment is carried out as follows.

[0332] Here, a directory will be described as an example of the PAT processing object on the network. The directory manages the real email address and the disclosed information of the user in correspondence, and outputs the PAT upon receiving the search conditions presented from an arbitrary user.

[0333] The user transmits the real email address and the search conditions to the directory. Then, the directory acquires all the real email addresses which uniquely correspond to the disclosed information that satisfies these search conditions. Then, the directory generates a real email address list from the real email address of the user who presented the search conditions and all the real email addresses acquired as a search result. Then, the directory appends the holder index value, the validity period value, the transfer control flag value, and the distinguished name of the directory to the real email address list. Finally, the directory signs the resulting data using a secret key of the directory, and transmits it as the PAT to the user who presented the search conditions.

[0334] Next, the email access control in this eighth embodiment is carried out as follows.

[0335] The sender specifies the real email address of the sender in From: line, and "[PAT]@[real domain of sender]" in To: line of a mail.

[0336] The SCS acquires an email received by an MTA (Message Transfer Agent) such as SMTP (Simple Mail Transfer Protocol), and carries out the authentication by the following procedure.

(1) The signature of the PAT is verified using the public key of the PAT.

When the PAT is found to have been altered, the email is discarded and the processing is terminated.

When the PAT is found to have been not altered, the following processing (2) is executed.

(2) The search is carried out by presenting the sender's real email address to the PAT.

When a real email address that completely matches with the sender's real email address is not contained in the PAT, the email is discarded and the processing is terminated.

When a real email address that completely matches with the sender's real email address is contained in the PAT, the following processing (3) is executed.

(3) The validity period value of the PAT is evaluated.

When the PAT is outside the validity period, the email is discarded and the processing is terminated.

When the PAT is within the validity period, the following processing (4) is executed.

(4) Whether or not to authenticate the sender is determined by referring to the transfer control flag value of the PAT.

When the value is 1, the challenge/response authentication between the SCS and the sender is carried out, and the signature of the sender is verified. When the signature is valid, the recipient is specified and the PAT is attached. When the signature is invalid, the email is discarded and the processing is terminated.

When the value is 0, the recipient is specified and the PAT is attached without executing the challenge/response authentication.

[0337] An exemplary challenge/response authentication between the SCS and the sender in this eight embodiment can be carried out as follows.

[0338] First, the SCS generates an arbitrary informa-

tion such as a timestamp, for example, and transmits the generated information to the sender.

[0339] Then, the sender generates the secret key and the public key, signs the received information using the secret key, and transmits it along with the public key.

[0340] The SCS then verifies the signature of the received information using the public key presented from the sender. When the signature is valid, the recipient is specified and the PAT is attached. When the signature is invalid, the email is discarded and the processing is terminated.

[0341] The specifying of the recipient and the attaching of the PAT at the SCS in this eighth embodiment can be carried out as follows.

[0342] First, the SCS carries out the search by presenting the sender's real email address to the PAT, so as to acquire all the real email addresses which do not completely match the sender's real email address. Then, all these acquired real email addresses are specified as recipient's real email addresses.

[0343] Next, the SCS attaches the PAT to an arbitrary position in the email in order to transmit the PAT to all the recipient's email addresses so as to be able to realize the bidirectional communications. Finally, the SCS gives the email to the MTA.

[0344] The receiving refusal with respect to the PAT at the SCS in this eighth embodiment can be carried out as follows.

[0345] Receiving refusal setting: The bidirectional authentication is carried out by an arbitrary means between the user and the SCS 5. Then, the user transmits a registration command, his/her own real email address, and arbitrary PATs to the SCS 5. Then, the SCS 5 next verifies the signature of each received PAT using a public key of the ADS. Those PATs with the invalid signature are discarded by the SCS 5. When the signature is valid, the SCS 5 carries out the search by presenting the received real email address to each PAT. For each of those PATs which contain the real email address that completely matches with the received real email address, the SCS 5 presents the registration command and the PAT to the storage device such that the PAT is registered into the storage device. Those PATs which do not contain the real email address that completely matches with the received real email address are discarded by the SCS 5 without storing them into the storage device.

[0346] Receiving refusal execution: The SCS 5 carries out the search by presenting the PAT to the storage device. When a PAT that completely matches the presented PAT is registered in the storage device, the mail is discarded. When a PAT that completely matches the present PAT is not registered in the storage device, the mail is not discarded.

[0347] Receiving refusal cancellation: The bidirectional authentication is carried out by an arbitrary means between the user and the SCS 5. Then, the user presents his/her own real email address to the SCS 5. Then, the SCS 5 next presents the presented real email address as a search condition to the storage device and acquire all the PATs that contain the presented real email address, and then presents all the acquired PATs to the user. Then, the user selects all the PATs for which the receiving refusal is to be cancelled by referring to all the PATs presented from the SCS 5, and transmits all the selected PATs along with a deletion command to the SCS 5. Upon receiving the deletion command and all the PATs for which the receiving refusal is to be cancelled, the SCS 5 presents the deletion command and all the PATs received from the user to the storage device, such that all the received PATs are deleted from the storage device.

[0348] The editing of the PAT in this eighth embodiment can be carried out as follows.

[0349] The MakePAT, the MergePAT, the SplitPAT, and the TransPAT processings for the PAT using real email addresses as its elements can be obtained from the the MakePAT, the MergePAT, the SplitPAT, and the TransPAT processings for the PAT using AIDs as its elements described above, by replacing the AID by the real email address and the Enabler of AID by the Enabler of real email address.

[0350] A Null operator is an information comprising a data which is uniquely indicating that it is Null and which has a format of the real email address, which is signed by the secret key of the PAT processing device or the PAT processing object on the network.

[0351] Similarly, the God operator is an information comprising a data which is uniquely indicating that it is God and which has a format of the real email address, which is signed by the secret key of the PAT processing device or the PAT processing object on the network.

[0352] The Enabler of Null operator is an information comprising a data which is uniquely indicating that it is Enabler and the Null operator itself, which is signed by the secret key of the PAT processing device or the PAT processing object on the network.

[0353] The processings involving the Null operator and the God operator can be obtained from the processings for the PAT using AIDs as its elements described above, by replacing the AID by the real email address, the Enabler of AID by the Enabler of real email address, the Null-AID by the Null operator, the God-AID by the God operator, and the Enabler of Null-AID by the Enabler of Null operator.

[0354] As described, according to the present invention, a PAT is used for verifying the access right of a sender and the email access control among users is carried out when the verification result is valid, so that it becomes possible to disclose the information indicative of characteristics of a user while concealing the true identification of a user and carrying out communications appropriately according to this disclosed information while preventing conventionally possible attacks from a third person. In addition, even when a recipient receives an attack from a sender who maliciously utilizes the

anonymity, damages of a recipient due to that attack can be minimized.

[0355] Also, according to the present invention, the generation and the content change of the personalized access ticket can be made by the initiative of a user by using an AID assigned to each user and an Enabler defined in correspondence to the AID, so that it becomes possible to appropriately manage information such as that of a point of contact of each member of the group communication (mailing list, etc.) which changes dynamically.

[0356] Also, according to the present invention, a Null-AID and an Enabler of Null-AID can be introduced in order to carry out the generation of a new PAT (MakePAT) and the merging of PATs (MergePAT) without giving the member AID and the Enabler of the member AID to the holder of the PAT, so that it becomes possible to prevent the pretending using the member AID.

[0357] Also, according to the present invention, the Null-AID can be used only as the holder AID of the PAT (the Null-AID cannot be used as the member AID), that is PAT<AID_{Null} | $AID_{member1}$, $AID_{member2}$, ·····, $AID_{memberN}$ > is allowed, but PAT<AID_{holder} | AID_{Null}, $AID_{member1}$, $AID_{member2}$, ·····, $AID_{memberN}$ > is not allowed, so that the holder of PAT<AID_{holder} | AID_{member} > cannot produce PAT<AID_{Null} | AID_{member} > from this PAT<AID_{holder} | AID_{member} > as long as the holder does not know Enabler of AID_{member}.

[0358] Also, according to the present invention, a God-AID can be introduced in order to set up a read only attribute to the PAT, so that it becomes possible to fix the participants in the group communication.

[0359] Also, according to the present invention, the link information for uniquely specifying the AID can be introduced and the PAT can be given in terms of the link information such that the PAT does not contain the AID itself, so that it becomes possible to realize the receiving refusal function without using the AID itself.

[0360] It is to be noted that, besides those already mentioned above, many modifications and variations of the above embodiments may be made without departing from the novel and advantageous features of the present invention. Accordingly, all such modifications and variations are intended to be included within the scope of the appended claims.

Claims

1. A method of email access control, comprising the steps of:

   receiving a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, at a secure communication service for connecting communications between the sender and the receiver; and

   controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket at the secure communication service.

2. The method of claim 1, wherein at the controlling step the secure communication service authenticates the personalized access ticket presented by the sender, and refuses a delivery of the email when the personalized access ticket presented by the sender has been altered.

3. The method of claim 2, wherein the personalized access ticket is signed by a secret key of a secure processing device which issued the personalized access ticket, and at the controlling step the secure communication service authenticates the personalized access ticket by verifying a signature of the secure processing device in the personalized access ticket using a public key of the secure processing device.

4. The method of claim 1, wherein at the receiving step the secure communication service also receives the sender's identification presented by the sender along with the personalized access ticket, and at the controlling step the secure communication service checks whether the sender's identification presented by the sender is contained in the personalized access ticket presented by the sender, and refuses a delivery of the email when the sender's identification presented by the sender is not contained in the personalized access ticket presented by the sender.

5. The method of claim 1, wherein the personalized access ticket also contains a validity period indicating a period for which the personalized access ticket is valid, and at the controlling step the secure communication service checks the validity period contained in the personalized access ticket presented by the sender and refuses a delivery of the email when the personalized access ticket presented by the sender contains the validity period that has already been expired.

6. The method of claim 5, wherein the validity period of the personalized access ticket is set by a trusted third party.

7. The method of claim 1, further comprising the step of:

   issuing the personalized access ticket to the sender at a directory service for managing an

identification of each registrant and a disclosed information of each registrant which has a lower secrecy than a personal information, in a state which is accessible for search by unspecified many, in response to search conditions specified by the sender, by using an identification of a registrant whose disclosed information matches the search conditions as the recipient's identification and the sender's identification specified by the sender along with the search conditions.

8. The method of claim 1, further comprising the step of:

registering in advance the personalized access ticket containing an identification of a specific user from which a delivery of emails to a specific registrant is to be refused as the sender's identification and an identification of the specific registrant as the recipient's identification, at the secure communication service;
wherein the controlling step the secure communication service refuses a delivery of the email from the sender when the personalized access ticket presented by the sender is registered therein in advance at the registering step.

9. The method of claim 8, further comprising the step of:

deleting the personalized access ticket registered at the secure communication service upon request from the specific registrant who registered the personalized access ticket at the registering step.

10. The method of claim 1, wherein the personalized access ticket also contains a transfer control flag indicating whether or not the sender should be authenticated by the secure communication service, and at the controlling step, when the transfer control flag contained in the personalized access ticket indicates that the sender should be authenticated, the secure communication service authenticates the sender's identification presented by the sender and refuses a delivery of the email when an authentication of the sender's identification fails.

11. The method of claim 10, wherein the authentication of the sender's identification is realized by a challenge/response procedure between the sender and the secure communication service.

12. The method of claim 10, wherein the transfer control flag of the personalized access ticket is set by a trusted third party.

13. The method of claim 1, wherein the sender's identification and the recipient's identification in the personalized access ticket are given by real email addresses of the sender and the recipient.

14. The method of claim 1, wherein the sender's identification and the recipient's identification in the personalized access ticket are given by anonymous identifications of the sender and the recipient, where an anonymous identification of each user contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority.

15. The method of claim 14, wherein the anonymous identification of each user is an information containing the at least one fragment of the official identification of each user which is signed by the certification authority using a secret key of the certification authority.

16. The method of claim 14, wherein the official identification of each user is a character string uniquely assigned to each user by the certification authority and a public key of each user which are signed by a secret key of the certification authority.

17. The method of claim 14, further comprising the step of:

probabilistically identifying an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

18. The method of claim 1, wherein an anonymous identification of each user that contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority and a link information of each anonymous identification by which each anonymous identification can be uniquely identified are defined, and the sender's identification and the recipient's identification in the personalized access ticket are given by a link information of the anonymous identification of the sender and a link information of the anonymous identification of the recipient.

19. The method of claim 1, wherein the link information of each anonymous identification is an identifier uniquely assigned to each anonymous identification by the certification authority.

20. The method of claim 18, further comprising the step of:

probabilistically identifying an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender corresponding to the link information contained in a plurality of personalized access tickets used by the sender.

21. The method of claim 1, wherein the personalized access ticket contains a single sender's identification and a single recipient's identification in 1-to-1 correspondence.

22. The method of claim 1, wherein the personalized access ticket contains a single sender's identification and a plurality of recipient's identifications in 1-to-N correspondence, where N is an integer greater than 1.

23. The method of claim 22, wherein one identification among the single sender's identification and the plurality of recipient's identifications is a holder identification for identifying a holder of the personalized access ticket while other identifications among the single sender's identification and the plurality of recipient's identifications are member identifications for identifying members of a group to which the holder belongs.

24. The method of claim 23, further comprising the step of:

issuing an identification of each user and an enabler of the identification of each user indicating a right to change the personalized access ticket containing the identification of each user as the holder identification, to each user at a certification authority, such that prescribed processing on the personalized access ticket can be carried out at a secure processing device only by a user who presented both the holder identification contained in the personalized access ticket and the enabler corresponding to the holder identification to the secure processing device.

25. The method of claim 24, wherein the certification authority issues the enabler of the identification of each user as an information indicating that it is the enabler and the identification of each user itself which are signed by a secret key of the certification authority.

26. The method of claim 24, wherein the prescribed processing includes a generation of a new personalized access ticket, a merging of a plurality of personalized access tickets, a splitting of one personalized access ticket into a plurality of personalized access tickets, a changing of the holder of the personalized access ticket, changing of a validity period of the personalized access ticket, and a changing of a transfer control flag of the personalized access ticket.

27. The method of claim 26, wherein a special identification and a special enabler corresponding to the special identification which are known to all users are defined such that the generation of a new personalized access ticket and the changing of the holder of the personalized access ticket can be carried out by the holder of the personalized access ticket by using the special identification and the special enabler without using an enabler of a member identification.

28. The method of claim 27, wherein the special identification is defined to be capable of being used only as the holder identification of the personalized access ticket.

29. The method of claim 26, wherein a special identification which is known to all users is defined such that a read only attribute can be set to the personalized access ticket by using the special identification.

30. The method of claim 1, wherein at the controlling step, when the access right of the sender with respect to the recipient is verified according to the personalized access ticket, the secure communication service takes out the recipient's identification from the personalized access ticket by using the sender's identification presented by the sender, converts the mail by using a taken out recipient's identification into a format that can be interpreted by a mail transfer function for actually carrying out a mail delivery processing, and gives the mail after conversion to the mail transfer function by attaching the personalized access ticket.

31. A method of email access control, comprising the steps of:

defining an official identification of each user by which each user is uniquely identifiable by a certification authority, and an anonymous identification of each user containing at least one fragment of the official identification; and
identifying each user by the anonymous identification of each user in communications for emails on a communication network.

32. The method of claim 31, wherein the anonymous identification of each user is an information containing the at least one fragment of the official identification of each user which is signed by the

certification authority using a secret key of the certification authority.

33. The method of claim 31, wherein the official identification of each user is a character string uniquely assigned to each user by the certification authority and a public key of each user which are signed by a secret key of the certification authority.

34. The method of claim 31, further comprising the steps of:

receiving a personalized access ticket containing a sender's anonymous identification and a recipient's anonymous identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, at a secure communication service for connecting communications between the sender and the receiver; and

controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket at the secure communication service.

35. The method of claim 34, further comprising the step of:

probabilistically identifying an identity of the sender at the secure communication service by reconstructing the official identification of the sender while judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

36. The method of claim 31, wherein the defining step also defines a link information of each anonymous identification by which each anonymous identification can be uniquely identified, and each anonymous identification also contains the link information of each anonymous identification.

37. The method of claim 36, wherein the link information of each anonymous identification is an identifier uniquely assigned to each anonymous identification by the certification authority.

38. The method of claim 36, further comprising the steps of:

receiving a personalized access ticket containing a link information of a sender's anonymous identification and a link information of a recipient's anonymous identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, at a secure communication service for connecting communications between the sender and the receiver; and

controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket at the secure communication service.

39. The method of claim 38, further comprising the step of:

probabilistically identifying an identity of the sender by reconstructing the official identification of the sender while judging identity of a plurality of anonymous identifications of the sender corresponding to the link information contained in a plurality of personalized access tickets used by the sender.

40. A communication system realizing email access control, comprising:

a communication network to which a plurality of user terminals are connected; and

a secure communication service device for connecting communications between the sender and the receiver on the communication network, by receiving a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, and controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket.

41. The system of claim 40, wherein the secure communication service device authenticates the personalized access ticket presented by the sender, and refuses a delivery of the email when the personalized access ticket presented by the sender has been altered.

42. The system of claim 41, further comprising:

a secure processing device for issuing the personalized access ticket which is signed by a secret key of the secure processing device;

wherein the secure communication service device authenticates the personalized access ticket by verifying a signature of the secure processing device in the personalized access ticket using a public key of the secure process-

ing device.

43. The system of claim 40, wherein the secure communication service device also receives the sender's identification presented by the sender along with the personalized access ticket, checks whether the sender's identification presented by the sender is contained in the personalized access ticket presented by the sender, and refuses a delivery of the email when the sender's identification presented by the sender is not contained in the personalized access ticket presented by the sender.

44. The system of claim 40, wherein the personalized access ticket also contains a validity period indicating a period for which the personalized access ticket is valid, and the secure communication service device checks the validity period contained in the personalized access ticket presented by the sender and refuses a delivery of the email when the personalized access ticket presented by the sender contains the validity period that has already been expired.

45. The system of claim 44, further comprising:

a trusted third party for setting the validity period of the personalized access ticket.

46. The system of claim 40, further comprising:

a directory service device for managing an identification of each registrant and and a disclosed information of each registrant which has a lower secrecy than a personal information, in a state which is accessible for search by unspecified many, and issuing the personalized access ticket to the sender in response to search conditions specified by the sender, by using an identification of a registrant whose disclosed information matches the search conditions as the recipient's identification and the sender's identification specified by the sender along with the search conditions.

47. The system of claim 40, wherein the secure communication service device registers in advance the personalized access ticket containing an identification of a specific user from which a delivery of emails to a specific registrant is to be refused as the sender's identification and an identification of the specific registrant as the recipient's identification, and refuses a delivery of the email from the sender when the personalized access ticket presented by the sender is registered therein in advance.

48. The system of claim 47, wherein the secure communication service device deletes the personalized access ticket registered therein upon request from the specific registrant who registered the personalized access ticket.

49. The system of claim 40, wherein the personalized access ticket also contains a transfer control flag indicating whether or not the sender should be authenticated by the secure communication service, and when the transfer control flag contained in the personalized access ticket indicates that the sender should be authenticated, the secure communication service device authenticates the sender's identification presented by the sender and refuses a delivery of the email when an authentication of the sender's identification fails.

50. The system of claim 49, wherein the authentication of the sender's identification is realized by a challenge/response procedure between the sender and the secure communication service device.

51. The system of claim 49, further comprising a trusted third party for setting the transfer control flag of the personalized access ticket.

52. The system of claim 40, wherein the sender's identification and the recipient's identification in the personalized access ticket are given by real email addresses of the sender and the recipient.

53. The system of claim 40, further comprising:

a certification authority device for issuing an anonymous identification of each user which contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by the certification authority device;
wherein the sender's identification and the recipient's identification in the personalized access ticket are given by anonymous identifications of the sender and the recipient.

54. The system of claim 53, wherein the anonymous identification of each user is an information containing the at least one fragment of the official identification of each user which is signed by the certification authority device using a secret key of the certification authority device.

55. The system of claim 53, wherein the official identification of each user is a character string uniquely assigned to each user by the certification authority device and a public key of each user which are signed by a secret key of the certification authority device.

56. The system of claim 53, wherein the secure com-

munication service device probabilistically identifies an identity of the sender by reconstructing the official identification of the sender while judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

57. The system of claim 40, further comprising:

a certification authority device for issuing an anonymous identification of each user which contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by the certification authority device and a link information of each anonymous identification by which each anonymous identification can be uniquely identified; wherein the sender's identification and the recipient's identification in the personalized access ticket are given by a link information of the anonymous identification of the sender and a link information of the anonymous identification of the recipient.

58. The system of claim 57, wherein the link information of each anonymous identification is an identifier uniquely assigned to each anonymous identification by the certification authority device.

59. The system of claim 57, wherein the secure communication service device probabilistically identifies an identity of the sender by reconstructing the official identification of the sender while judging identity of a plurality of anonymous identifications of the sender corresponding to the link information contained in a plurality of personalized access tickets used by the sender.

60. The system of claim 40, wherein the personalized access ticket contains a single sender's identification and a single recipient's identification in 1-to-1 correspondence.

61. The system of claim 40, wherein the personalized access ticket contains a single sender's identification and a plurality of recipient's identifications in 1-to-N correspondence, where N is an integer greater than 1.

62. The system of claim 61, wherein one identification among the single sender's identification and the plurality of recipient's identifications is a holder identification for identifying a holder of the personalized access ticket while other identifications among the single sender's identification and the plurality of recipient's identifications are member identifications for identifying members of a group to which the holder belongs.

63. The system of claim 62, further comprising:

a certification authority device for issuing to each user an identification of each user and an enabler of the identification of each user indicating a right to change the personalized access ticket containing the identification of each user as the holder identification; and
a secure processing device at which prescribed processing on the personalized access ticket can be carried out only by a user who presented both the holder identification contained in the personalized access ticket and the enabler corresponding to the holder identification to the secure processing device.

64. The system of claim 63, wherein the certification authority device issues the enabler of the identification of each user as an information indicating that it is the enabler and the identification of each user itself which are signed by a secret key of the certification authority device.

65. The system of claim 63, wherein the prescribed processing includes a generation of a new personalized access ticket, a merging of a plurality of personalized access tickets, a splitting of one personalized access ticket into a plurality of personalized access tickets, a changing of the holder of the personalized access ticket, changing of a validity period of the personalized access ticket, and a changing of a transfer control flag of the personalized access ticket.

66. The system of claim 65, wherein a special identification and a special enabler corresponding to the special identification which are known to all users are defined such that the generation of a new personalized access ticket and the changing of the holder of the personalized access ticket can be carried out by the holder of the personalized access ticket by using the special identification and the special enabler without using an enabler of a member identification.

67. The system of claim 66, wherein the special identification is defined to be capable of being used only as the holder identification of the personalized access ticket.

68. The system of claim 65, wherein a special identification which is known to all users is defined such that a read only attribute can be set to the personalized access ticket by using the special identification.

69. The system of claim 40, wherein when the access right of the sender with respect to the recipient is

verified according to the personalized access ticket, the secure communication service device takes out the recipient's identification from the personalized access ticket by using the sender's identification presented by the sender, converts the mail by using a taken out recipient's identification into a format that can be interpreted by a mail transfer function for actually carrying out a mail delivery processing, and gives the mail after conversion to the mail transfer function by attaching the personalized access ticket.

70. A communication system realizing email access control, comprising:

a certification authority device for defining an official identification of each user by which each user is uniquely identifiable by the certification authority device, and an anonymous identification of each user which contains at least one fragment of the official identification; and

a communication network on which each user is identified by the anonymous identification of each user in communications for emails on the communication network.

71. The system of claim 70, wherein the anonymous identification of each user is an information containing the at least one fragment of the official identification of each user which is signed by the certification authority device using a secret key of the certification authority device.

72. The system of claim 70, wherein the official identification of each user is a character string uniquely assigned to each user by the certification authority device and a public key of each user which are signed by a secret key of the certification authority device.

73. The system of claim 70, further comprising:

a secure communication service device for connecting communications between the sender and the receiver on the communication network, by receiving a personalized access ticket containing a sender's anonymous identification and a recipient's anonymous identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, and controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket.

74. The system of claim 73, wherein the secure communication service device probabilistically identifies an identity of the sender by reconstructing the official identification of the sender while judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

75. The system of claim 70, wherein the certification authority device also defines a link information of each anonymous identification by which each anonymous identification can be uniquely identified, and each anonymous identification also contains the link information of each anonymous identification.

76. The system of claim 75, wherein the link information of each anonymous identification is an identifier uniquely assigned to each anonymous identification by the certification authority device.

77. The system of claim 75, further comprising:

a secure communication service device for connecting communications between the sender and the receiver on the communication network, by receiving a personalized access ticket containing a link information of a sender's anonymous identification and a link information of a recipient's anonymous identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, and controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket.

78. The system of claim 77, wherein the secure communication service device probabilistically identifies an identity of the sender by reconstructing the official identification of the sender while judging identity of a plurality of link informations of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

79. A secure communication service device for use in a communication system realizing email access control, comprising:

a computer hardware; and

a computer software for causing the computer hardware to connect communications between the sender and the receiver, by receiving a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is presented by a

sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email, and controlling accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket.

80. The secure communication service device of claim 79,
wherein the computer software causes the computer hardware to authenticate the personalized access ticket presented by the sender, and refuse a delivery of the email when the personalized access ticket presented by the sender has been altered.

81. The secure communication service device of claim 80,
wherein the personalized access ticket is signed by a secret key of a secure processing device which issued the personalized access ticket, and the computer software causes the computer hardware to authenticate the personalized access ticket by verifying a signature of the secure processing device in the personalized access ticket using a public key of the secure processing device.

82. The secure communication service device of claim 79,
wherein the computer software causes the computer hardware to also receive the sender's identification presented by the sender along with the personalized access ticket, check whether the sender's identification presented by the sender is contained in the personalized access ticket presented by the sender, and refuse a delivery of the email when the sender's identification presented by the sender is not contained in the personalized access ticket presented by the sender.

83. The secure communication service device of claim 79,
wherein the personalized access ticket also contains a validity period indicating a period for which the personalized access ticket is valid, and the computer software causes the computer hardware to check the validity period contained in the personalized access ticket presented by the sender and refuse a delivery of the email when the personalized access ticket presented by the sender contains the validity period that has already been expired.

84. The secure communication service device of claim 79,
wherein the computer software causes the computer hardware to register in advance the personalized access ticket containing an identification of a specific user from which a delivery of emails to a specific registrant is to be refused as the sender's identification and an identification of the specific registrant as the recipient's identification, at the secure communication service device, and refuse a delivery of the email from the sender when the personalized access ticket presented by the sender is registered at the secure communication service device in advance.

85. The secure communication service device of claim 84,
wherein the computer software causes the computer hardware to delete the personalized access ticket registered at the secure communication service device upon request from the specific registrant who registered the personalized access ticket.

86. The secure communication service device of claim 79,
wherein the personalized access ticket also contains a transfer control flag indicating whether or not the sender should be authenticated by the secure communication service device, and when the transfer control flag contained in the personalized access ticket indicates that the sender should be authenticated, the computer software causes the computer hardware to authenticate the sender's identification presented by the sender and refuse a delivery of the email when an authentication of the sender's identification fails.

87. The secure communication service device of claim 86,
wherein the computer software causes the computer hardware to realize the authentication of the sender's identification by a challenge/response procedure between the sender and the secure communication service device.

88. The secure communication service device of claim 79,
wherein the sender's identification and the recipient's identification in the personalized access ticket are given by anonymous identifications of the sender and the recipient, where an anonymous identification of each user contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority, and the computer software also causes the computer hardware to probabilistically identify an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

89. The secure communication service device of claim 79,

wherein an anonymous identification of each user that contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority and a link information of each anonymous identification by which each anonymous identification can be uniquely identified are defined, the sender's identification and the recipient's identification in the personalized access ticket are given by a link information of the anonymous identification of the sender and a link information of the anonymous identification of the recipient, and the computer software also causes the computer hardware to probabilistically identify an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender corresponding to the link information contained in a plurality of personalized access tickets used by the sender.

90. The secure communication service device of claim 79,
wherein when the access right of the sender with respect to the recipient is verified according to the personalized access ticket, the computer software causes the computer hardware to take out the recipient's identification from the personalized access ticket by using the sender's identification presented by the sender, convert the mail by using a taken out recipient's identification into a format that can be interpreted by a mail transfer function for actually carrying out a mail delivery processing, and give the mail after conversion to the mail transfer function by attaching the personalized access ticket.

91. A secure processing device for use in a communication system realizing email access control, comprising:

a computer hardware; and
a computer software for causing the computer hardware to receive a request for a personalized access ticket from a user, and issue a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is signed by a secret key of the secure processing device.

92. A directory service device for use in a communication system realizing email access control, comprising:

a computer hardware; and
a computer software for causing the computer hardware to manage an identification of each registrant and a disclosed information of each registrant which has a lower secrecy than a personal information, in a state which is accessible for search by unspecified many, and issue a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, to the sender in response to search conditions specified by the sender, by using an identification of a registrant whose disclosed information matches the search conditions as the recipient's identification and the sender's identification specified by the sender along with the search conditions.

93. A certification authority device for use in a communication system realizing email access control, comprising:

a computer hardware; and
a computer software for causing the computer hardware to issue to each user an official identification of each user by which each user is uniquely identifiable by the certification authority device, and an anonymous identification of each user which contains at least one fragment of the official identification.

94. A certification authority device for use in a communication system realizing email access control, comprising:

a computer hardware; and
a computer software for causing the computer hardware to issue to each user an identification of each user and an enabler of the identification of each user indicating a right to change any personalized access ticket that contains the identification of each user as a holder identification, where the personalized access ticket generally contains a sender's identification and a plurality of recipient's identifications in correspondence, and one of the sender's identification and the recipient's identifications is a holder identification.

95. A secure processing device for use in a communication system realizing email access control, comprising:

a computer hardware; and
a computer software for causing the computer hardware to receive from a user a request for prescribed processing on a personalized access ticket containing a sender's identification and a plurality of recipient's identifications in correspondence, where one of the sender's identification and the recipient's identifications is a holder identification, and execute the prescribed processing on the personalized access ticket when the user presented both the holder

identification contained in the personalized access ticket and an enabler corresponding to the holder identification which indicates a right to change the personalized access ticket containing the identification of the user as the holder identification.

96. A computer usable medium having computer readable program code means embodied therein for causing a computer to function as a secure communication service device for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to receive a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is presented by a sender who wishes to send an email to a recipient so as to specify the recipient as an intended destination of the email; and
second computer readable program code means for causing said computer to control accesses between the sender and the recipient by verifying an access right of the sender with respect to the recipient according to the personalized access ticket, so as to connect communications between the sender and the receiver on the communication network.

97. The computer usable medium of claim 96, the second computer readable program code means causes said computer to authenticate the personalized access ticket presented by the sender, and refuse a delivery of the email when the personalized access ticket presented by the sender has been altered.

98. The computer usable medium of claim 97, wherein the personalized access ticket is signed by a secret key of a secure processing device which issued the personalized access ticket, and the second computer readable program code means causes said computer to authenticate the personalized access ticket by verifying a signature of the secure processing device in the personalized access ticket using a public key of the secure processing device.

99. The computer usable medium of claim 96, wherein the first computer readable program code means causes said computer to also receive the sender's identification presented by the sender along with the personalized access ticket, and the second computer readable program code means causes said computer to check whether the sender's identification presented by the sender is contained in the personalized access ticket presented by the sender and refuse a delivery of the email when the sender's identification presented by the sender is not contained in the personalized access ticket presented by the sender.

100. The computer usable medium of claim 96, wherein the personalized access ticket also contains a validity period indicating a period for which the personalized access ticket is valid, and the second computer readable program code means causes said computer to check the validity period contained in the personalized access ticket presented by the sender and refuse a delivery of the email when the personalized access ticket presented by the sender contains the validity period that has already been expired.

101. The computer usable medium of claim 96, wherein the second computer readable program code means causes said computer to register in advance the personalized access ticket containing an identification of a specific user from which a delivery of emails to a specific registrant is to be refused as the sender's identification and an identification of the specific registrant as the recipient's identification, at the secure communication service device, and refuse a delivery of the email from the sender when the personalized access ticket presented by the sender is registered at the secure communication service device in advance.

102. The computer usable medium of claim 101, wherein the second computer readable program code means causes said computer to delete the personalized access ticket registered at the secure communication service device upon request from the specific registrant who registered the personalized access ticket.

103. The computer usable medium of claim 96, wherein the personalized access ticket also contains a transfer control flag indicating whether or not the sender should be authenticated by the secure communication service device, and when the transfer control flag contained in the personalized access ticket indicates that the sender should be authenticated, the second computer readable program code means causes said computer to authenticate the sender's identification presented by the sender and refuse a delivery of the email when an authentication of the sender's identification fails.

104. The computer usable medium of claim 103, wherein the second computer readable program code means causes said computer to realize the authentication of the sender's identification by a challenge/response procedure between the sender and the secure communication service device.

105. The computer usable medium of claim 96, wherein the sender's identification and the recipient's identification in the personalized access ticket are given by anonymous identifications of the sender and the recipient, where an anonymous identification of each user contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority, and the second computer readable program code means also causes said computer to probabilistically identify an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender contained in a plurality of personalized access tickets used by the sender.

106. The computer usable medium of claim 96, wherein an anonymous identification of each user that contains at least one fragment of an official identification of each user by which each user is uniquely identifiable by a certification authority and a link information of each anonymous identification by which each anonymous identification can be uniquely identified are defined, the sender's identification and the recipient's identification in the personalized access ticket are given by a link information of the anonymous identification of the sender and a link information of the anonymous identification of the recipient, and the second computer readable program code means also causes said computer to probabilistically identify an identity of the sender by reconstructing the official identification of the sender by judging identity of a plurality of anonymous identifications of the sender corresponding to the link information contained in a plurality of personalized access tickets used by the sender.

107. The computer usable medium of claim 96, wherein when the access right of the sender with respect to the recipient is verified according to the personalized access ticket, the second computer readable program code means causes said computer to take out the recipient's identification from the personalized access ticket by using the sender's identification presented by the sender, convert the mail by using a taken out recipient's identification into a format that can be interpreted by a mail transfer function for actually carrying out a mail delivery processing, and give the mail after conversion to the mail transfer function by attaching the personalized access ticket.

108. A computer usable medium having computer readable program code means embodied therein for causing a computer to function as a secure processing device for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to receive a request for a personalized access ticket from a user; and

second computer readable program code means for causing said computer to issue the personalized access ticket containing a sender's identification and a recipient's identification in correspondence, which is signed by a secret key of the secure processing device.

109. A computer usable medium having computer readable program code means embodied therein for causing a computer to function as a directory service devicer for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to manage an identification of each registrant and a disclosed information of each registrant which has a lower secrecy than a personal information, in a state which is accessible for search by unspecified many, and

second computer readable program code means for causing said computer to issue a personalized access ticket containing a sender's identification and a recipient's identification in correspondence, to the sender in response to search conditions specified by the sender, by using an identification of a registrant whose disclosed information matches the search conditions as the recipient's identification and the sender's identification specified by the sender along with the search conditions.

110. A computer usable medium having computer readable program code means embodied therein for causing a computer to function as a certification authority device for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to issue to each user an official identification of each user by which each user is uniquely identifiable by the certification authority device; and

second computer readable program code means for causing said computer to issue to each user an anonymous identification of each user which contains at least one fragment of the official identification.

111. A computer usable medium having computer read-

able program code means embodied therein for causing a computer to function as a certification authority device for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to issue to each user an identification of each user; and

second computer readable program code means for causing said computer to issue to each user an enabler of the identification of each user indicating a right to change any personalized access ticket that contains the identification of each user as a holder identification, where the persnalized access ticket generally contains a sender's identification and a plurality of recipient's identifications in correspondence, and one of the sender's identification and the recipient's identifications is a holder identification.

112.A computer usable medium having computer readable program code means embodied therein for causing a computer to function as a secure processing device for use in a communication system realizing email access control, the computer readable program code means includes:

first computer readable program code means for causing said computer to receive from a user a request for prescribed processing on a personalized access ticket containing a sender's identification and a plurality of recipient's identifications in correspondence, where one of the sender's identification and the recipient's identifications is a holder identification; and

second computer readable program code means for causing said computer to execute the prescribed processing on the personalized access ticket when the user presented both the holder identification contained in the personalized access ticket and an enabler corresponding to the holder identification which indicates a right to change the personalized access ticket containing the identification of the user as the holder identification.

FIG.1

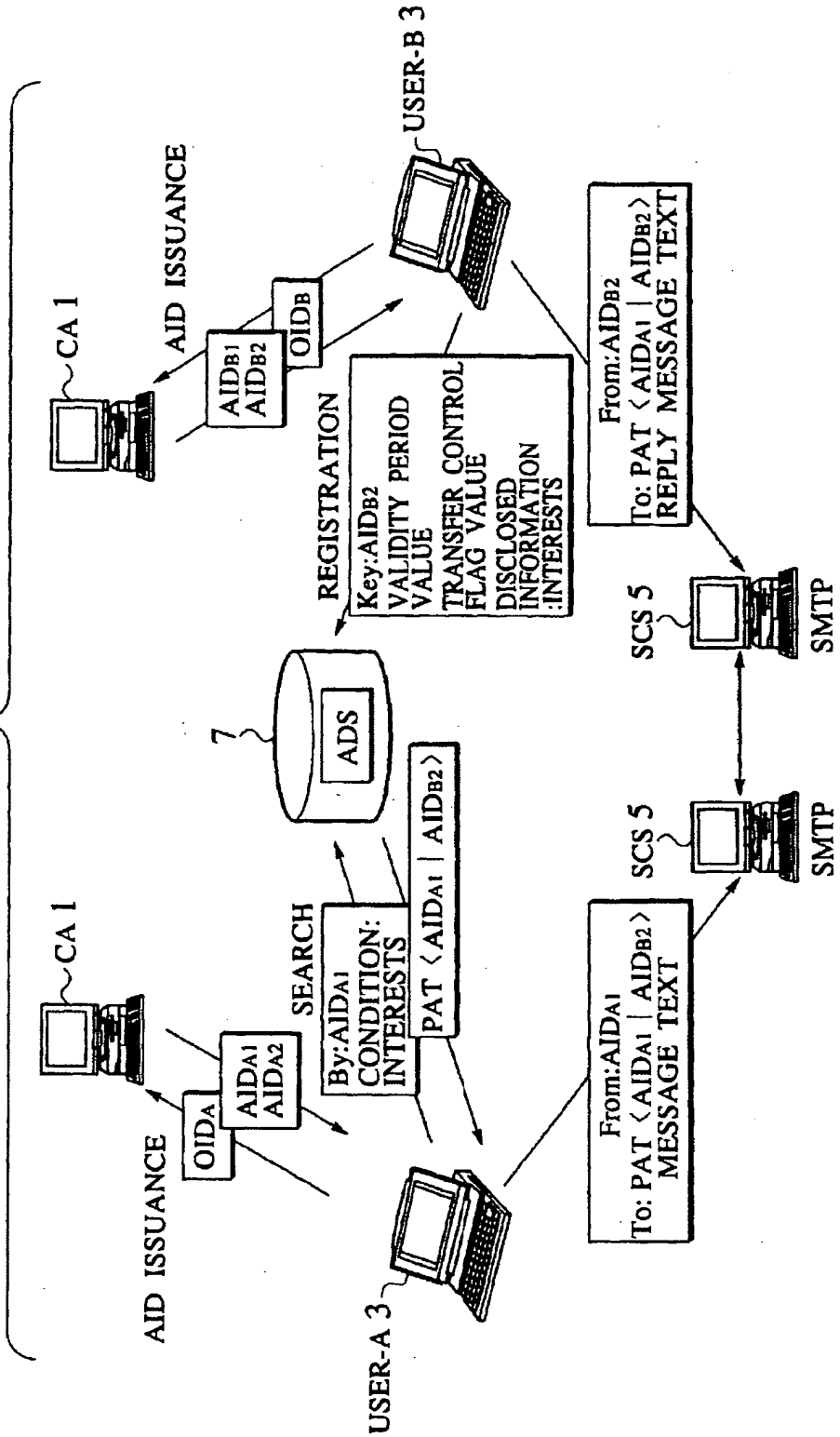


FIG.2

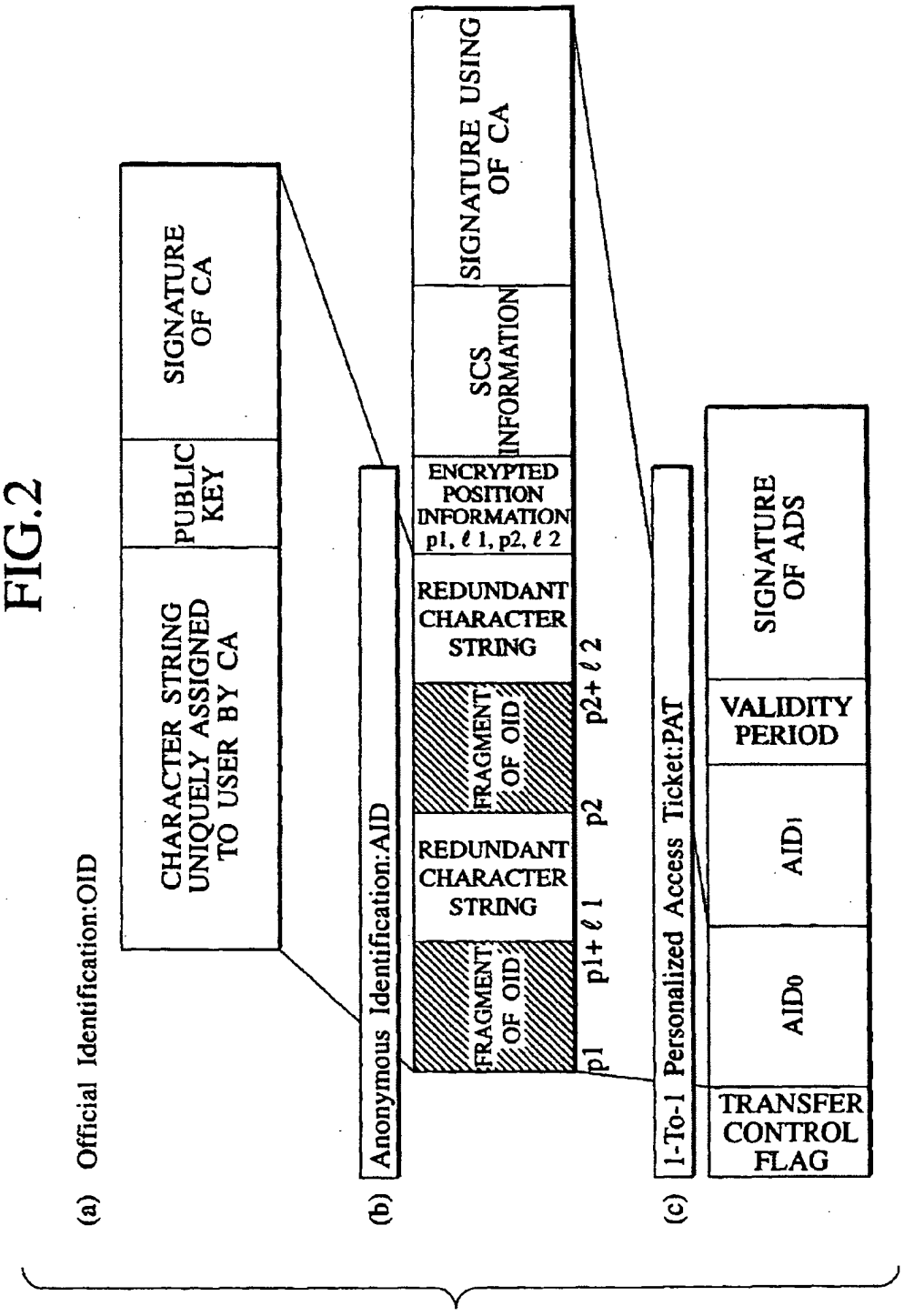


FIG.3

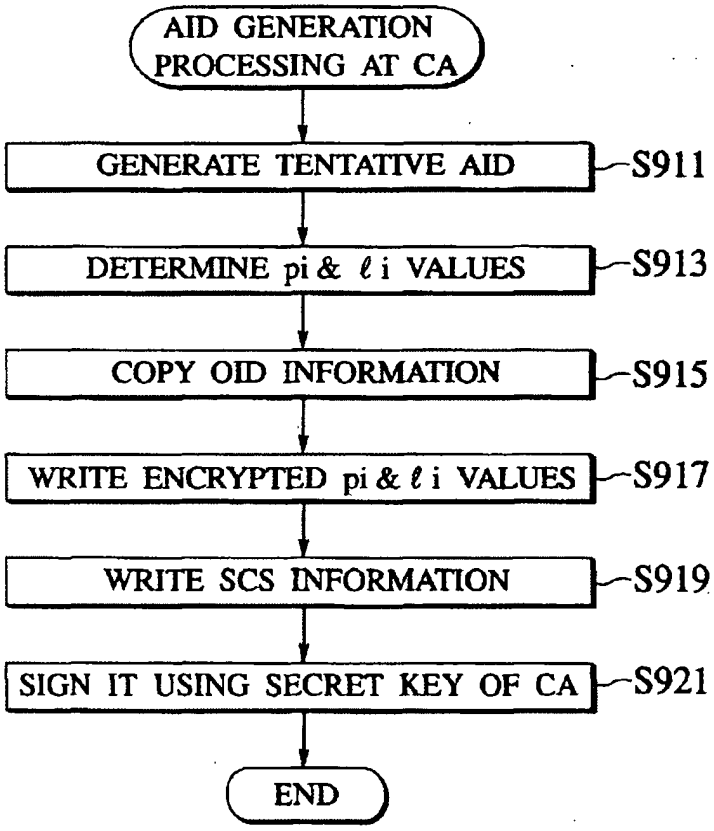


FIG.4

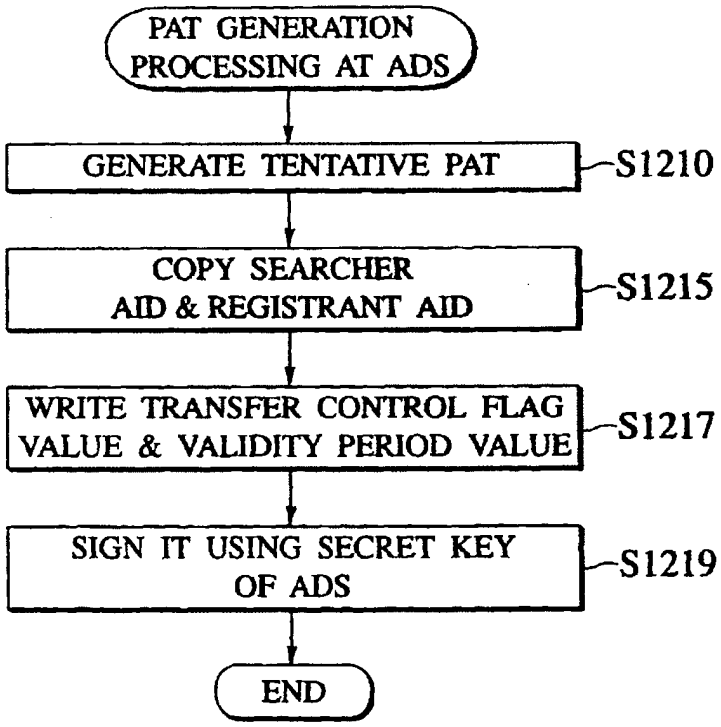


FIG.5

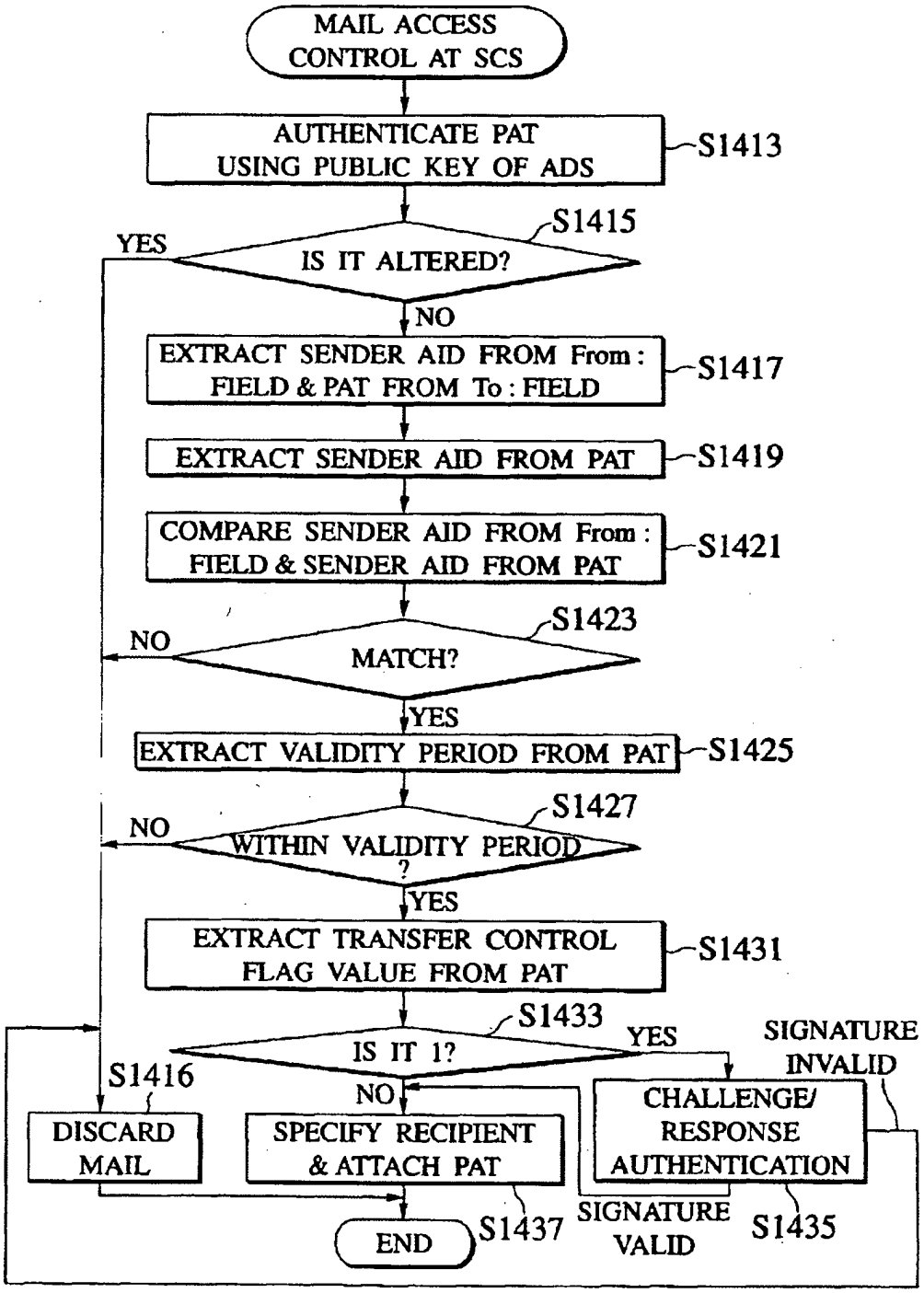


FIG.6

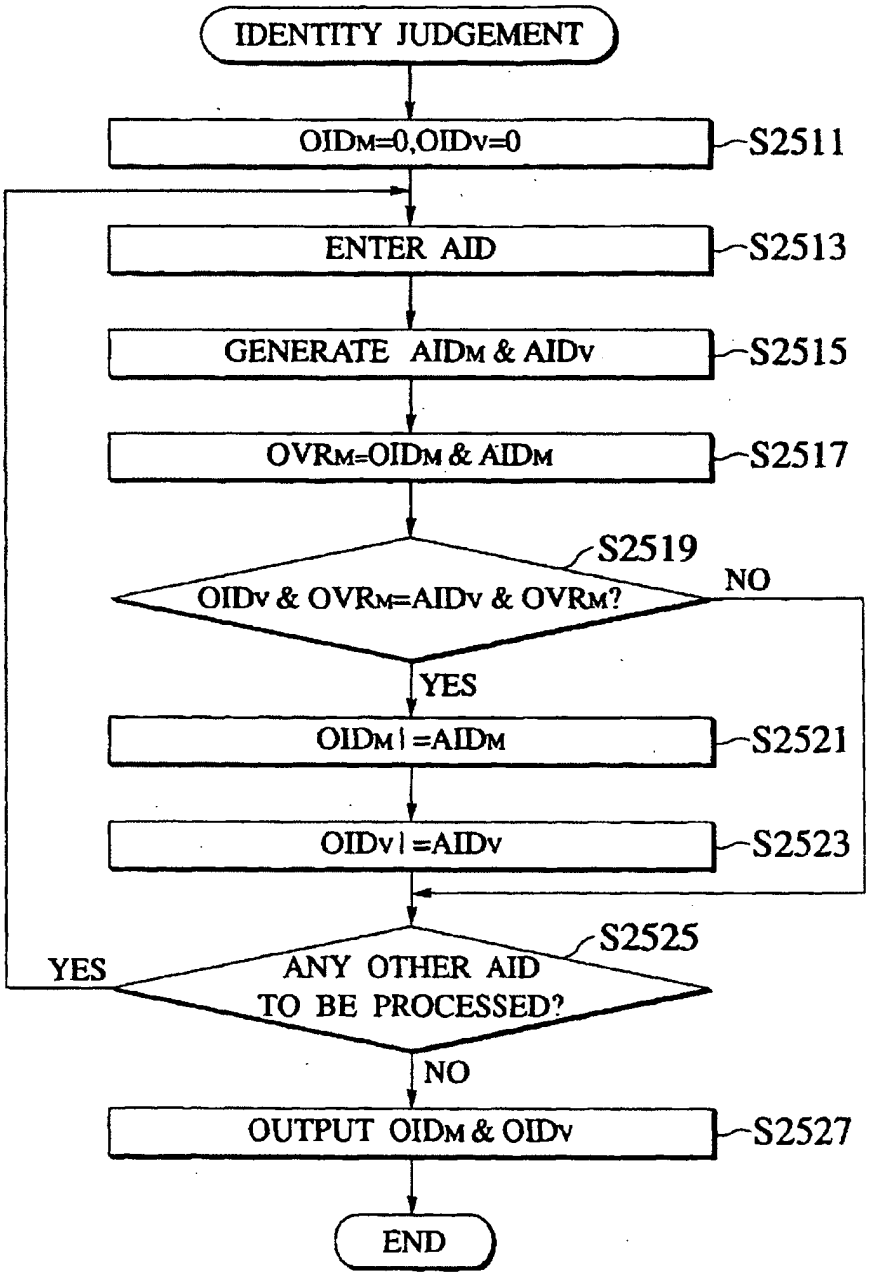


FIG.7

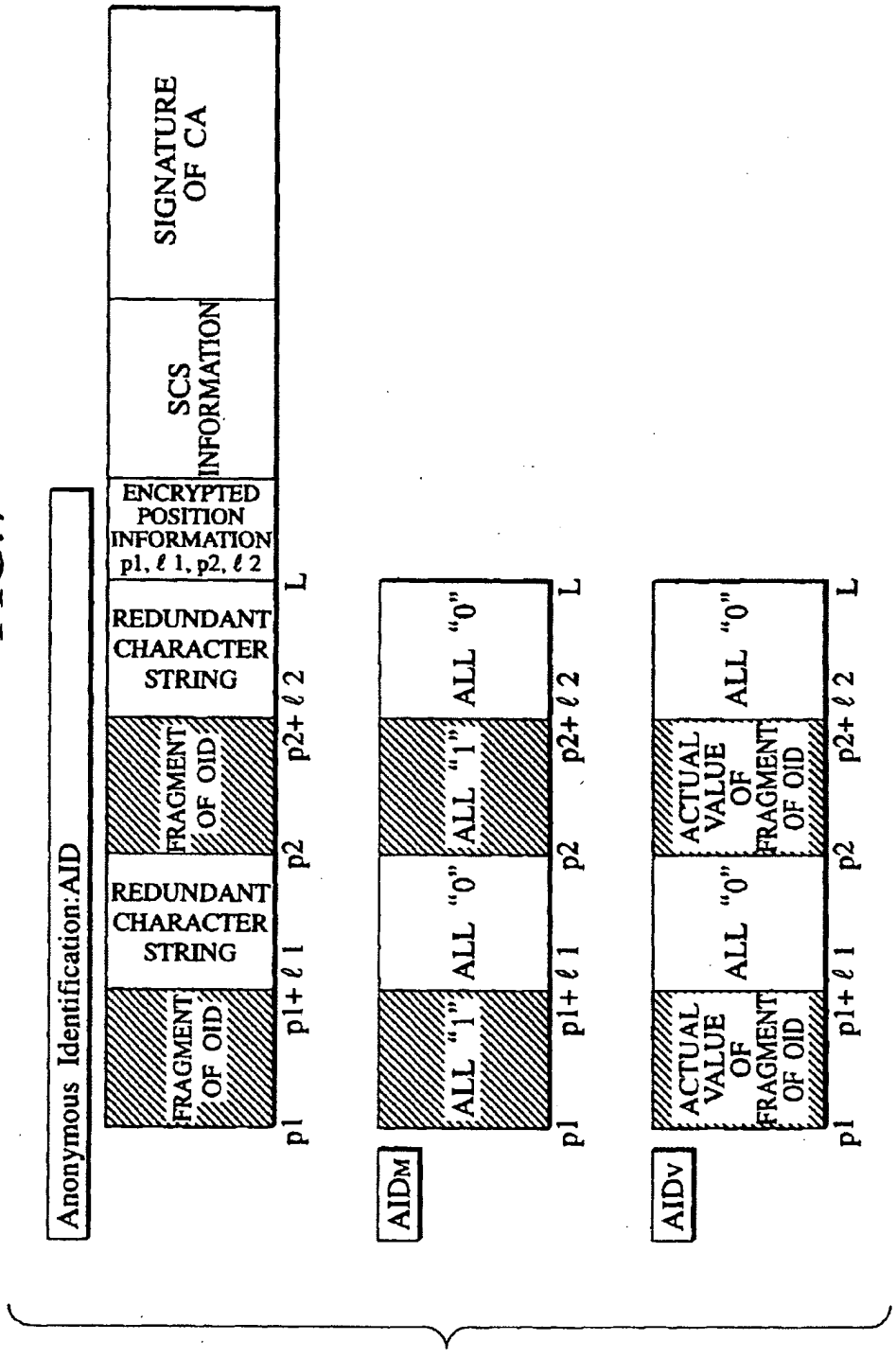


FIG.8

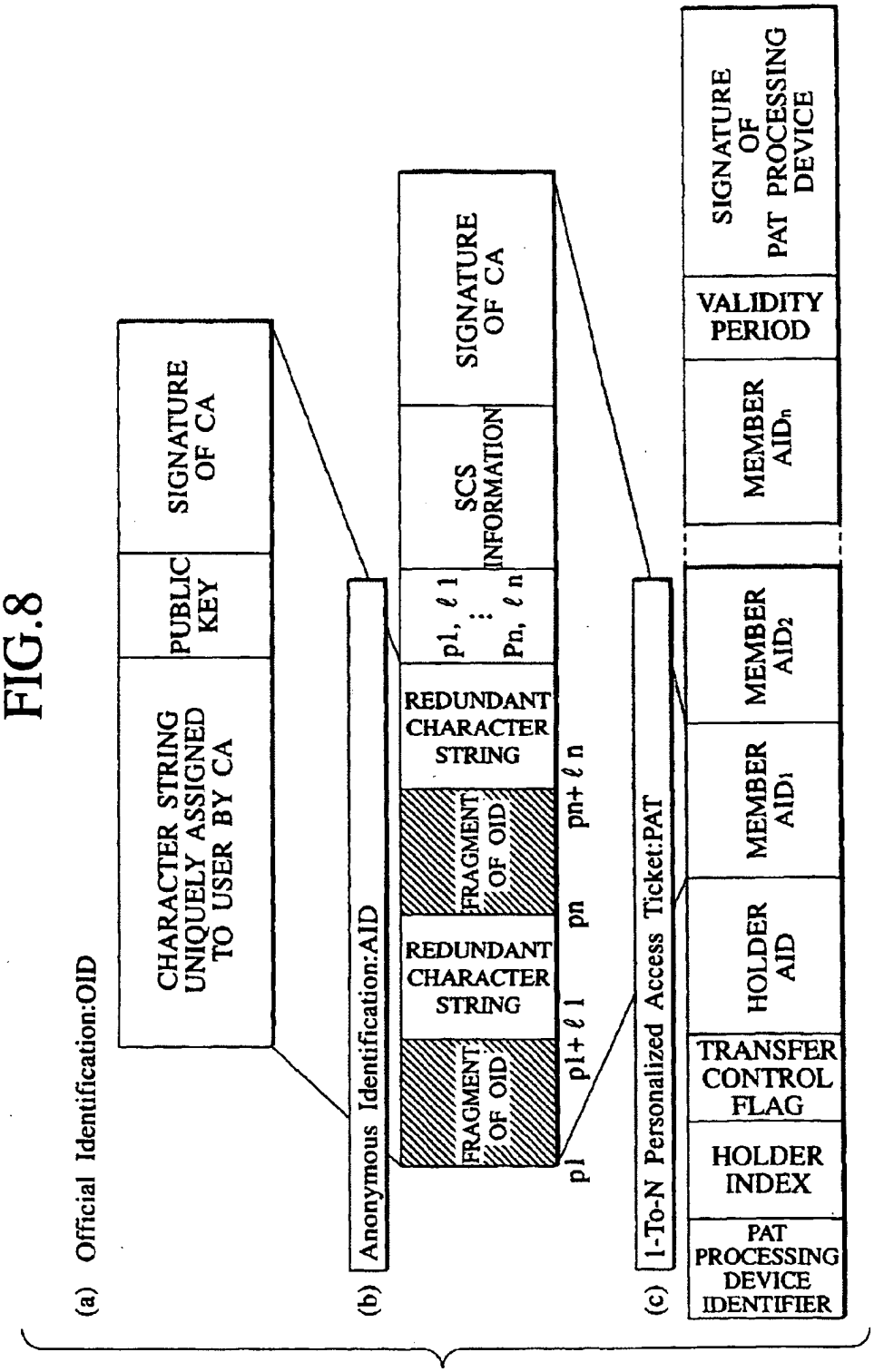


FIG.9

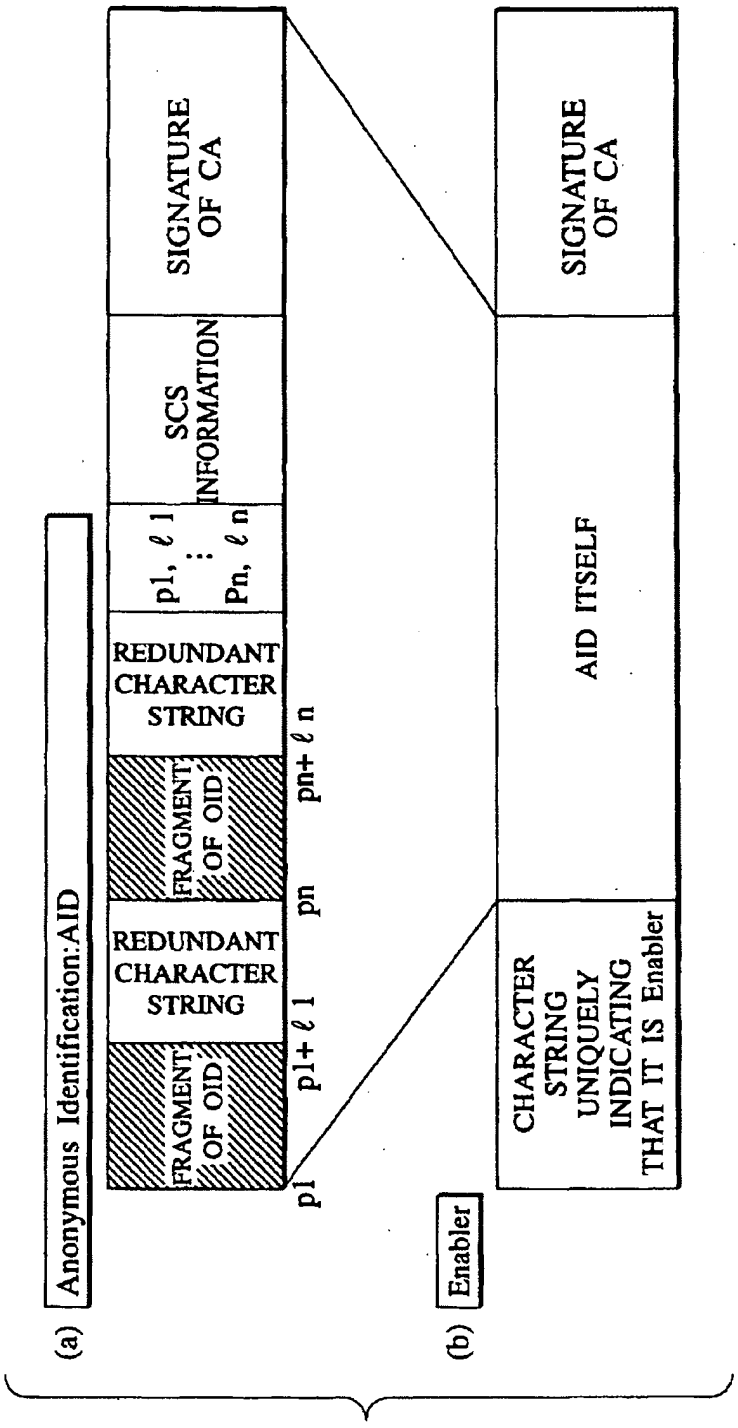


FIG.10

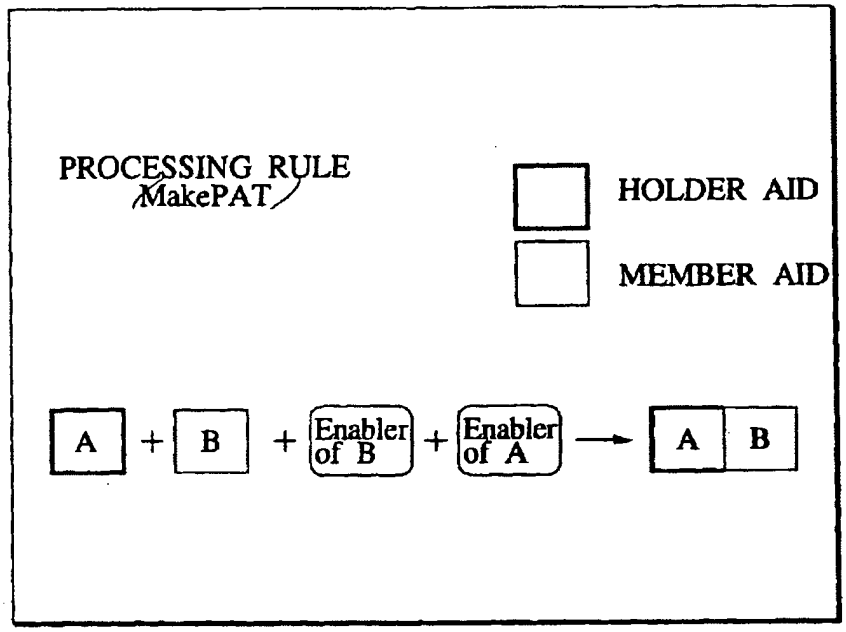


FIG.11

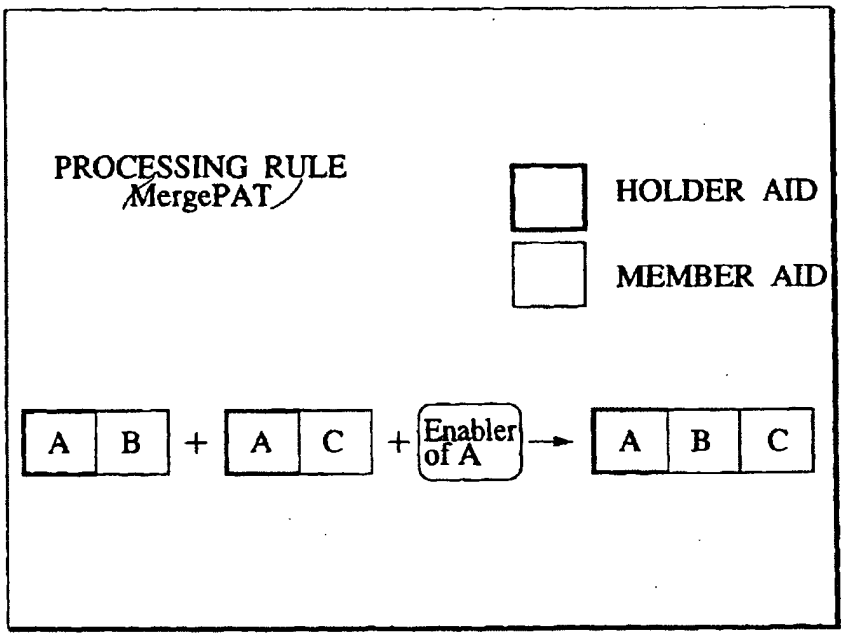


FIG.12

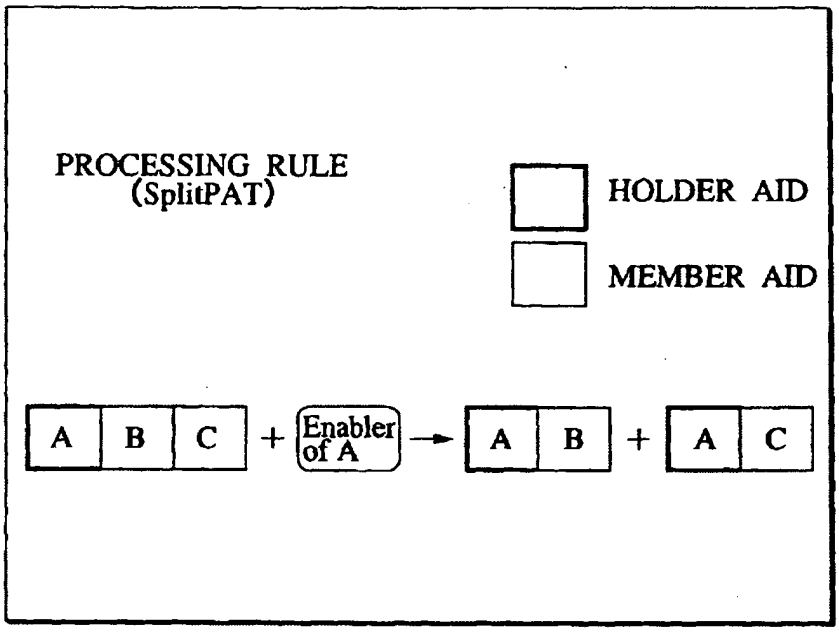


FIG.13

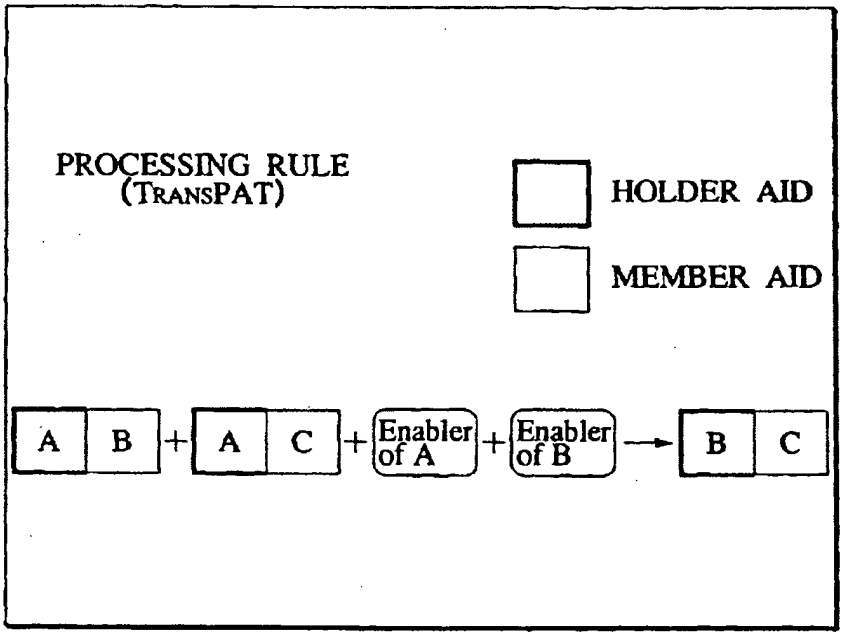


FIG.14

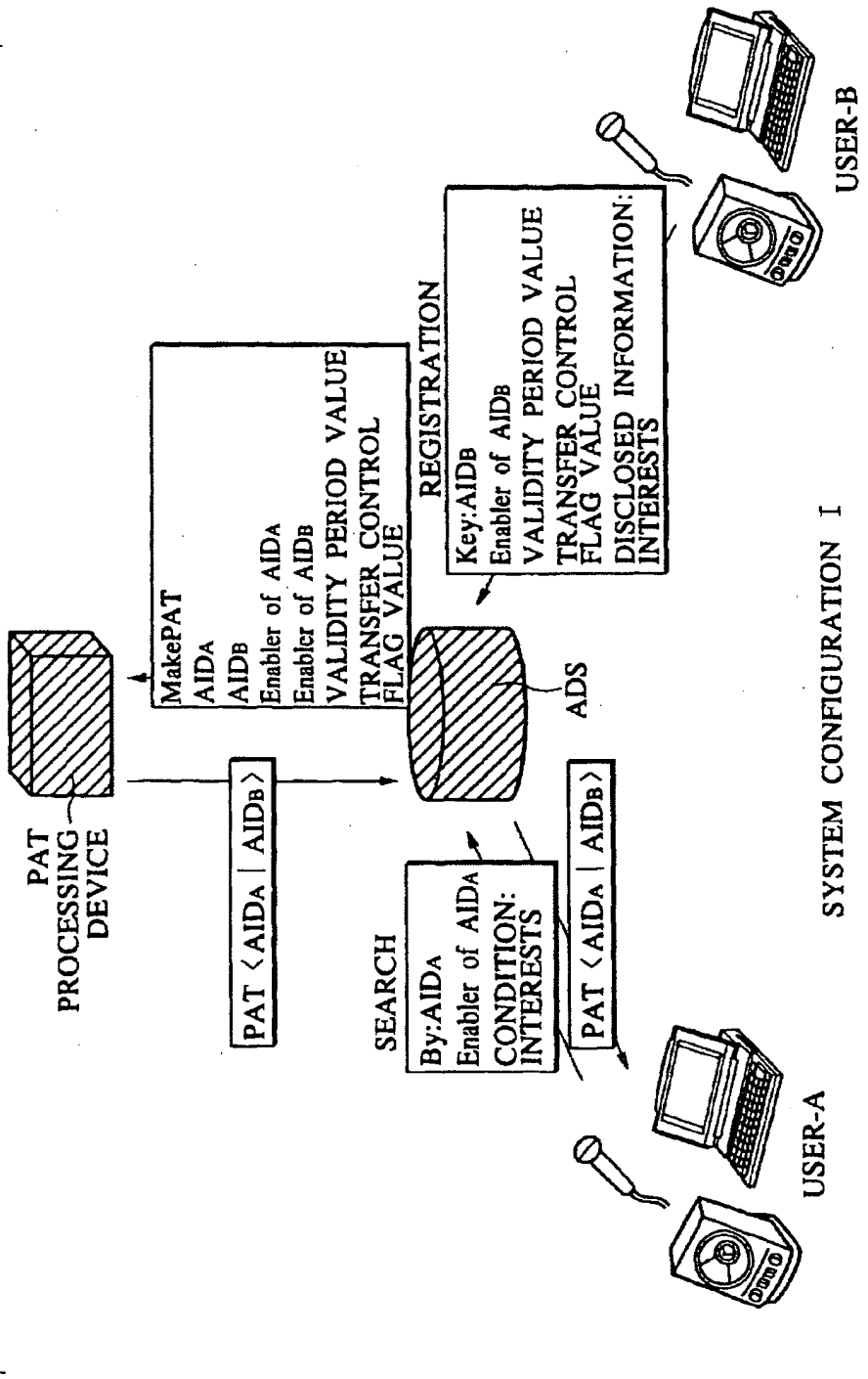


FIG.15

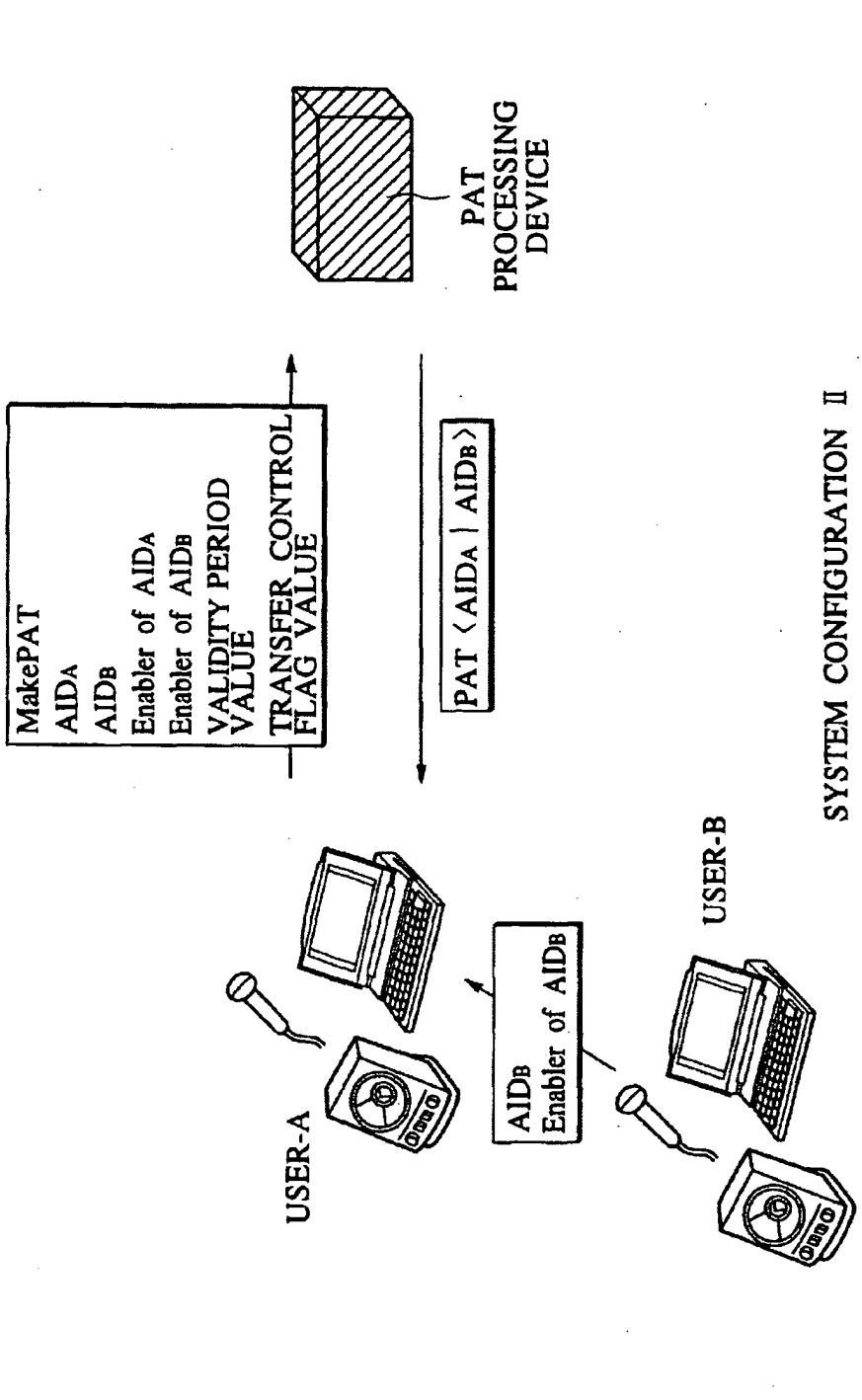


FIG.16

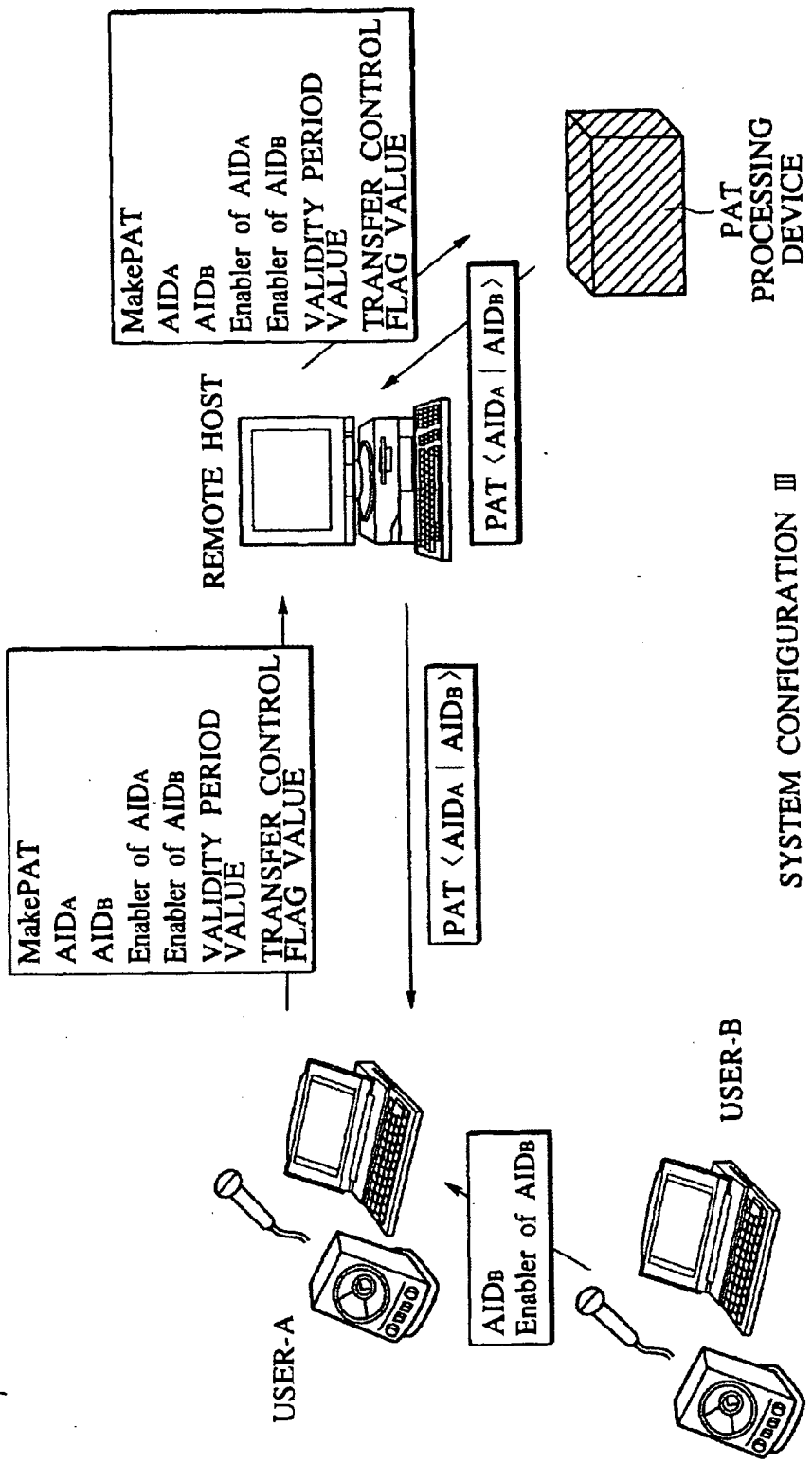


FIG.17

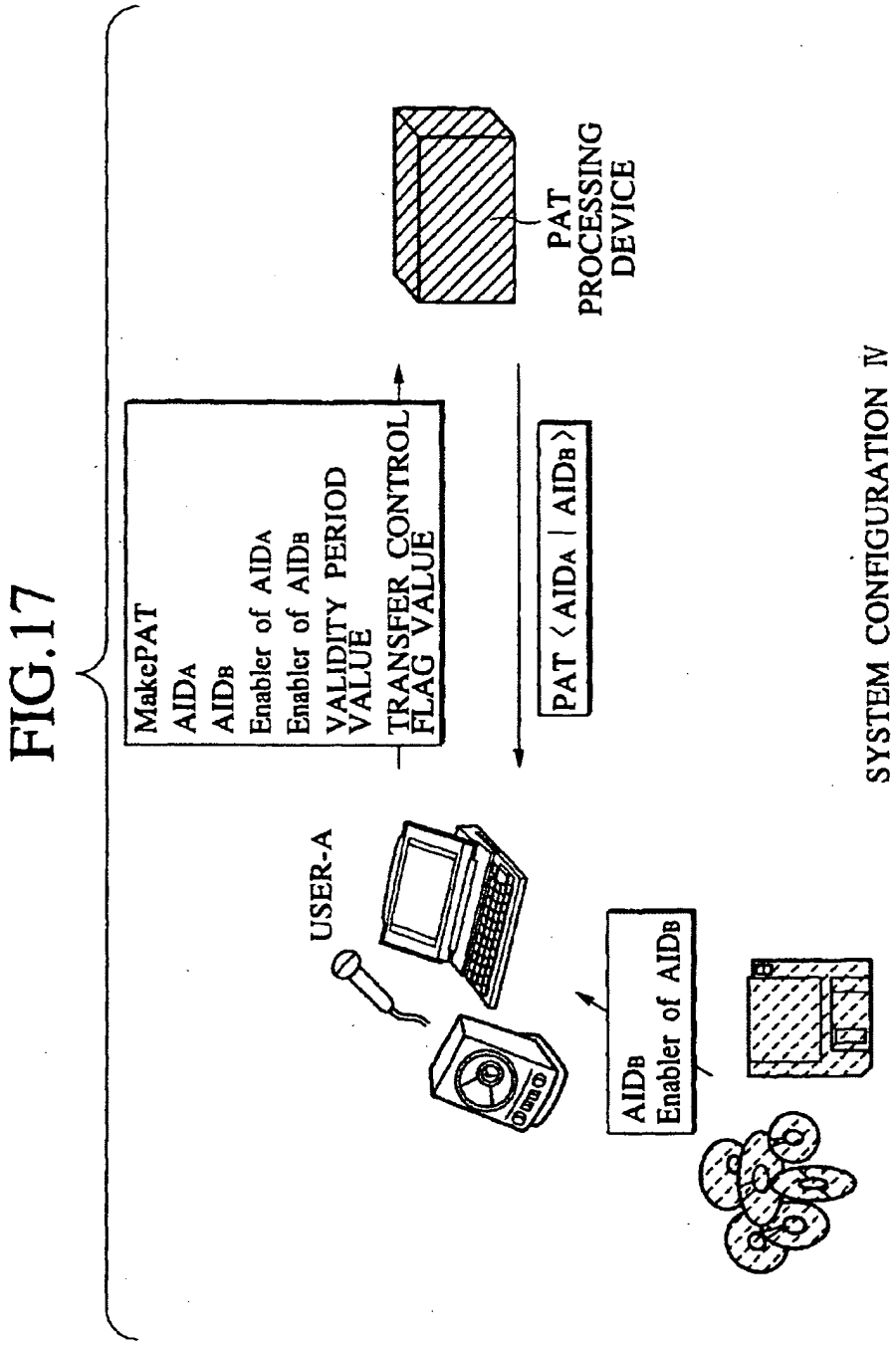


FIG.18

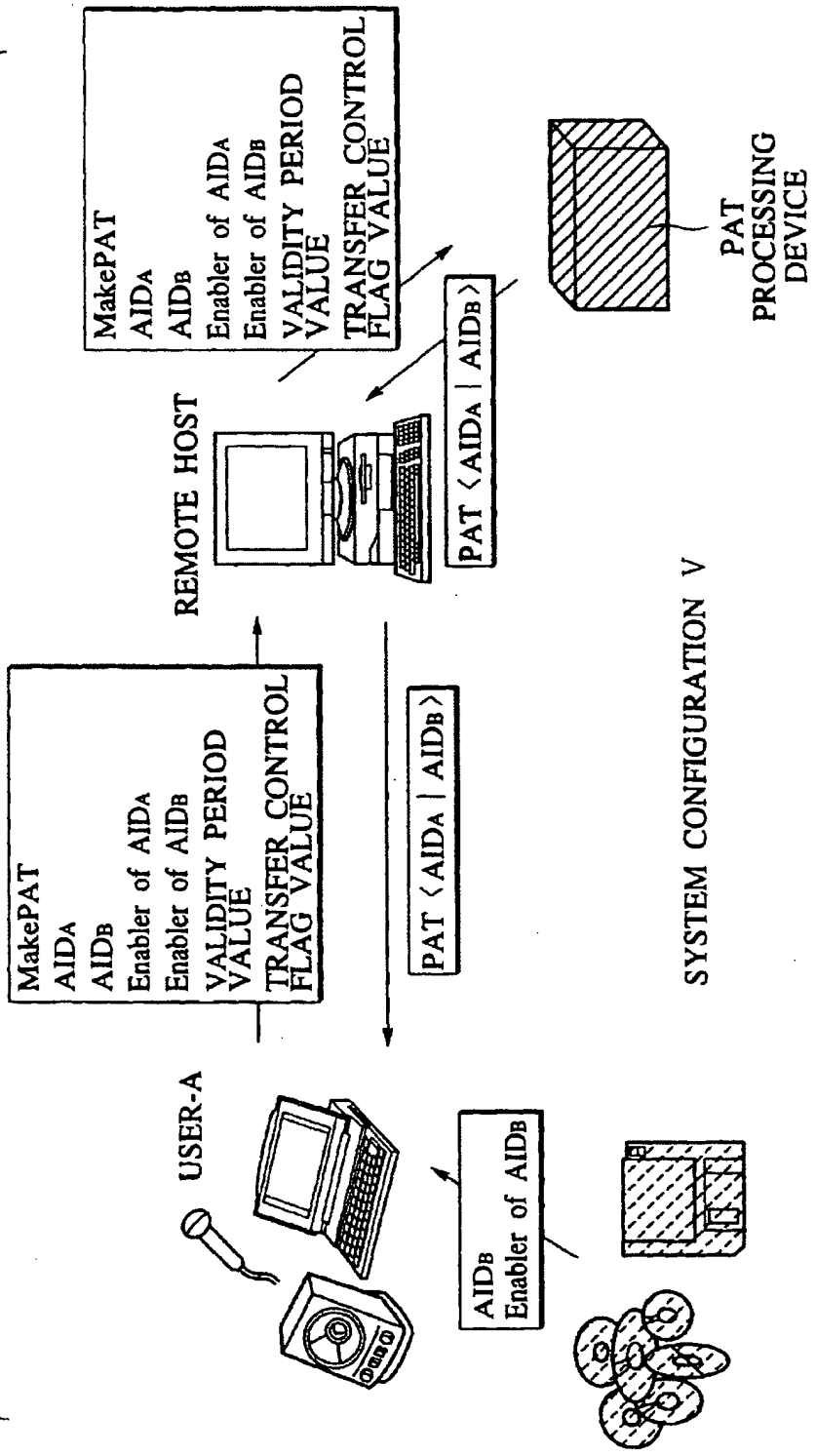


FIG.19

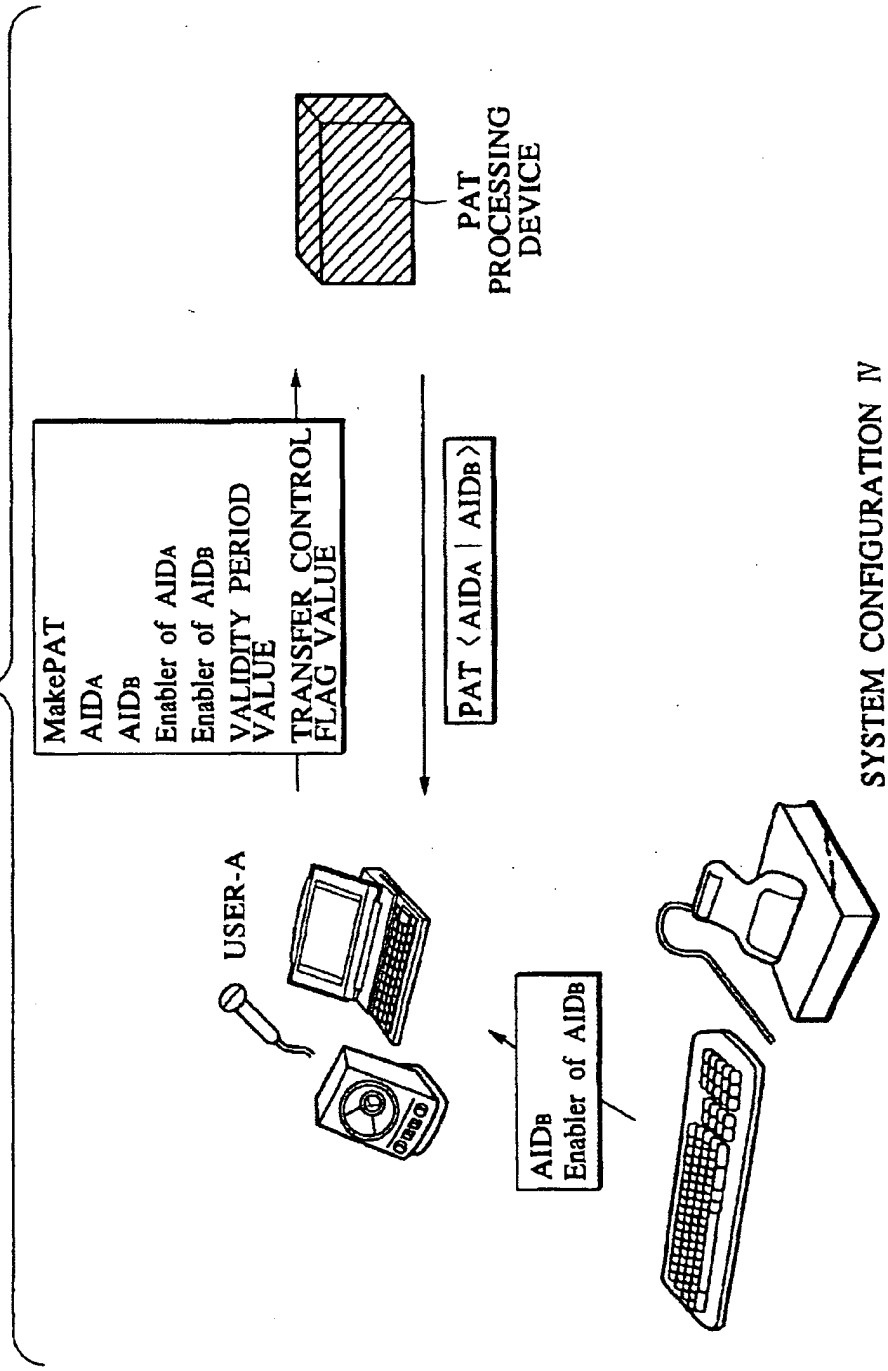


FIG.20

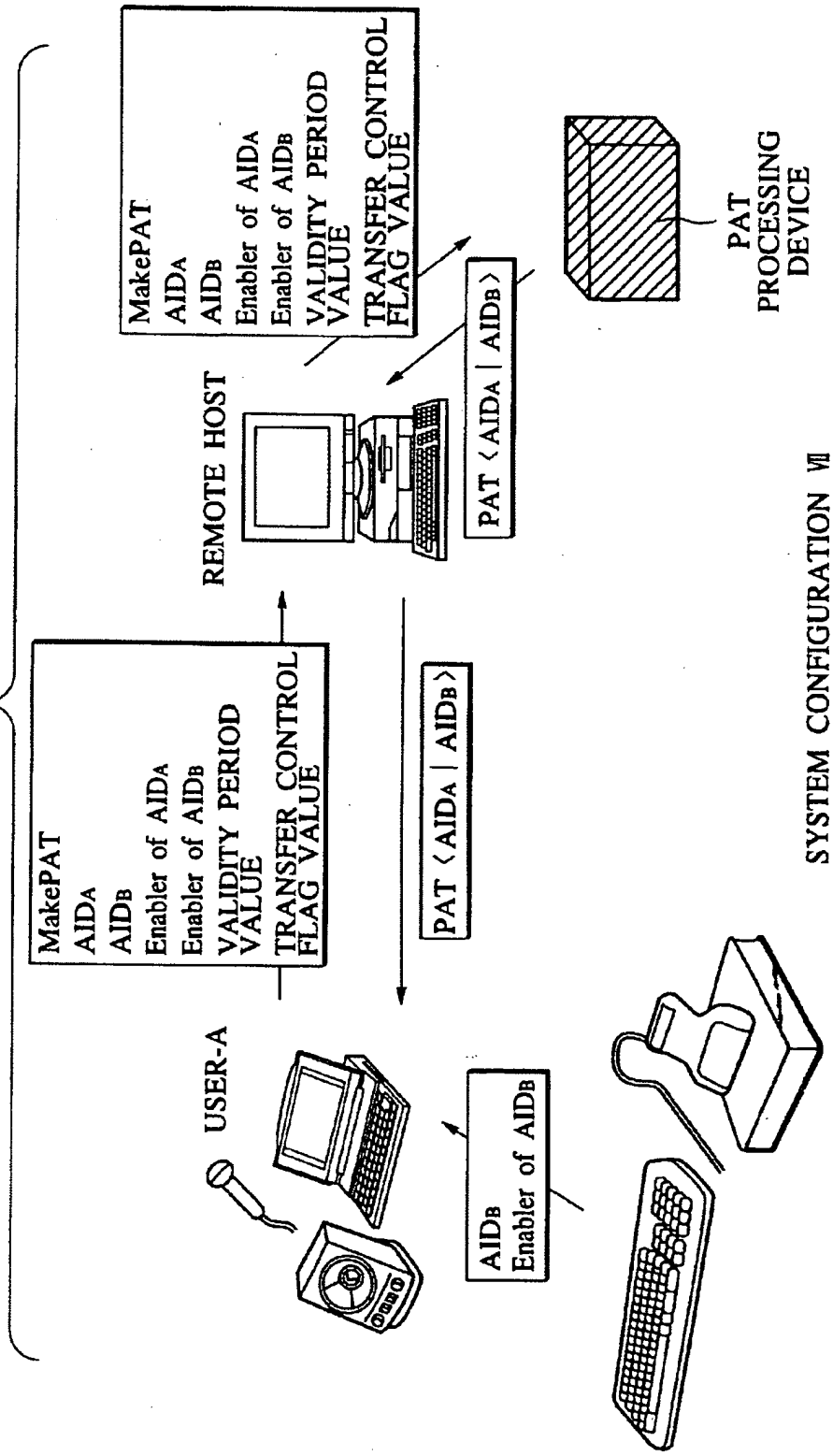


FIG.21

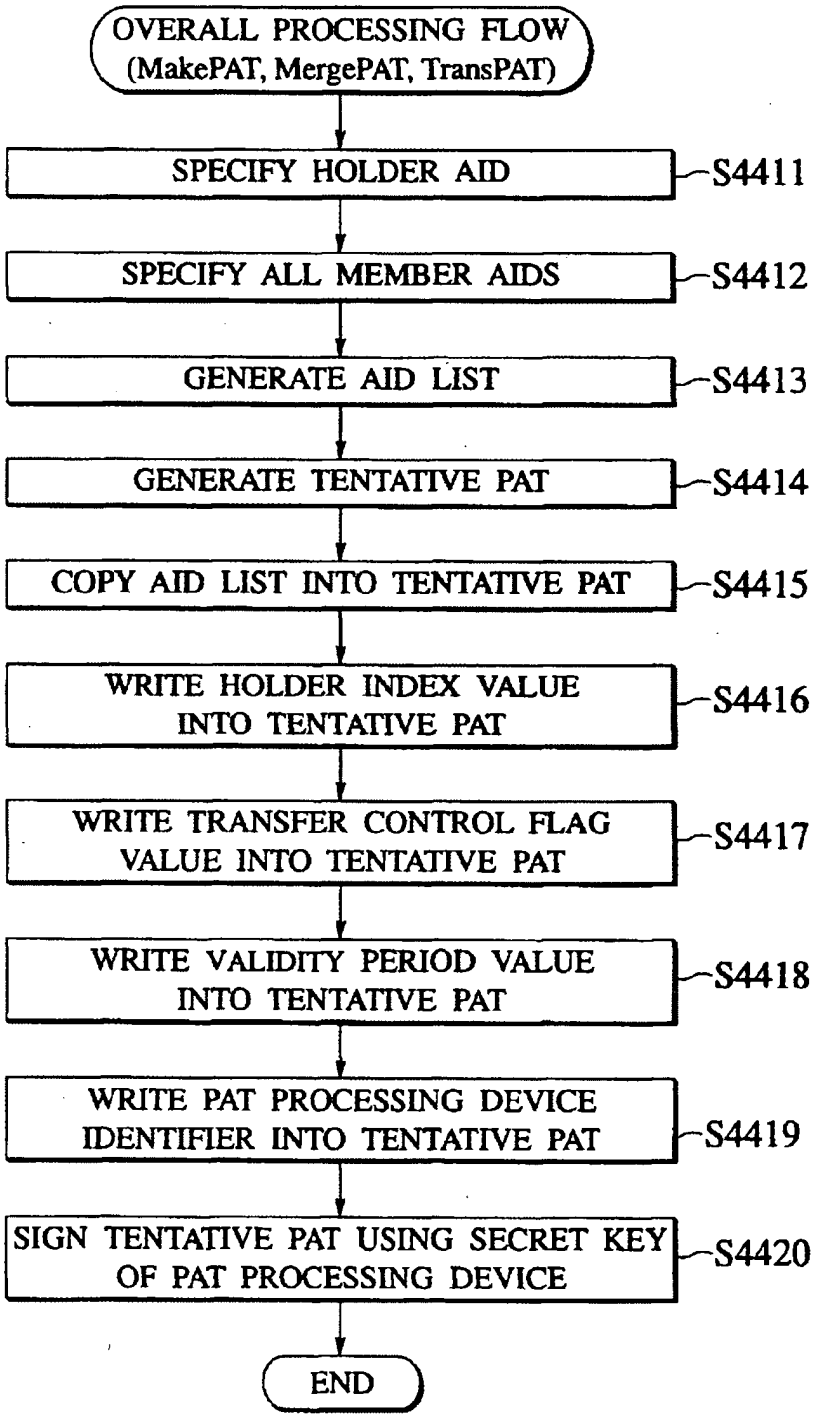


FIG.22

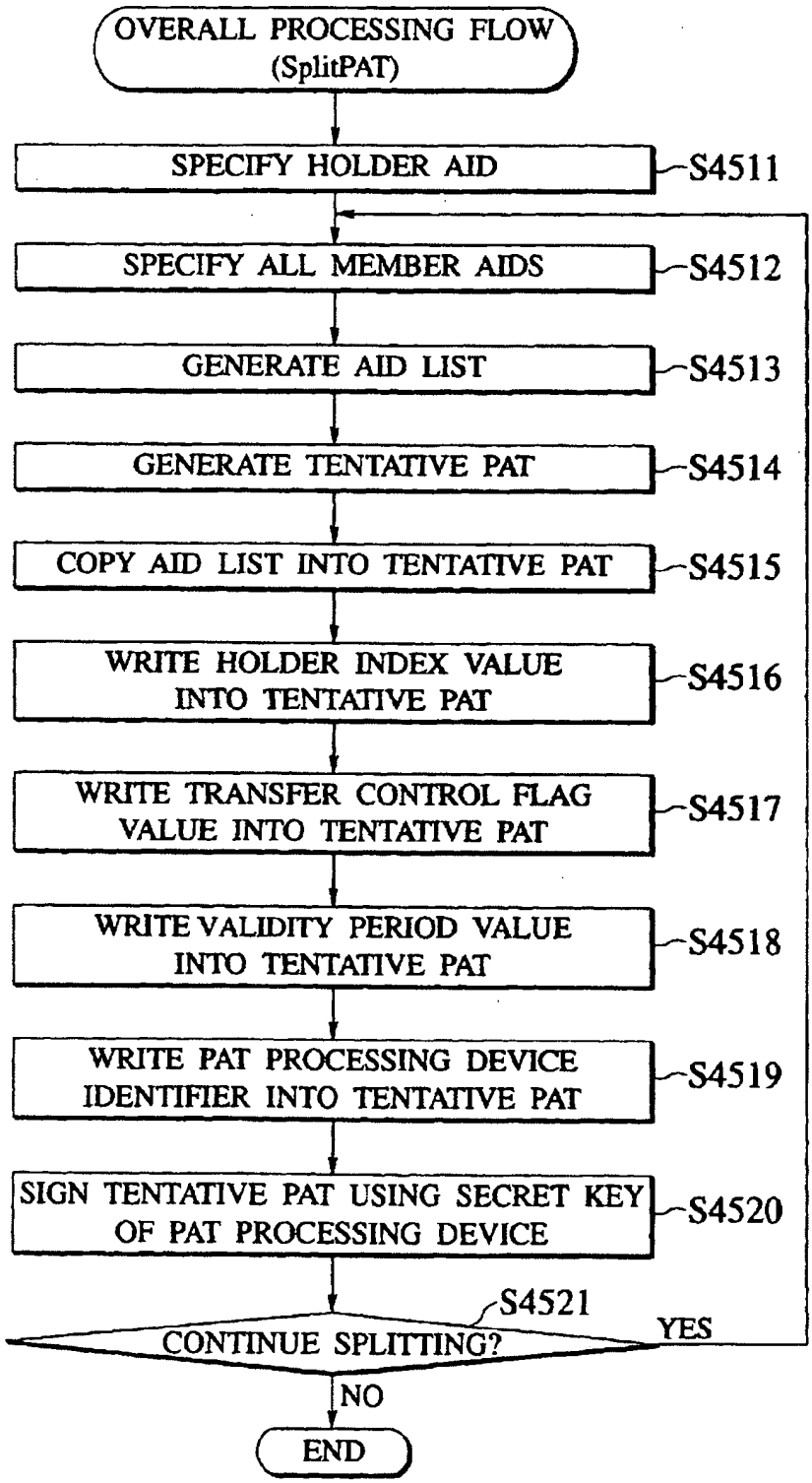


FIG.23

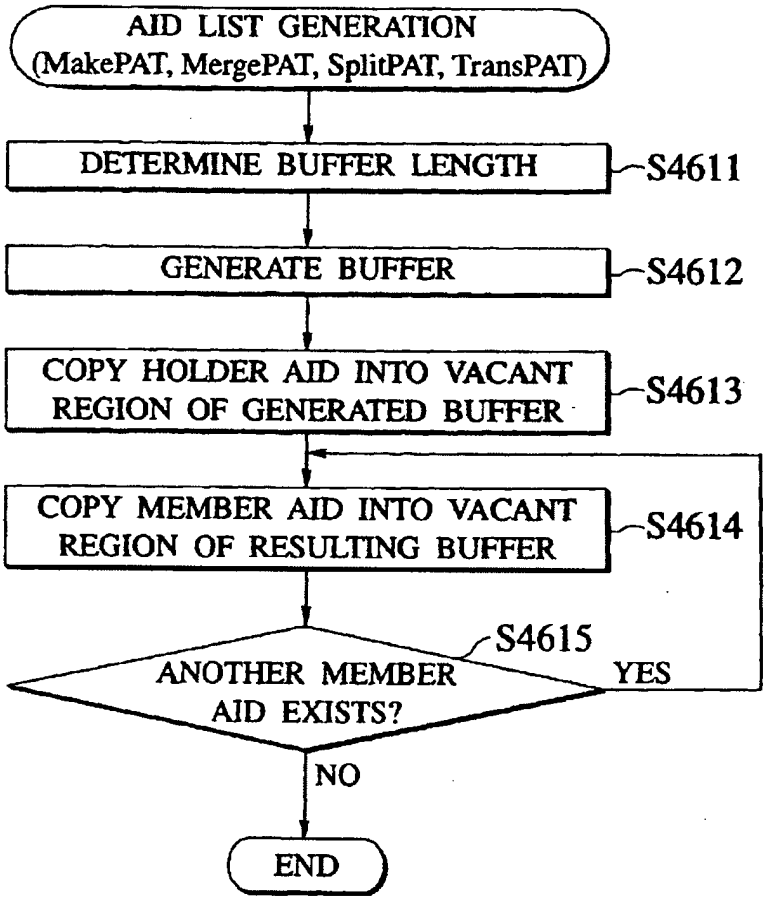


FIG.24

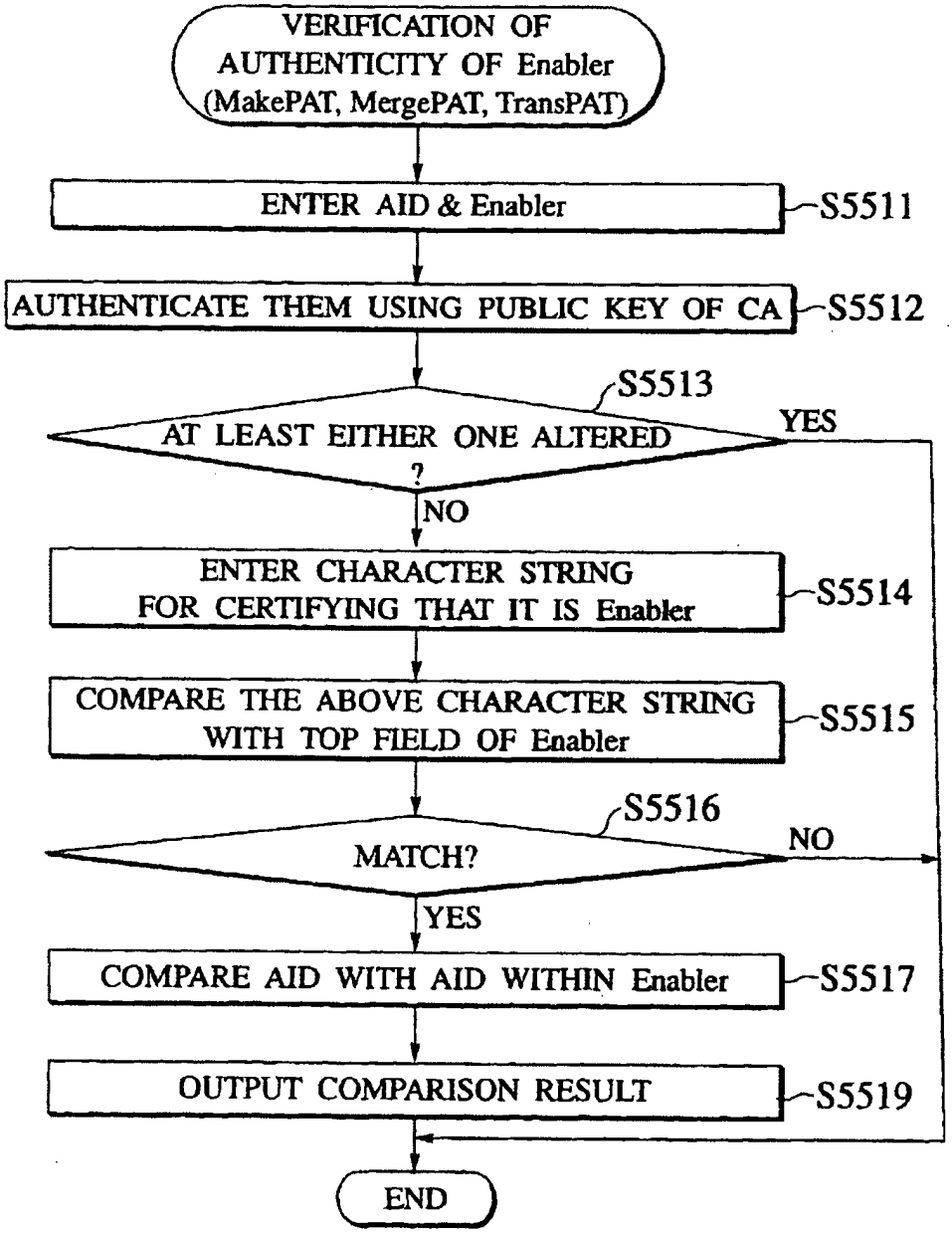
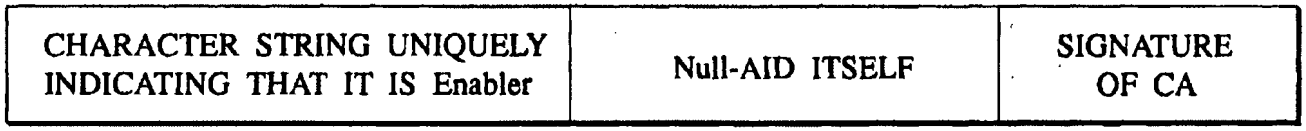


FIG.25

DATA STRUCTURE OF Null-AID

FIG.26

DATA STRUCTURE OF Enabler of Null-AID



EP 0 946 022 A2

FIG.27

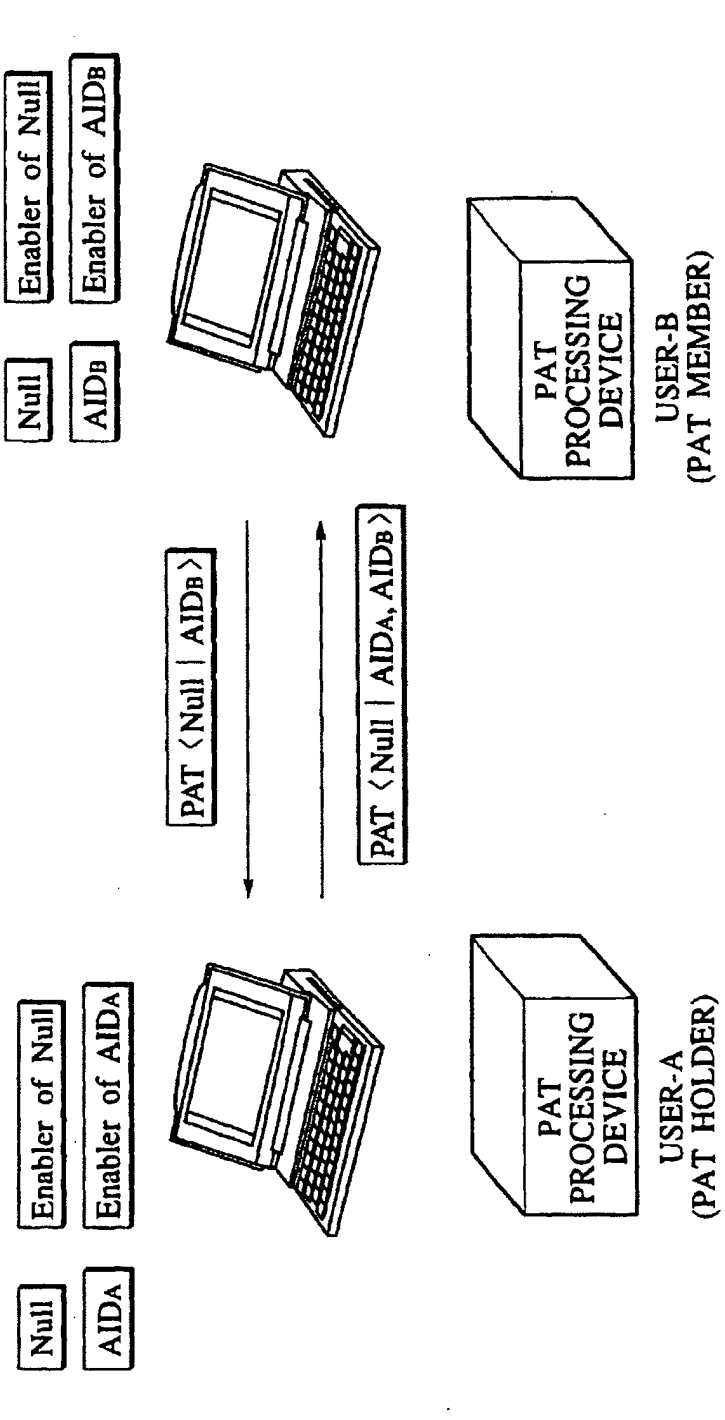


FIG.28

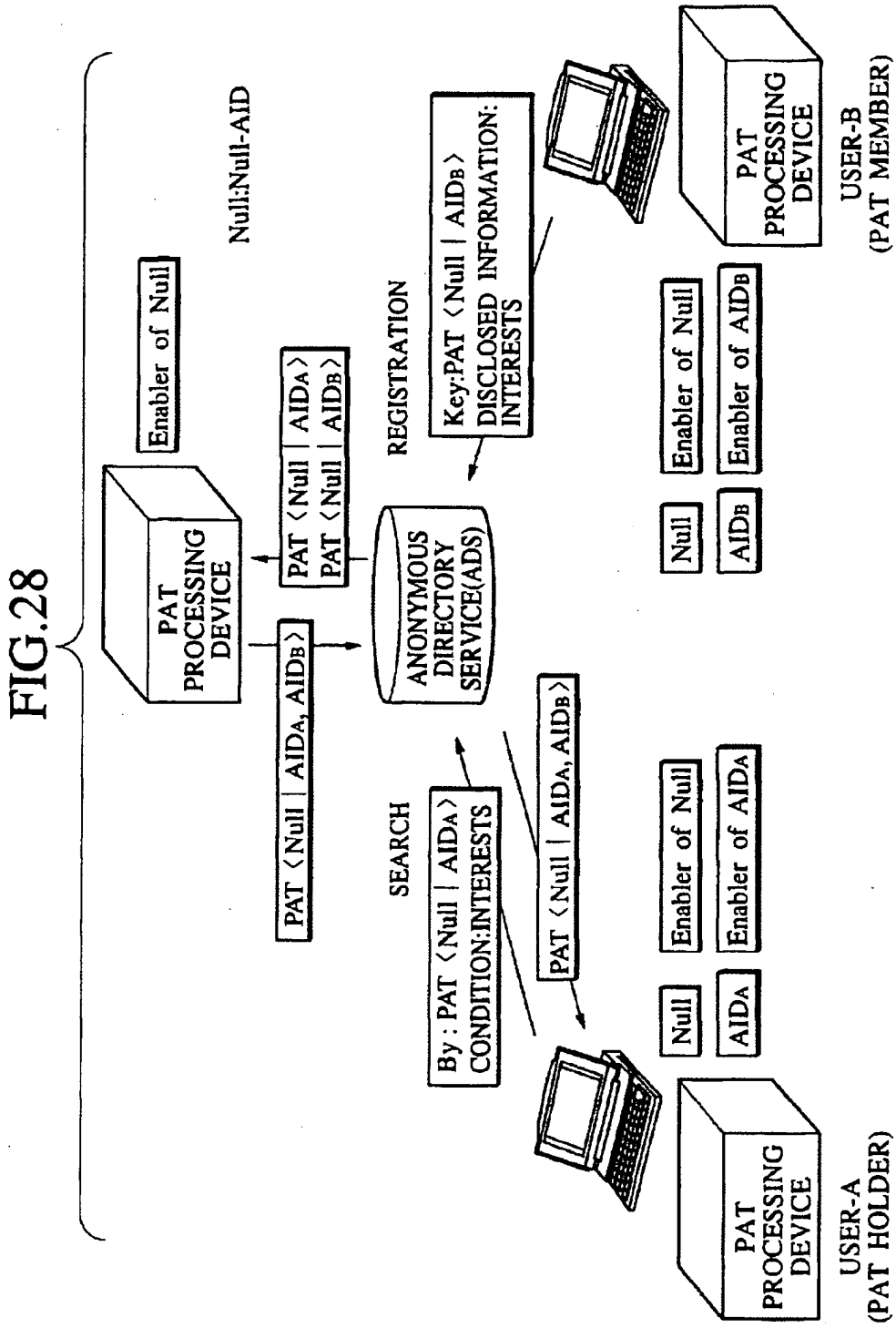
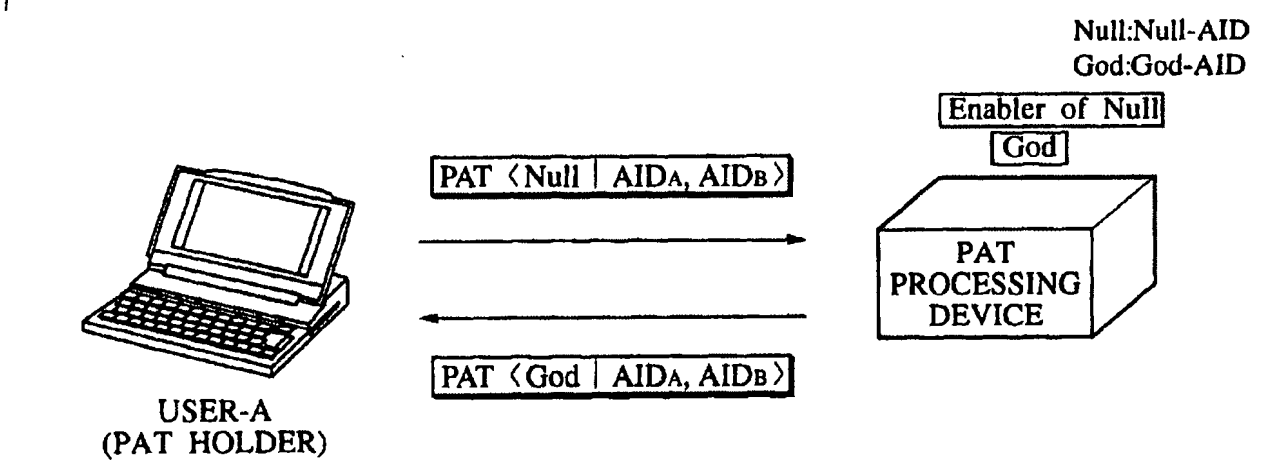


FIG.29

DATA STRUCTURE OF God-AID

CHARACTER STRING UNIQUELY INDICATING THAT IT IS God-AID	SIGNATURE OF CA

FIG.30



EP 0 946 022 A2

FIG.31

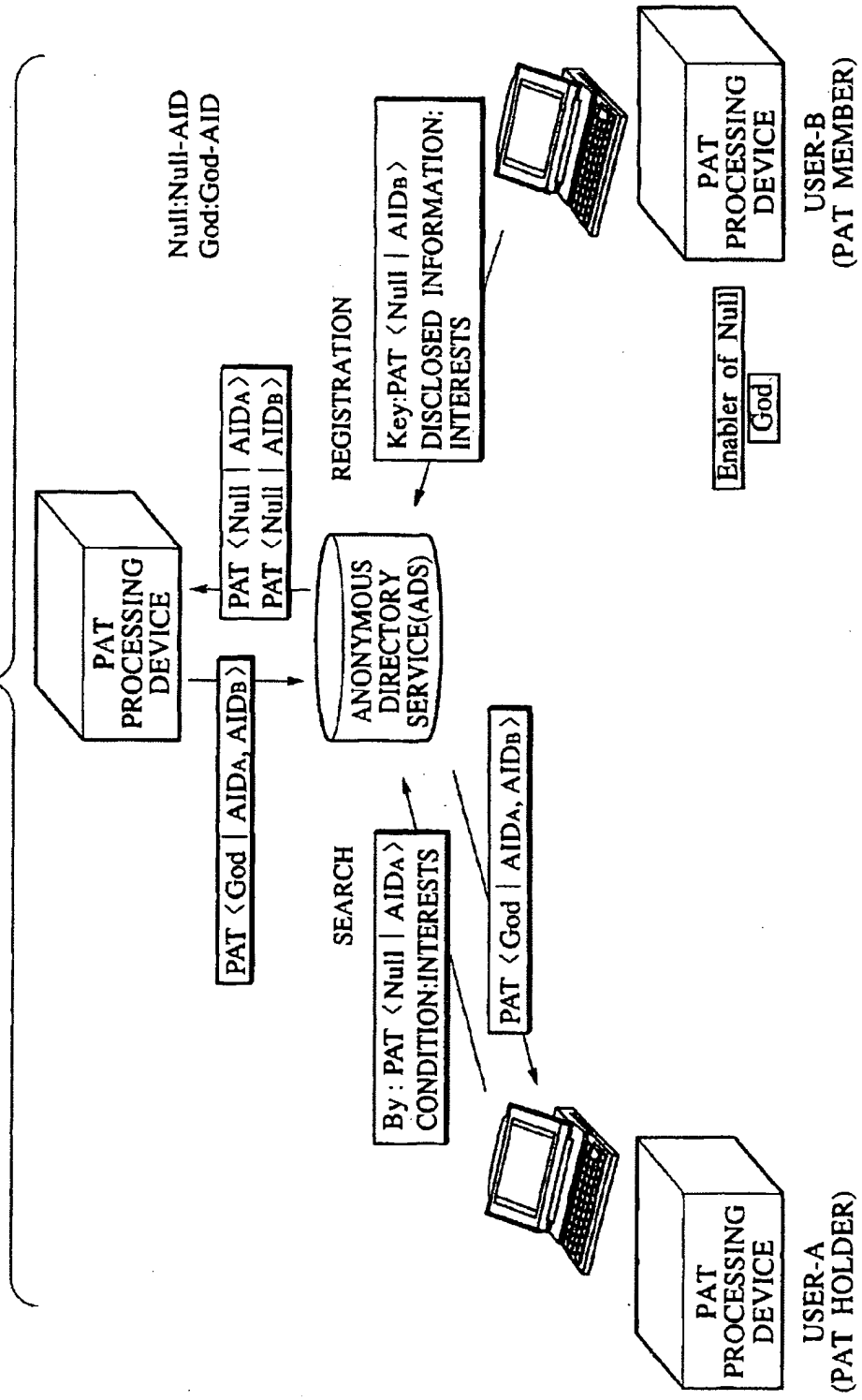


FIG.32

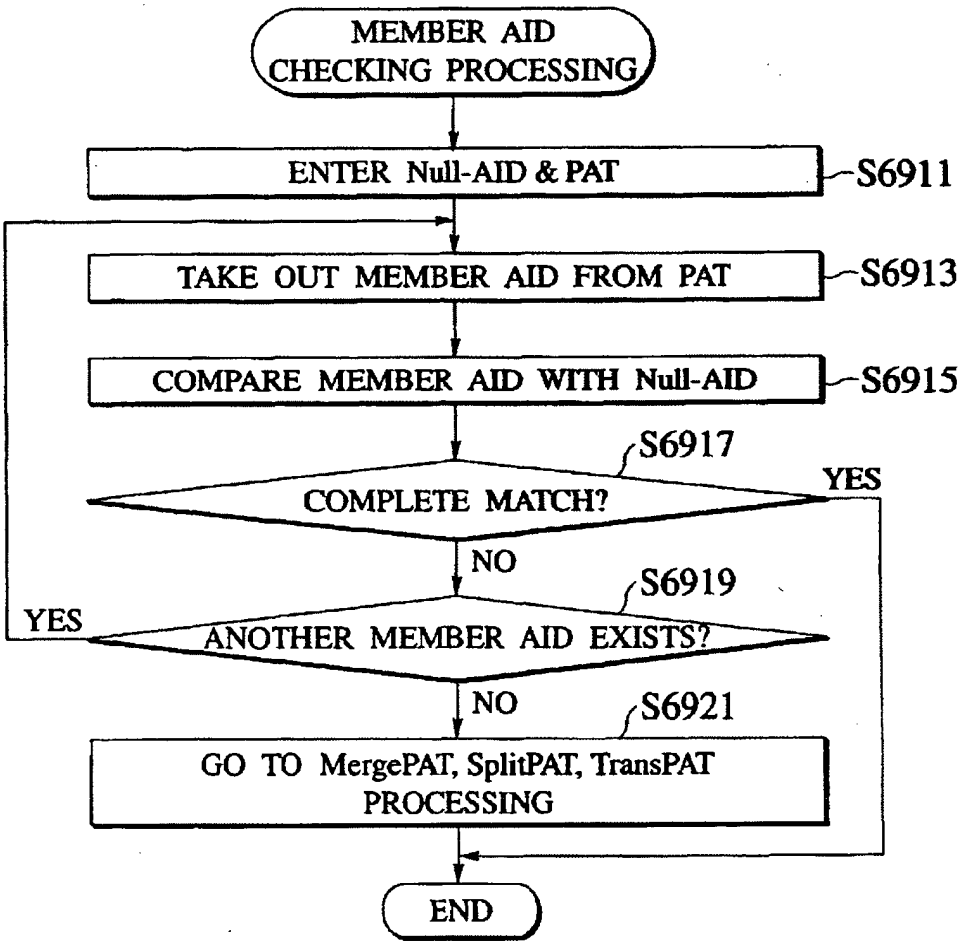


FIG.33

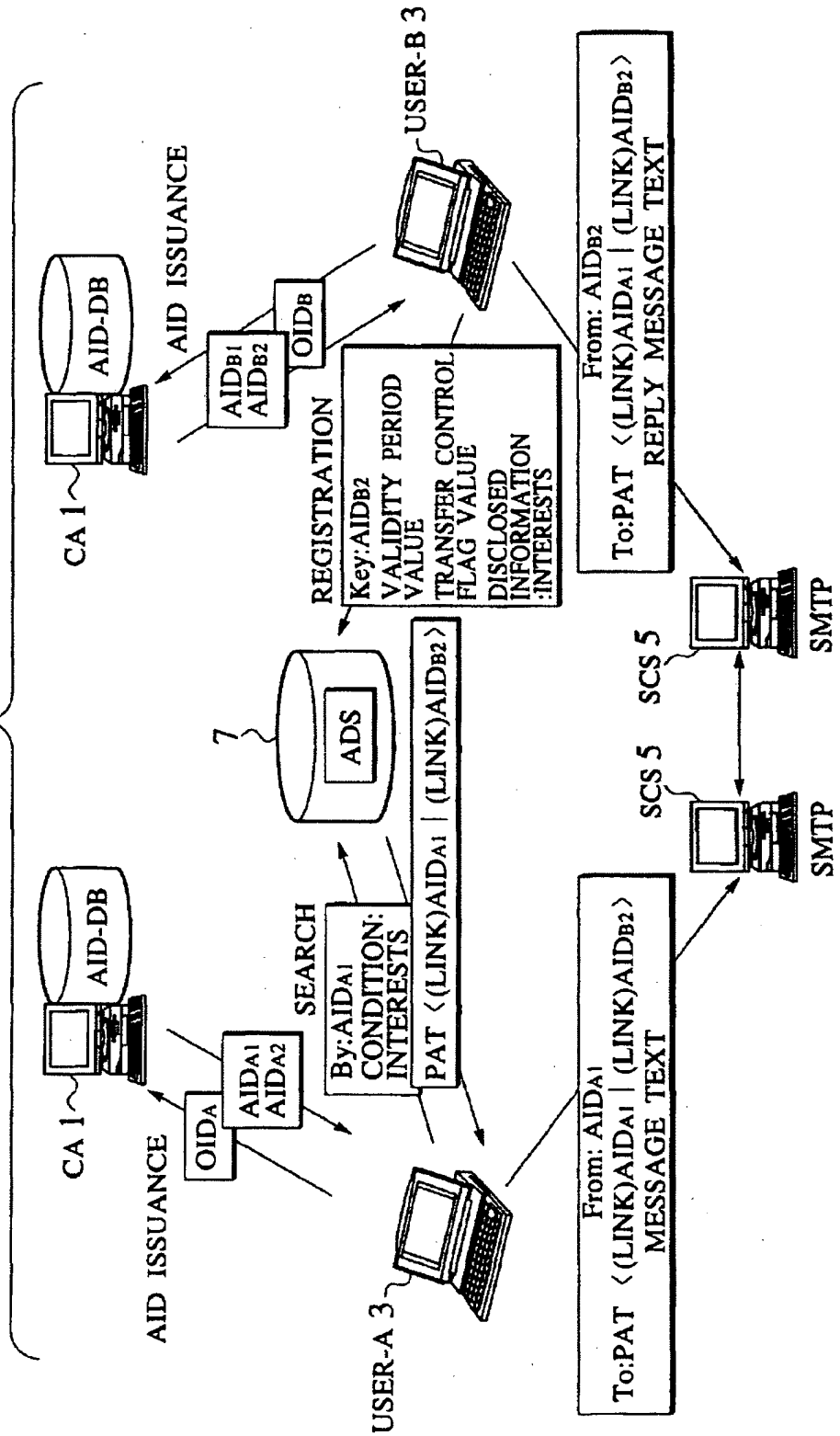


FIG.34

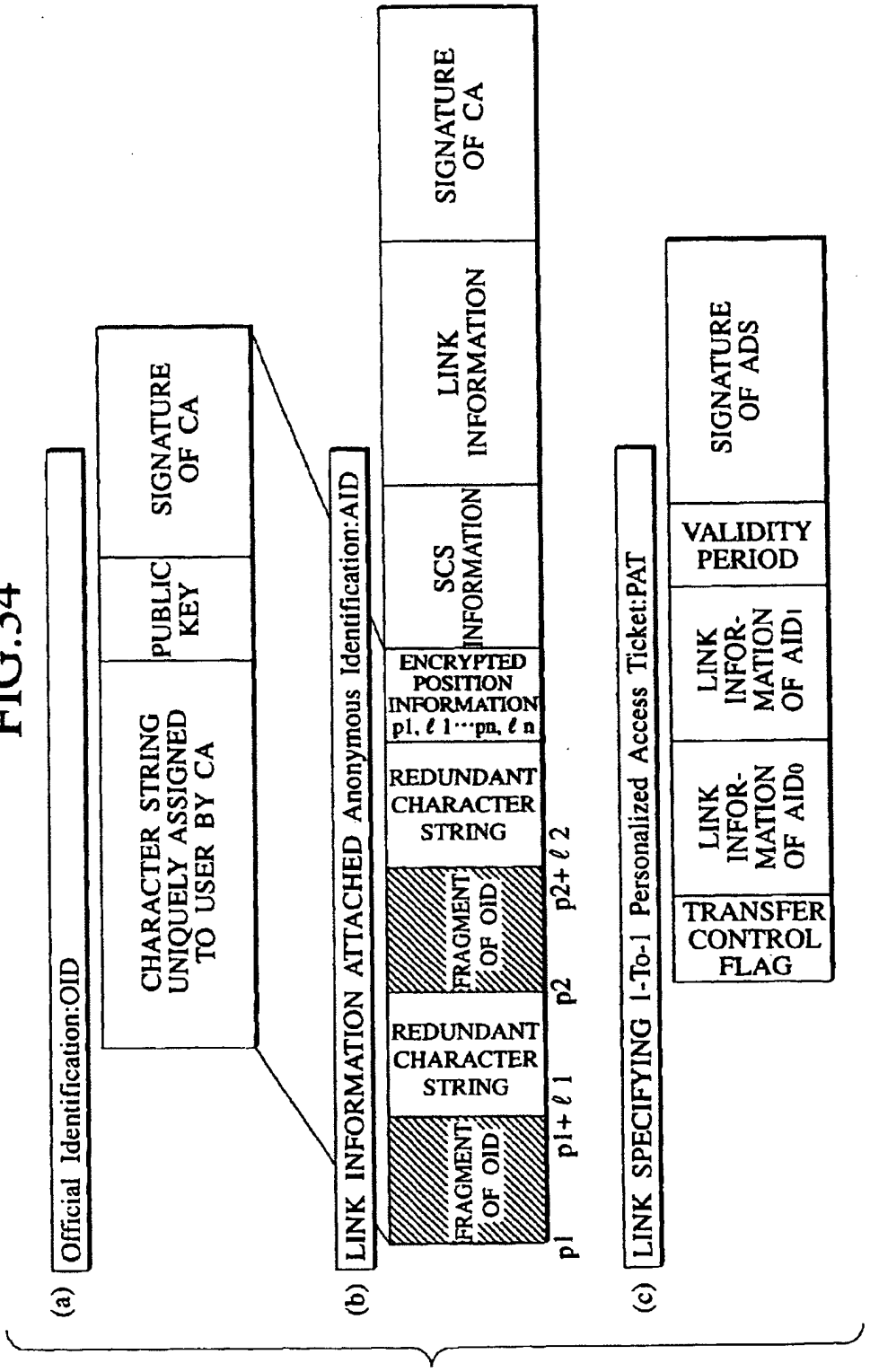


FIG.35

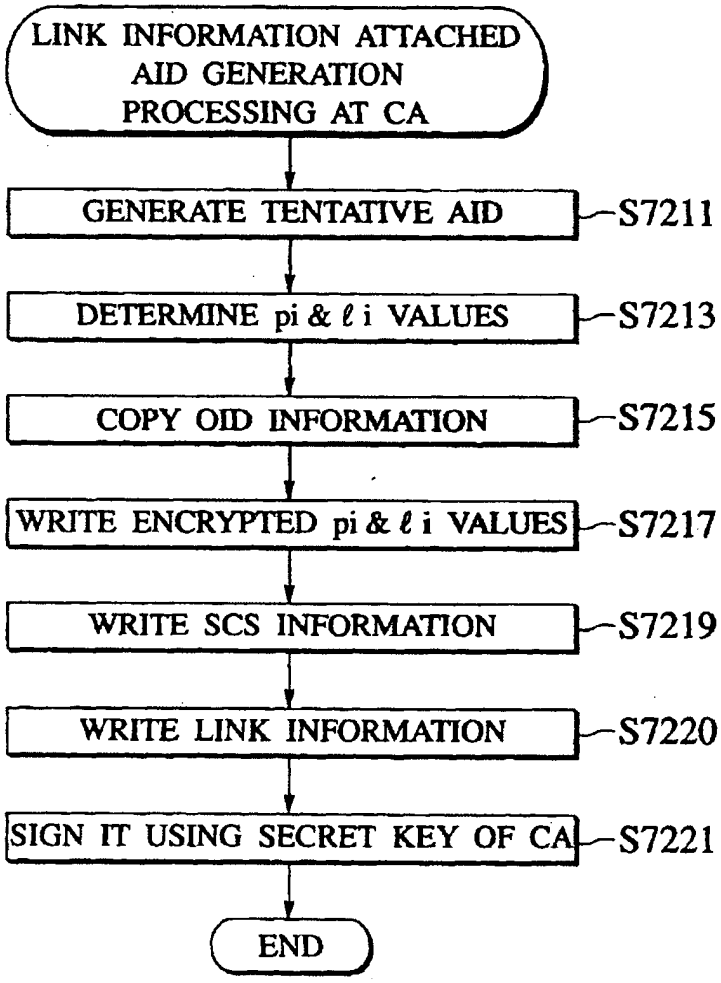


FIG.36

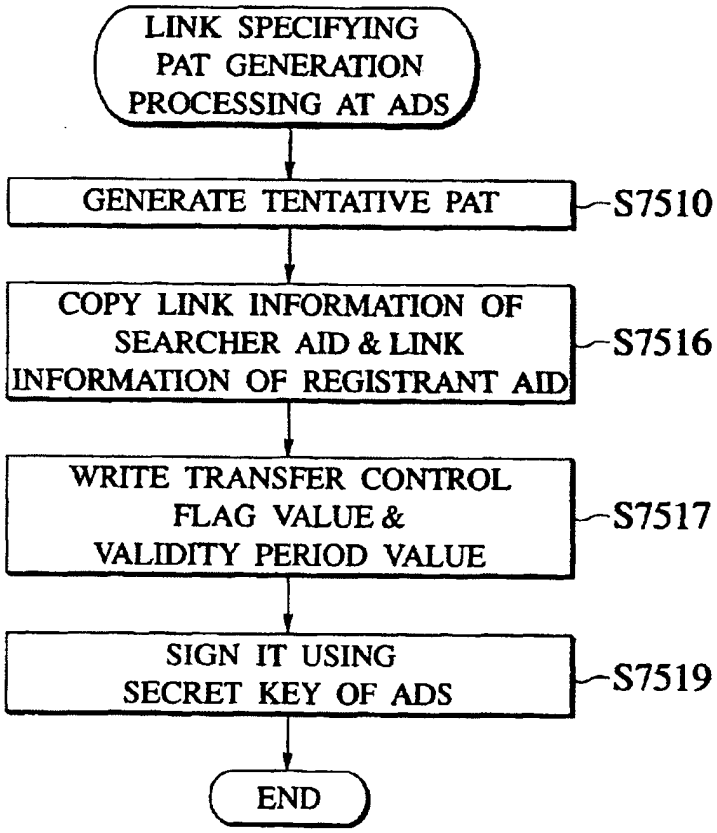


FIG.37

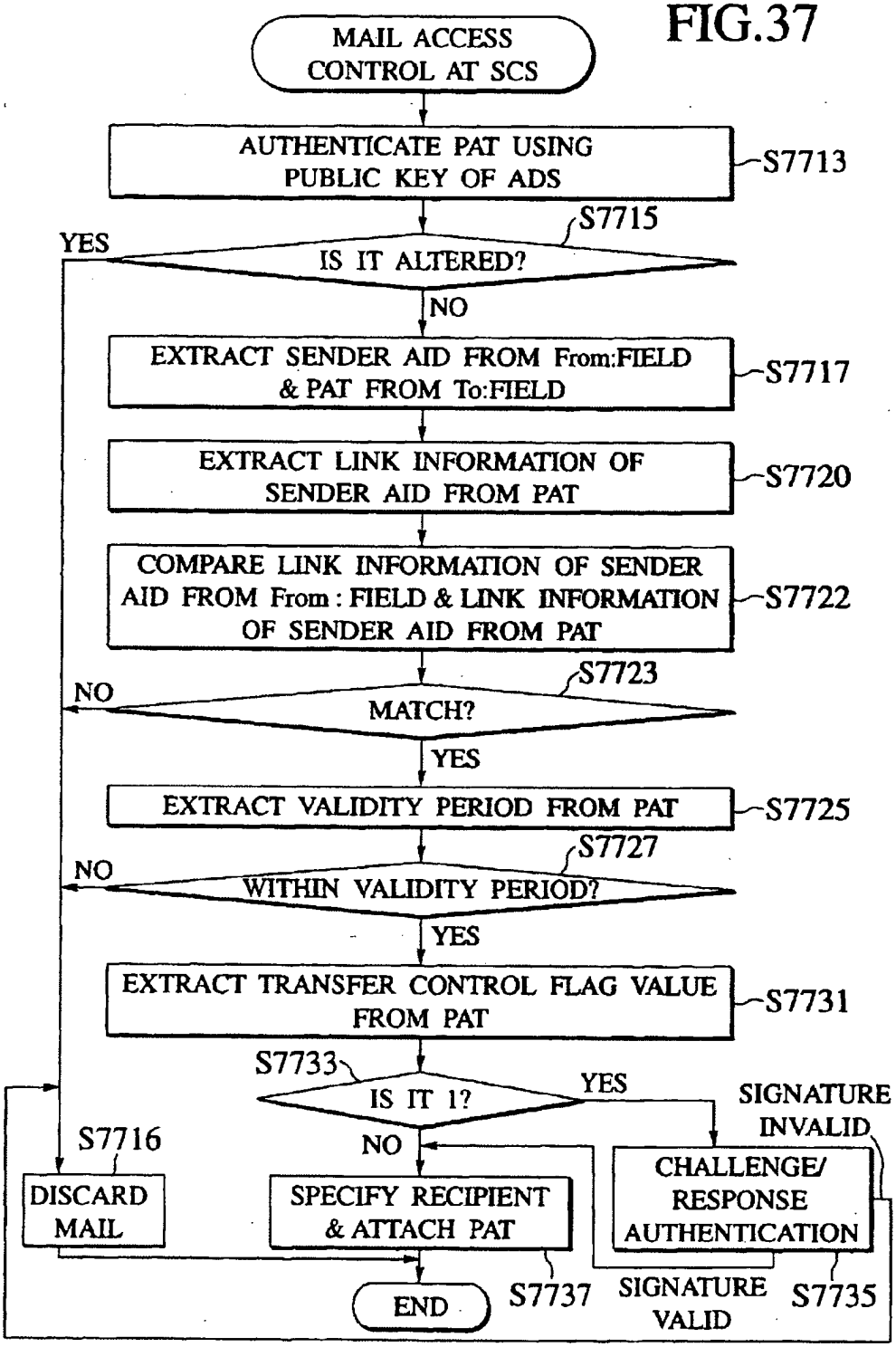


FIG.38

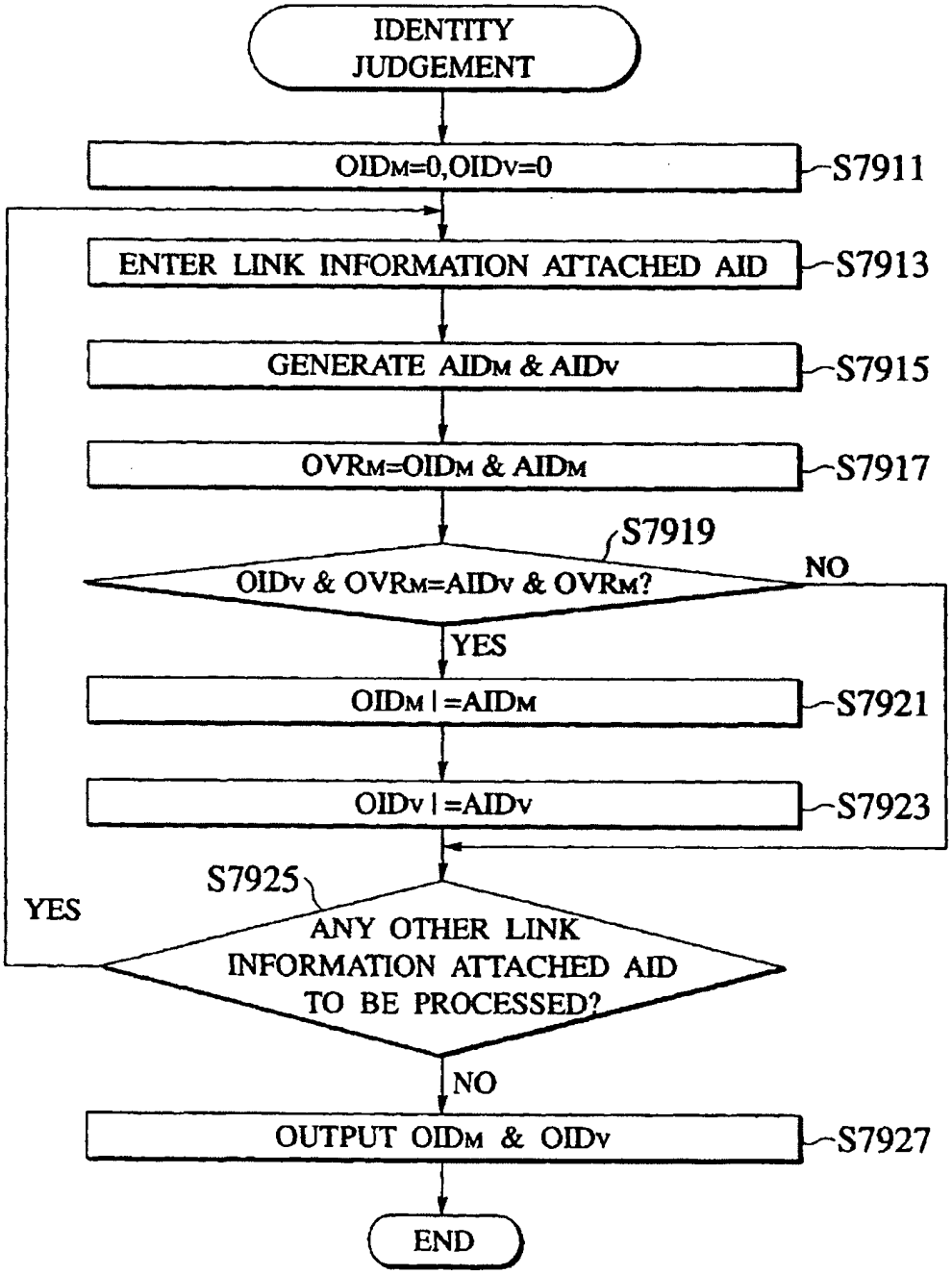


FIG.39

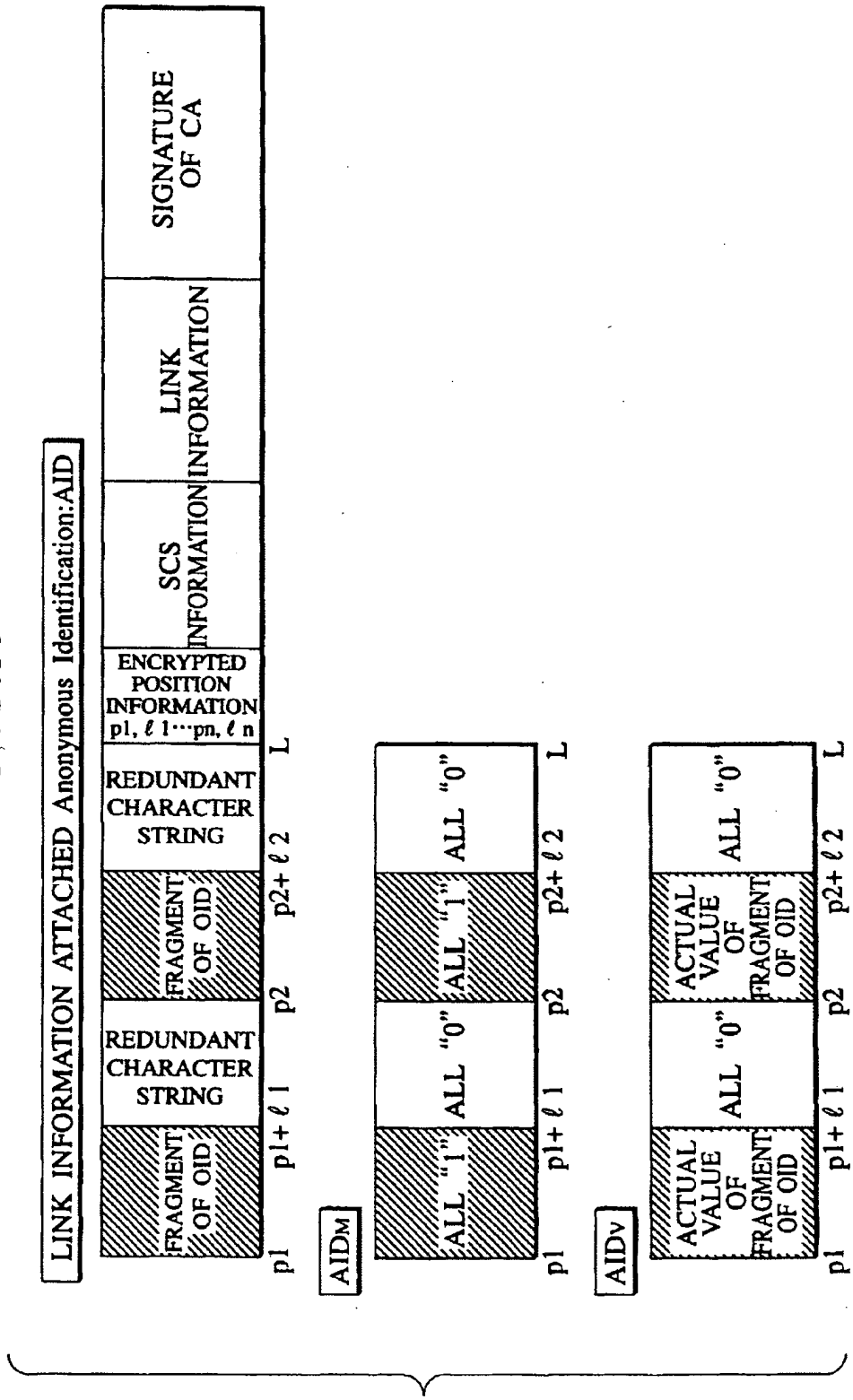


FIG.40

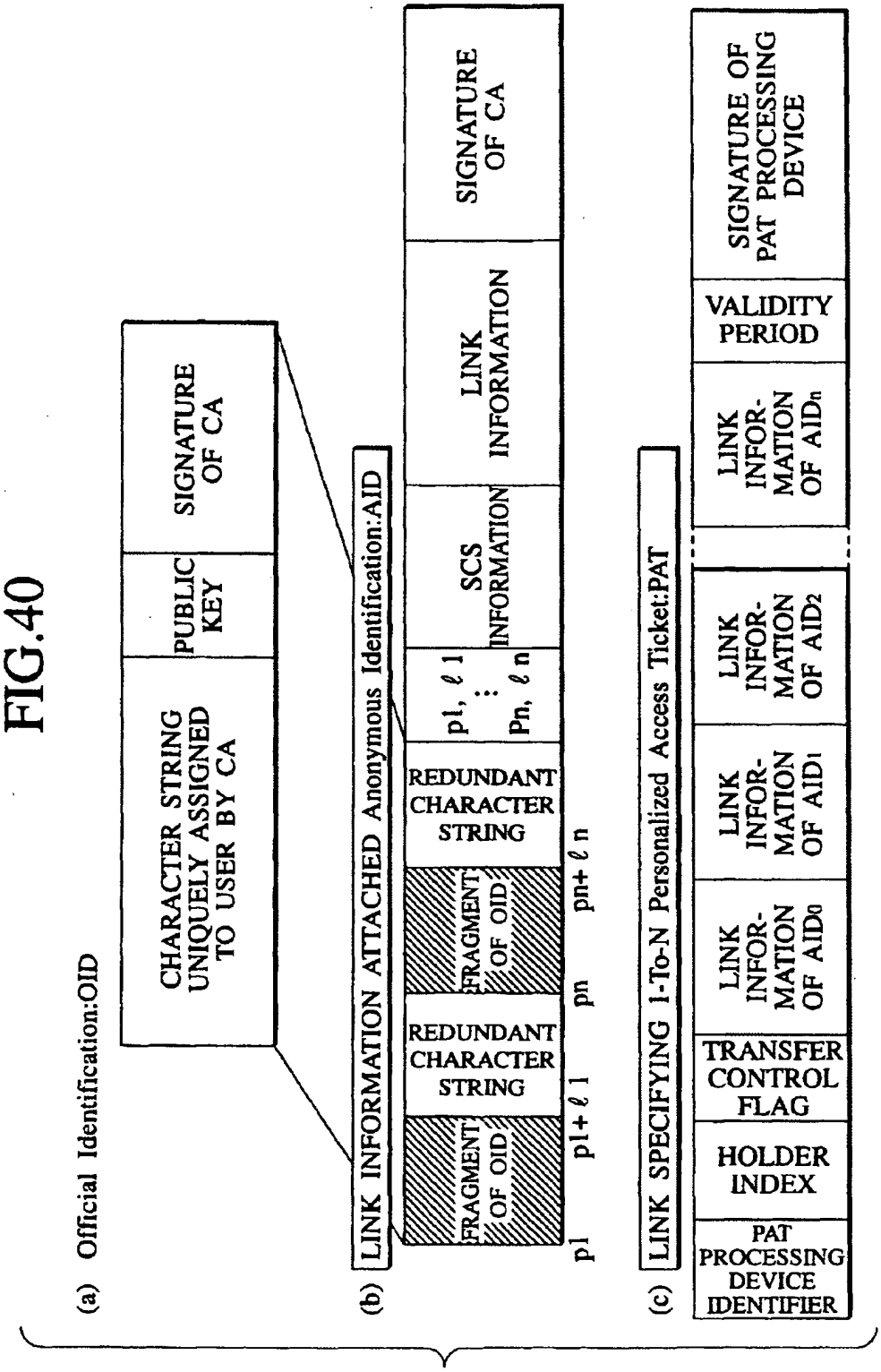


FIG.41

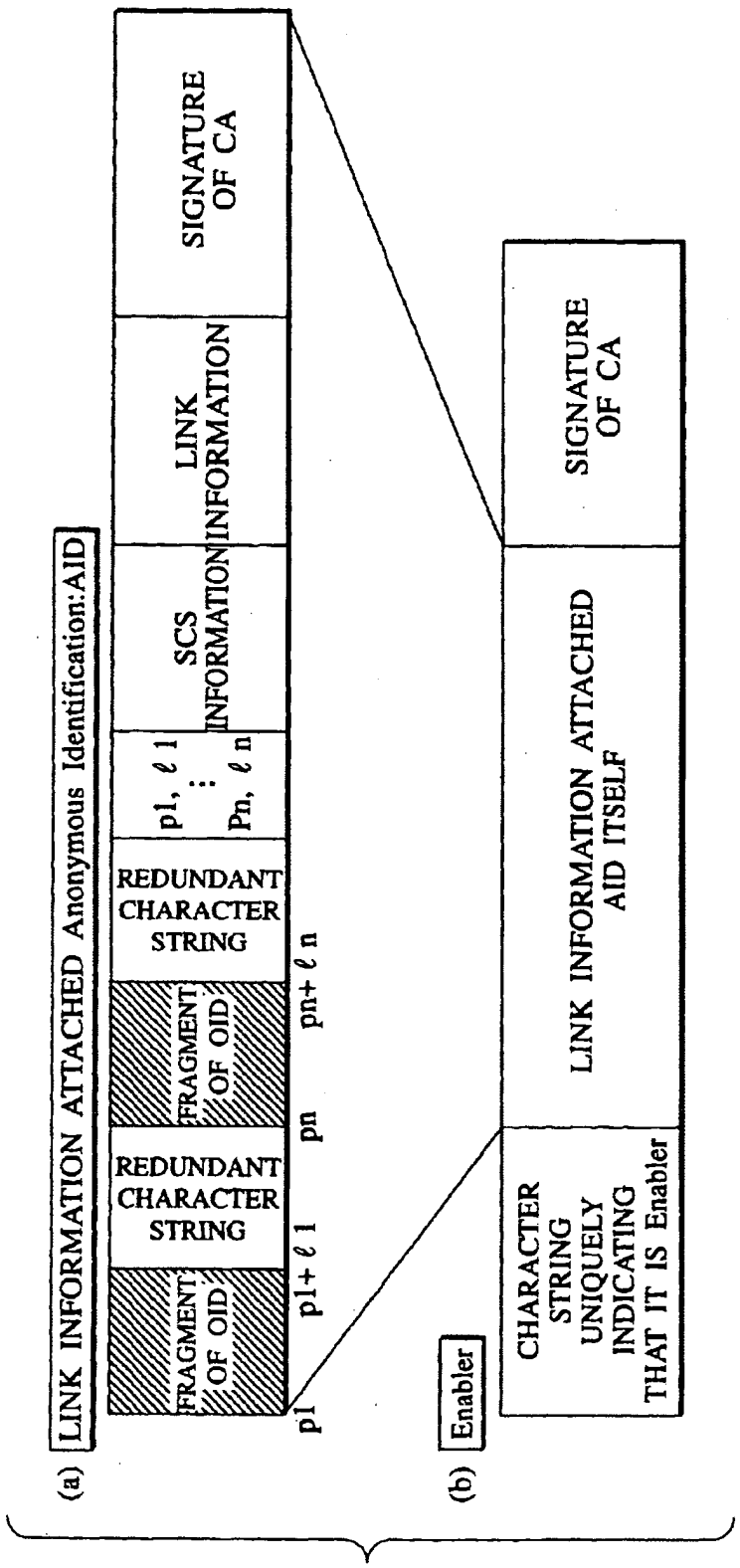


FIG.42

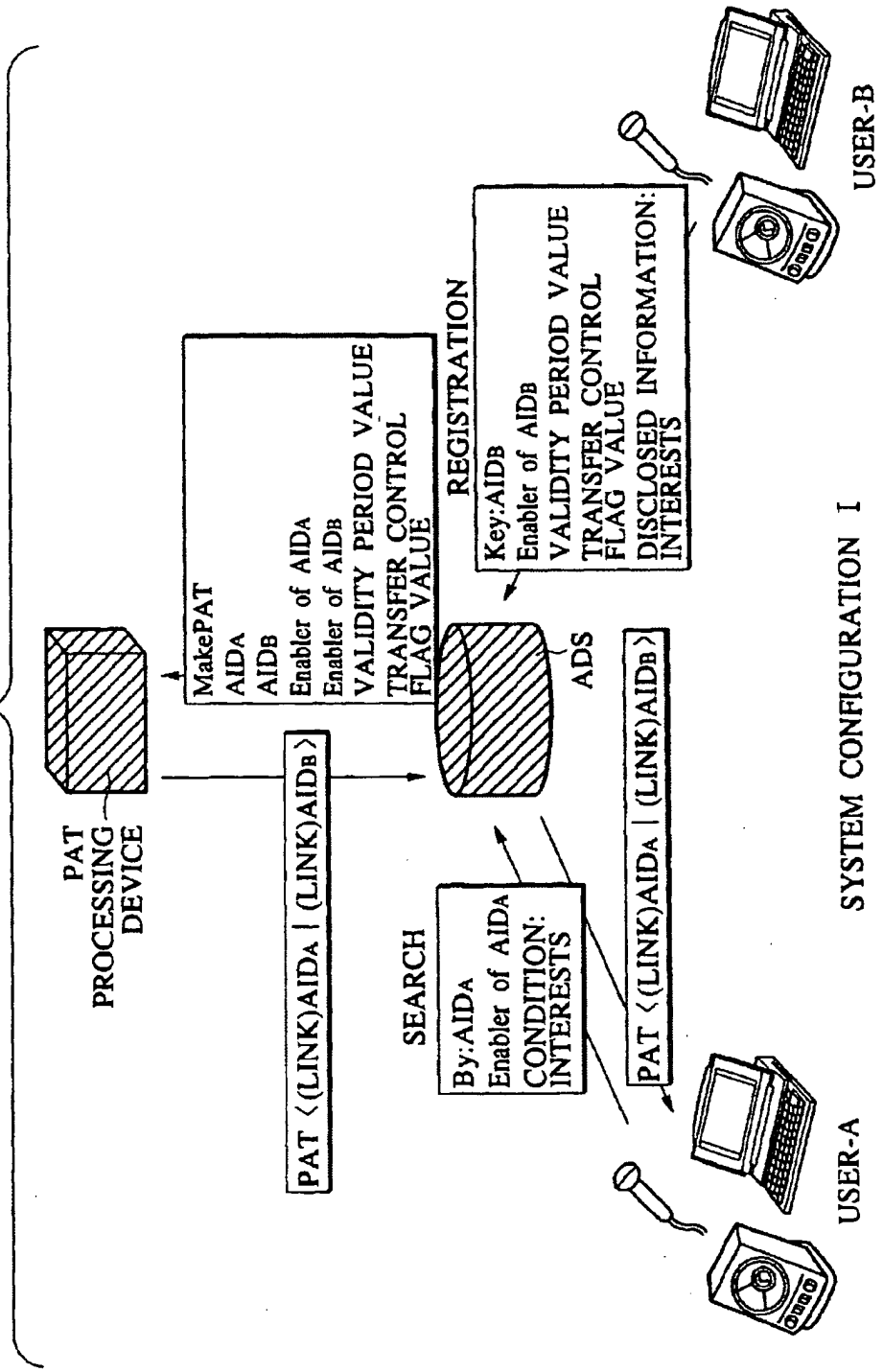


FIG.43

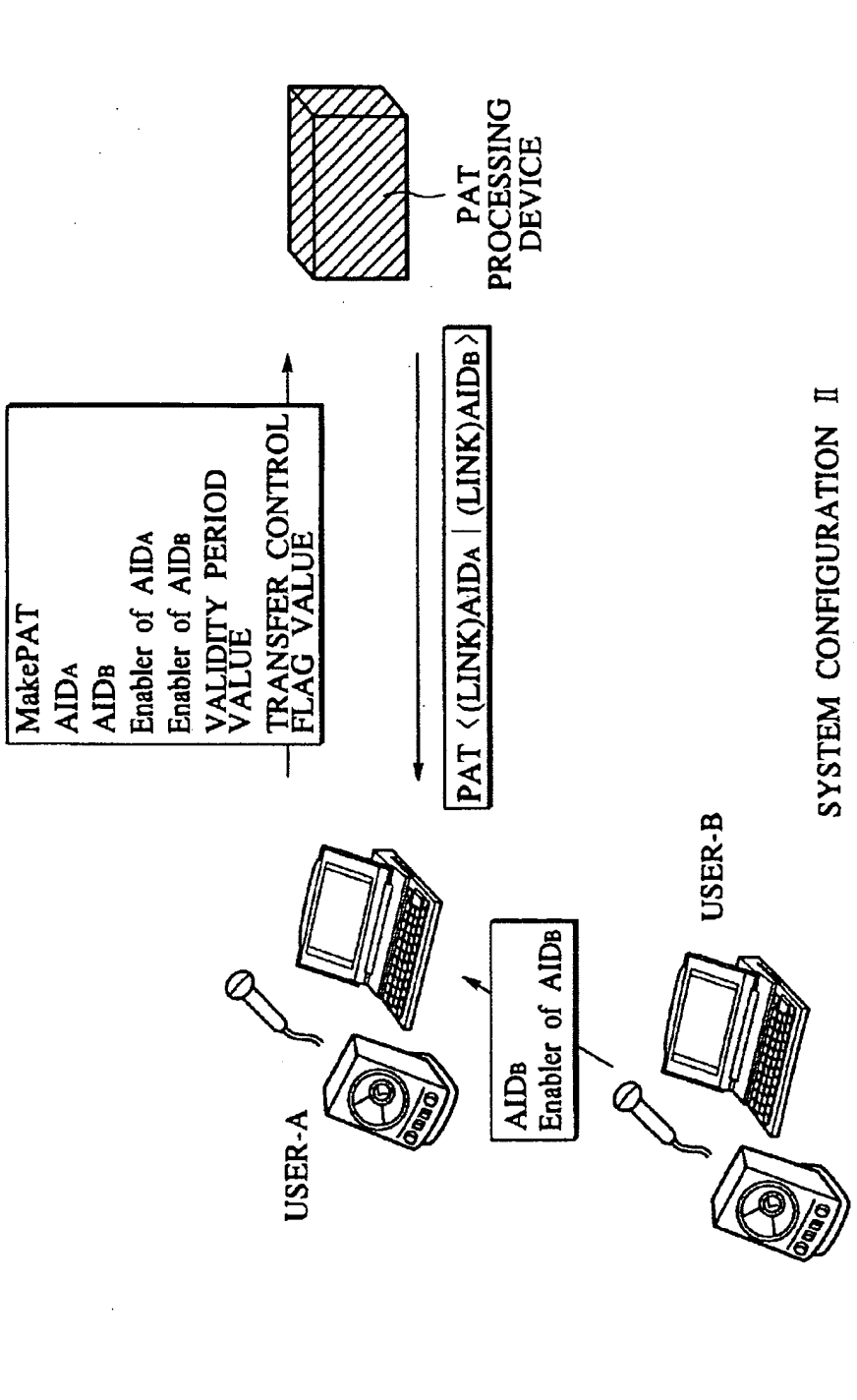


FIG.44

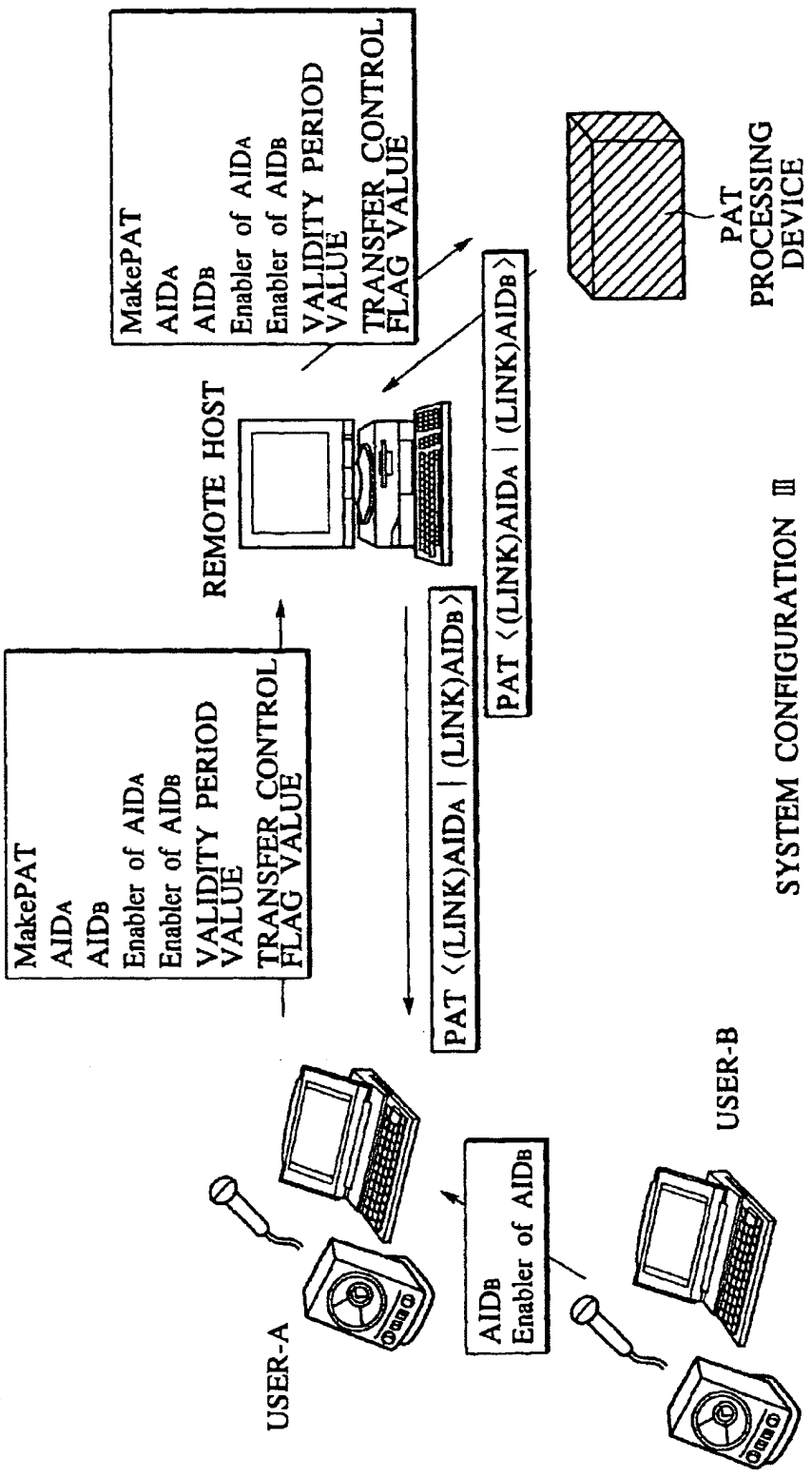


FIG.45

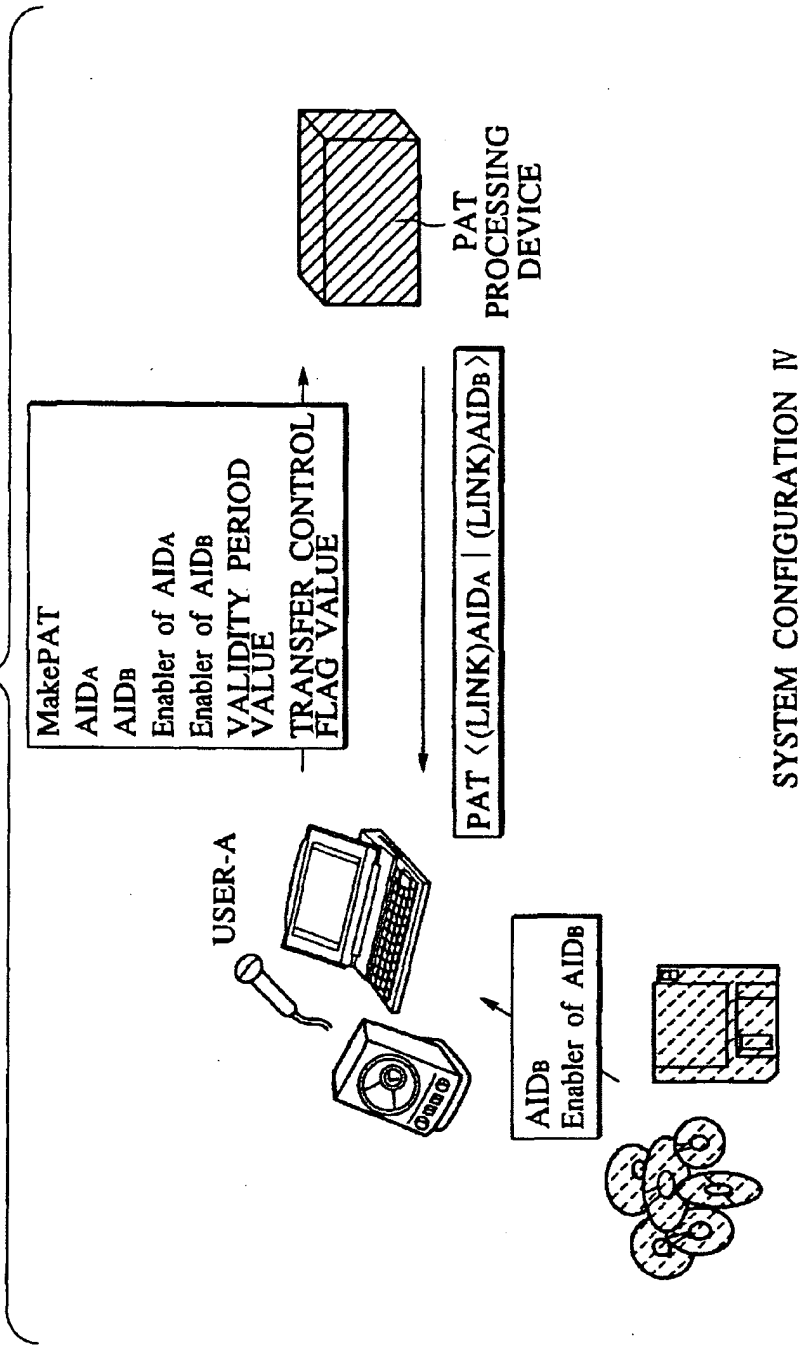


FIG.46

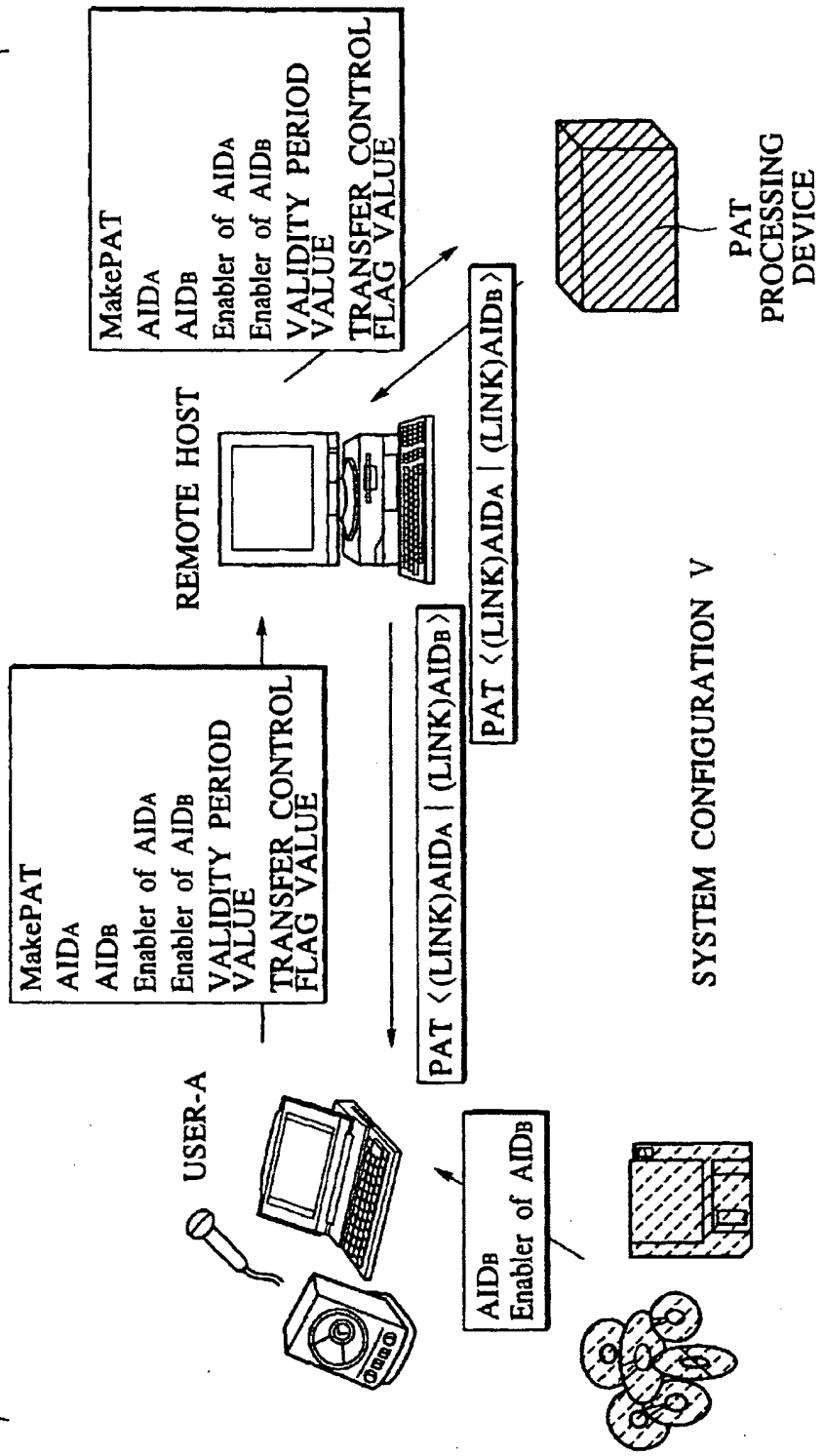


FIG.47

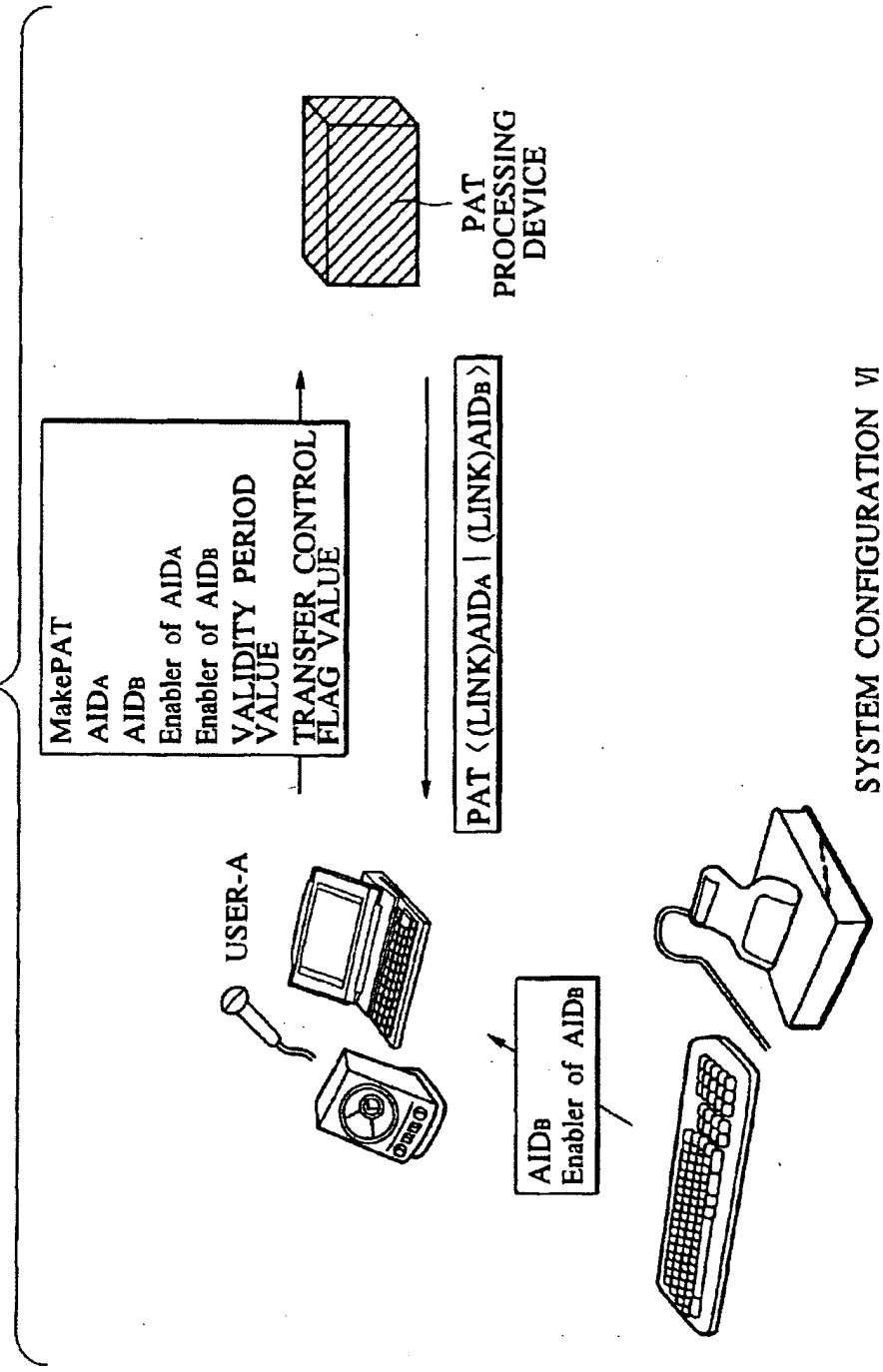


FIG.48

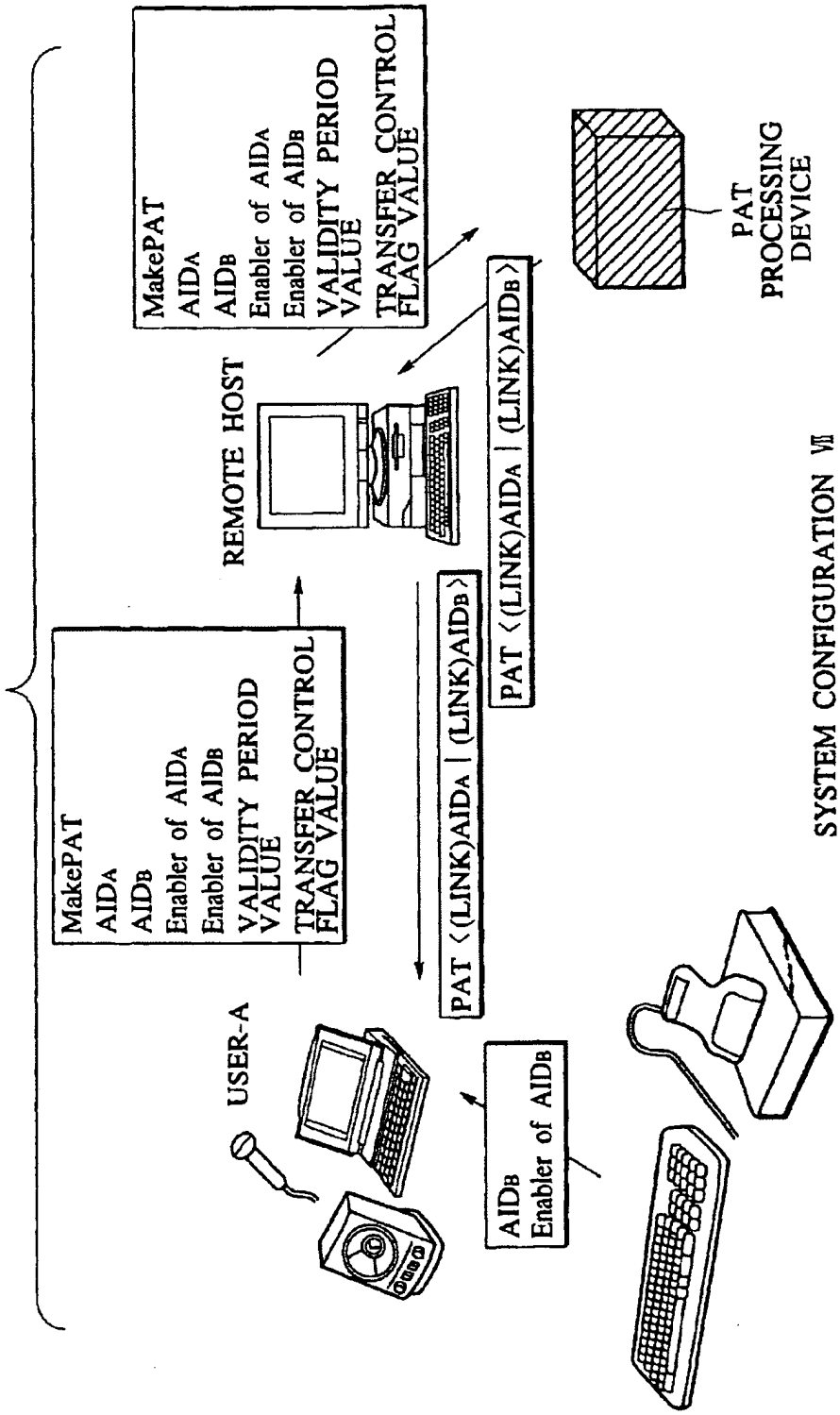
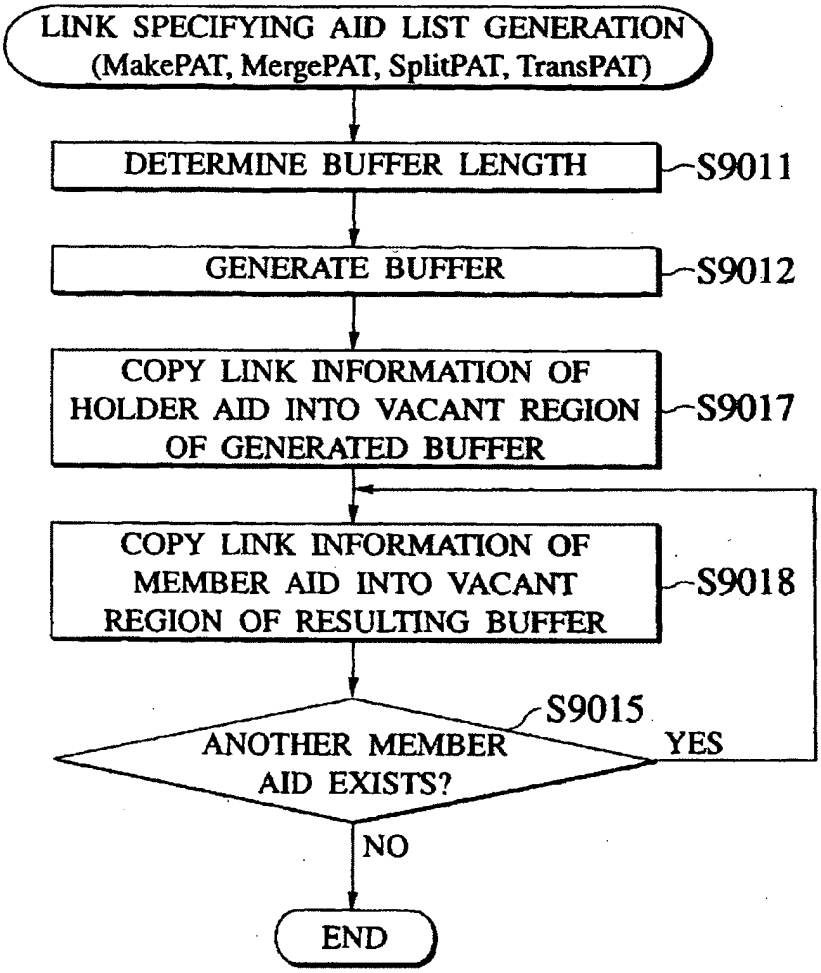


FIG.49



(12) UK Patent Application (19) GB (11) 2 354 102 (13) A

(43) Date of A Publication 14.03.2001

(21) Application No 9921227.6

(22) Date of Filing 08.09.1999

(51) INT CL[7]
G07F 7/10 , G06F 17/60

(52) UK CL (Edition S)
G4V VAK

(56) Documents Cited
EP 0813175 A2 WO 98/32260 A1 WO 97/50207 A1
WO 97/29416 A2 US 5809143 A

(58) Field of Search
UK CL (Edition R) G4V VAK , H4P PDCSA
INT CL[7] G06F 17/60 , G07F 7/10
Online: WPI, EPODOC, JAPIO

(71) Applicant(s)
Barron McCann Limited
(Incorporated in the United Kingdom)
BeMac House, Fifth Avenue, LETCHWORTH,
Hertfordshire, SG6 2HF, United Kingdom

(72) Inventor(s)
Peter Alderson
Robert Andrew Edge

(74) Agent and/or Address for Service
Williams, Powell & Associates
4 St Paul's Churchyard, LONDON, EC4M 8AY,
United Kingdom

(54) Abstract Title
System for communicating over a public network

(57) A system for communicating with a remote service over a public network 18, such as the Internet, includes a client device 10 with a memory card 28 or the like, a card reader 26 and a public network communication device such as a personal computer or television, and a processor unit, such as a central gateway 12, which is located remotely from the client device. The memory card includes user details which are transmitted by the client device to the processor unit, and may be encrypted. The card reader may activate communication with the processor unit upon insertion of the memory card, which may be a smart card or magnetic card. The processor unit may determine which of a plurality of services 14,16 a user is authorised to access. The system provides for secure communication without burdening the user with encryption or authorisation tasks.

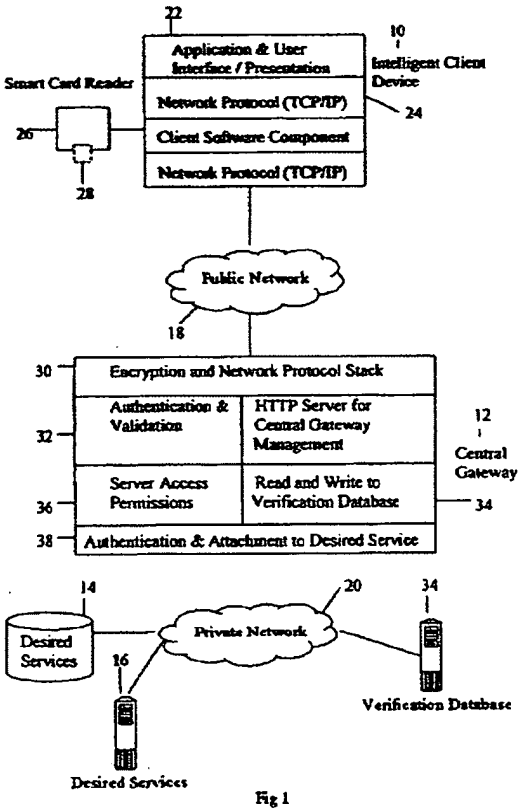


Fig 1

GB 2 354 102 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

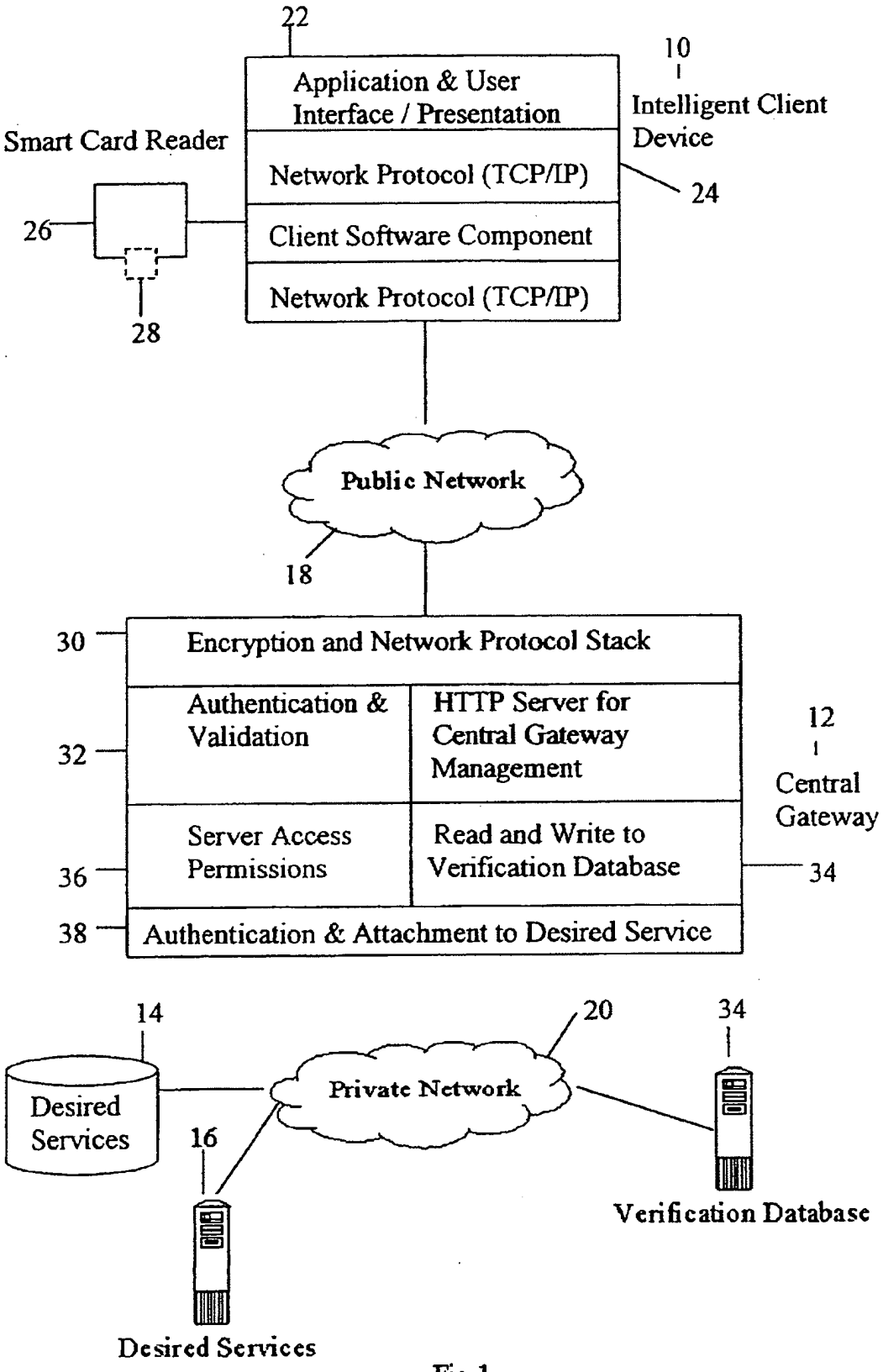

22
10
Application & User Interface / Presentation
Intelligent Client Device
Smart Card Reader
Network Protocol (TCP/IP)
24
26
Client Software Component
Network Protocol (TCP/IP)
28
Public Network
18
30
Encryption and Network Protocol Stack
Authentication & Validation
HTTP Server for Central Gateway Management
12
32
Central Gateway
Server Access Permissions
Read and Write to Verification Database
36
34
38
Authentication & Attachment to Desired Service
14
20
34
Desired Services
Private Network
16
Verification Database
Desired Services

Fig 1

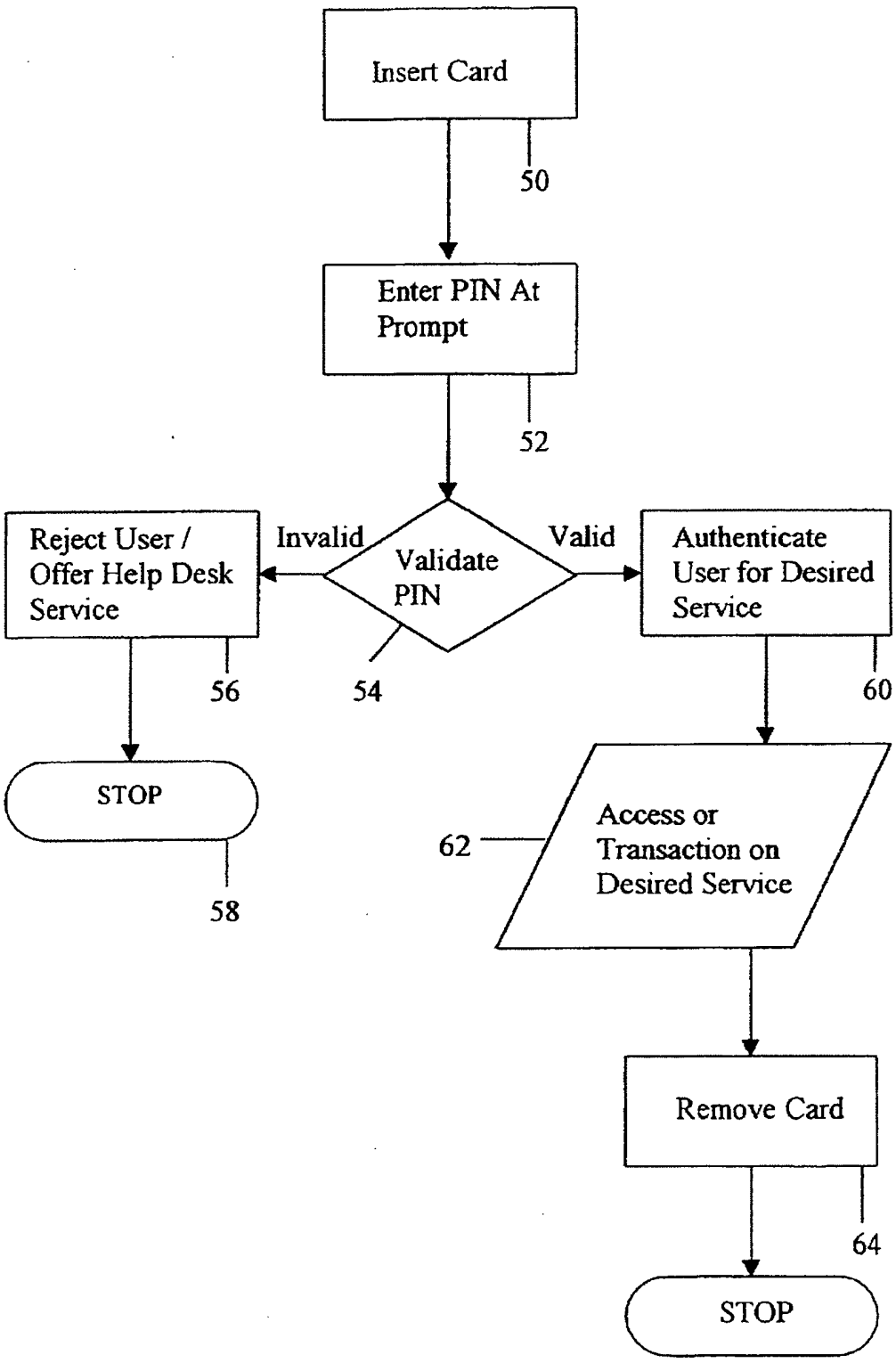

Insert Card
50
Enter PIN At Prompt
52
Validate PIN
Invalid
Valid
Reject User / Offer Help Desk Service
Authenticate User for Desired Service
54
56
60
STOP
58
62
Access or Transaction on Desired Service
Remove Card
64
STOP

Fig 2

SECURITY SYSTEM

The present invention relates to a security system, for use for example in accessing remote services such as on the Internet.

With the advent of modern technology, a growing number of transactions are being carried out by the user across insecure networks. These can be, for example, transactions involving confidential data and money for payment or investment. With such transactions there are problems with security, fraud and so on. Various security systems have been devised, such as use of personal identification numbers, encryption of transmissions. While these systems usually work well for the particular environment for which they have been designed, they can be a nuisance to use and can be difficult or expensive to implement for a new service provider.

Systems have also been developed for Internet use. These systems concentrate on authentication of the user and then, once this has been established, provide for un-encrypted connection to the service. When particular transactions are undertaken, the service determines whether encryption is necessary, for example to secure credit card details. Other solutions require entry of credit card details for each transaction. These systems inevitably must provide a balance between security and user convenience as the encryption mechanisms used cause additional work for and complication to the user.

The present invention seeks to provide an improved security system.

According to an aspect of the present invention, there is provided a security system for communicating with a remote service over a public network including a user card or other memory device, a user located card or memory device reader, a user located public network communication device and a processor unit located remotely from the user located public network communication device, wherein the user card includes user details and the user located public network communication device is operable to transmit the user details to the processor unit.

Advantageously, the processor unit is operable to carry out encryption between it and the user and to provide to the user a transparent path to the service. Thus, the user need not be aware of any security steps taken or any encryption system used, this being carried out by the card reader and the processor unit or central gateway.

The card may be any suitable device which can store user information and, preferably, encryption data. The card, can for example be a smart card, a magnetic card such as a credit/debit card or store loyalty card or any other suitable device. In addition to the card, the user may be required to input a secret identification code, such as an identification number.

In the preferred embodiment, the system provides for the user to insert the card into his/her card reader and to initiate the connection to the processor unit or central gateway. Once the connection is made, the processor unit obtains the relevant data from the card and upon verification by the identification code, allows the user access to the authorised service without any intermediate tasks, such as requirements to encrypt or decrypt transmitted data, to provide other user details and, where appropriate account or payment details. Thus, as with the preferred embodiment, all communications between the processor unit and the user can be encrypted, without the user necessarily being aware of or involved in this encryption. The communication between the user and the processor unit can therefore be totally secure yet without user inconvenience.

Advantageously, communications between the service and the processor unit, which are preferably carried out via a secure link, need not be encrypted.

The splitting of the encryption from the service results in being able to provide a dedicated encryption device, the processor unit, which can therefore be designed to maximise encrypted communication efficiency. Typically, encryption of all communications from the service unit is not practicable because the service unit is not designed for such a task and even if it were it would result in a loss of efficiency in providing the service itself.

In the preferred embodiment, the processor unit is also able to determine which of a plurality of services the user is authorised to access and/or the level of access such as spending limit, and to control access to the service or relevant service on this basis. It can also or alternatively undertake transactions against an account identified by the card.

An embodiment of the present invention is described below, by way of example only, with reference to the accompanying drawings, in which:

Figure 1 is a schematic diagram of an embodiment of security system coupled to a processor unit or central gateway and a service; and

Figure 2 is a flow chart of an example of validation routine for use with the system of Figure 1.

Referring to Figure 1, the embodiment of security system shown is designed for communications through the Internet or a similar public network.

The system includes an intelligence client device 10, which may be a personal computer, television, or any other suitable device which can communicate with a remote system. A processor unit, in this example a central gateway 12 is coupled between the client device 10 and one or more service units 14.

Communication between the client device 10 and the central gateway 12 is, in this embodiment, via a public network 18 such as the Internet. Communication between the central gateway 12 and the service units 14, 16 is, on the other hand, via a private network 20 which cannot be accessed by the public.

The client device 10 is provided with an application and user interface 22; which can be the usual computer devices such as monitor, keyboard and software in the case that it is a personal computer; the screen and a suitable keyboard or keypad in the case that the

device 10 is a television or any other suitable device. The device 10 could also be a portable telephone with suitable display and keypad.

The device 10 also includes suitable network protocol 24 for allowing communication to the gateway 12 through the chosen network 18 or other public transmission medium.

The device 10 also includes a card reader 26 designed for reading the card-type chosen for the system and a card 28 which is specific to that user. The card 28 could be a smart card or magnetic card of the types well known or any other portable memory device. It is envisaged that the card 28 could have other functions in addition to the security function for this system, for example it could also be a credit/debit card, store loyalty card and the like.

The card 28 has stored thereon one or more user identifiers, one or more encryption keys and the desired service information, that is details of the service to which the user wants access. His/her level of authorisation in the service and so on will be determined by the central gateway 12.

The card reader 26 is designed, in the preferred embodiment, to be able to detect the insertion of the card 28 thereinto and in response to such insertion to commence immediately communication with the gateway 12 via the client device 10.

The central gateway 12 includes an encryption and network protocol stack 30 designed to allow communication via the chosen public network 18 and to provide encryption of all communications between itself and the client device 10. It also includes an authentication and validation unit 32 for authenticating the client data from the client card 28. The authentication and validation unit 32 is coupled to a verification database 34 of the gateway 12 in which is stored the identification data of all the users registered for the services 14,16. The database 34 may be provided either within the gateway 12 or in a remote database 34' accesses through secure network 20.

The authentication and validation unit 32 is also coupled to server access permission unit 36 designed to control the type of access to the service units 14,16 in dependence upon the user's authority.

Also provided in the gateway 12 are a typical HTTP server for management of the gateway 12 and an authentication and attachment unit 38 for communicating with the desired services 14,16 and with any remote verification database 34'.

The central gateway 12 is designed specifically for encrypting all communications over the public network 18 and for carrying out the authentication procedure.

The operation of the this embodiment will now be described with reference to Figure 2.

Insertion 50 of the card 28 into the card reader 26 prompts the card reader 26 to commence automatically the connection to the gateway 12. For this purpose, card reader 26 activates a software component in the device 10 to establish a communication link with the gateway 12 on the basis of information stored on the card 28 about the location on the Internet and access details of the gateway 12.

When a connection with the gateway 12 is established, the gateway 12 requests the user's personal identification code which is then inputted 52 at a suitable prompt on the user interface 22.

Validation 54 of the user's details and identification code is carried out either internally of the gateway 12, by the units 32 and 34, or externally at the verification database 34'.

If the gateway 12 determines 54 that the user's identification code is invalid, the user is rejected 56 and the connection is cut 58. On the other hand, if it is determined 54 the user's identification code is valid, the gateway 12 determines 60 the desired service 14, 16 and level of service to be provided and connects 62 to the desired service unit 14, 16.

During the connection to the desired service 14, 16, all data transfers between the gateway 12 and user device 10 are encrypted on the basis of the encryption keys on the user's card 28 and within verification database 34, while all data transfers between the gateway 12 and the service units 14, 16 through the private network 20 are not encrypted for ease of access and for increased efficiency. In practice, the user will not be aware of the encryption between him/her and the gateway 12 as this will be carried out as a background task. Moreover, the user will not need to re-confirm his/her identity or financial details as these will be provided by the card 28 or gateway 12.

The gateway 12, in some embodiments, records the activities of the client, such as transaction details, either within the gateway 12 or in a remote memory accessed via a private network.

Disconnection from the services 14, 16 is, in this embodiment, effected simply by removing 64 the card 28 from the card reader 26.

Thus, connection is made by a simple two step process of inserting the card 28 into the reader 26 and entering the user identification code and disconnection is effected by removing the card 28 from the card reader 26. The user is not involved in any other authentication or encryption process and need not re-enter personal details.

This system can be used for any remote service, including business to consumer (in which case the card could be designed also to function as a store or credit card), business to business (for example for transactions on account) and for internal networking (where the activity of staff, for example, needs to be secured).

It will be apparent from the above that the system can provide simple but absolutely secure access to a remote service. Moreover, by identifying the user to the desired service, user access can be customised. By removing the need for entry of account details, transactions into the desired service become quicker and less risky for the user's perspective.

Performance of the services can also be enhanced by carrying out the encryption tasks within the gateway rather than in the service units.

In addition, the service company can establish a relationship with the user by providing the user with the card and, possibly, also with the card reader.

It will be apparent that the card 28 and card reader 26 could be configured to communicate with a plurality of separate gateways 12.

CLAIMS

1. A security system for communicating with a remote service over a public network including a user card or other memory device, a user located card or memory device reader, a user located public network communication device and a processor unit located remotely from the user located public network communication device, wherein the user card includes user details and the user located public network communication device is operable to transmit the user details to the processor unit.

2. A security system according to claim 1, wherein the processor unit is operable to carry out encryption between itself and the user.

3. A security system according to claim 1 or 2, wherein the card has stored thereon user information and, preferably, encryption data.

4. A security system according to claim 3, wherein the card is a smart card, a magnetic card or any other suitable device.

5. A security system according to any preceding claim, wherein the card reader is operable to activate communication with the remote processor means upon insertion of a card thereinto.

6. A security system according to any preceding claim, wherein the processor unit is operable to encrypt substantially all communications between the user and itself.

7. A security system according to any preceding claim, wherein the processor unit is operable to determine which of a plurality of services a user is authenticated onto the desired service.

8. A security system substantially as hereinbefore described with reference to and as illustrated in the accompanying drawings.

The Patent Office

10.

INVESTOR IN PEOPLE

Application No: GB 9921227.6 **Examiner:** Michael Logan
Claims searched: All **Date of search:** 20 January 2000

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): G4V (VAK); H4P (PDCSA)

Int Cl (Ed.7): G06F 17/60; G07F 7/10

Other: Online: WPI, EPODOC, JAPIO

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	EP 0813175 A2	(NCR INTERNATIONAL) whole document relevant	1-6
X	WO 98/32260 A1	(COMMONWEALTH BANK OF AUSTRALIA) see page 2 and fig 1	1-6
X	WO 97/50207 A1	(TELIA AB) see page 9, lines 1-24	1-6
X	WO 97/29416 A2	(INTEGRATED TECHNOLOGIES OF AMERICA) see especially page 7, line 5 - page 8, line 16	1-7
X	US 5809143	(HUGHES) see for example column 10, lines 35-43	1-6

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

55) Family number: 10272458 (JP5168039 A2) | ✉ | 🖶 | full-text | status | citations | < | > | ^ |

Title: RECORDING ENCODE METHOD FOR HIGH FIDELITY TELEVISION SIGNAL

Priority: JP19910352059 19911213
Priority Map

Family: Family Explorer

Publication number	Publication date	Application number	Application date	Link
JP3185806 B2	20010711	JP19910352059	19911213	
JP5168039 A2	19930702	JP19910352059	19911213	

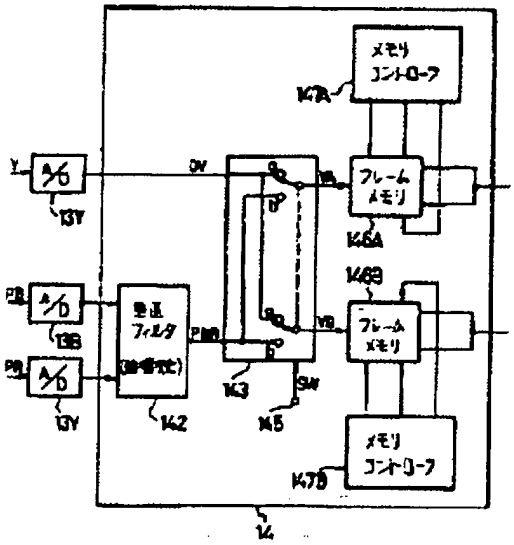
Assignee(s): SONY CORP

Inventor(s): ISHIMARU HIROYOSHI

International class (IPC 8): H04N11/22 H04N5/907 H04N9/80 H04N9/81 (Advanced/Invention); H04N11/06 H04N5/907 H04N9/80 H04N9/81 (Core/Invention)

International class (IPC 1-7): H04N11/22 H04N5/907 H04N9/80 H04N9/81

Abstract:
Source: JP5168039A2 PURPOSE: To encode a unit signal (TDM signal) for recording from a high fidelity television signal by controlling reading of plural output ports while using a serial access memory equipped with the plural output ports. CONSTITUTION: Memories 146A and 146B are serial access and two output ports are respectively provided in each memory. Then, write of input data VA and VB is controlled by memory controllers 147A and 147B, and reading of data from the respective output ports is independently controlled. Namely, TDM signals are written in memories 146A and 146B in the order of a luminance signal and a chrominance signal. In the case of reading, the same data are read from two output ports while deviating read timing, color difference signal data are extracted from the preceding output port, luminance signal data are extracted from the other output port, both data are synthesized and therefore, the required TDM signals are obtained.



PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
G07F 7/08, 19/00, G06F 17/60 // 157:00

A1

(11) International Publication Number: **WO 96/13814**

(43) International Publication Date: 9 May 1996 (09.05.96)

(21) International Application Number: PCT/FI95/00591

(22) International Filing Date: 25 October 1995 (25.10.95)

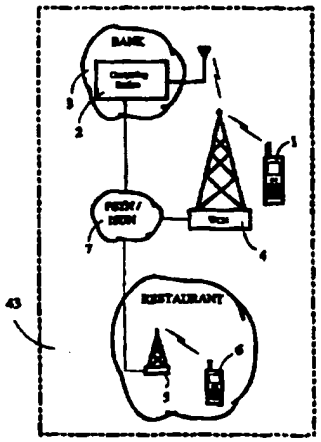
(30) Priority Data:
945075 28 October 1994 (28.10.94) FI

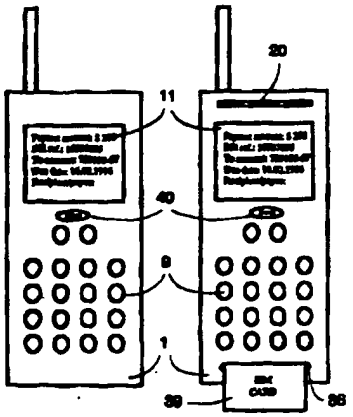
(71)(72) Applicant and Inventor: VAZVAN, Behruz [FI/FI]; Jämeräntaival 11 B 53, FIN-02150 Espoo (FI).

(81) Designated States: FI, JP, US, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
With amended claims and statement.

(54) Title: REAL TIME TELE-PAYMENT SYSTEM





(57) Abstract

This invention is a real time mobile tele-payment system that relates to payments of bills of mobile users, or providing the mobile users with the information about their bank account, the statement of account, or the movement on the account in a real time basis, by using their portable telephones under any wireless telecommunications systems. Certain features of this invention are intended as an expansion of value-added services of currently existing mobile communications systems. This invention also provides the retail and trading businessmen with the possibility to charge their customers, via wireless communications networks and in a real time basis, by using their mobile telephones. In this invention, in order to pay his/her bills, a mobile telephone subscriber enters the payment (bill) information and the payee's account number into the mobile payment part (10) which is included in his/her mobile telephone (1) or (6). After having dialled the telephone number of computing station (2) which is based in the bank (3), the payment information will be sent to the computing station (2) via a mobile communications network (4). In the computing station (2) the calling party's identity will be checked and then the payment will be transferred from the calling party's bank account to the payee's account and then both the calling party and the payee will be informed about the relevant payment. In this invention, the portable telephone is also equipped with a small charge slip printer which can print a receipt for customers of retail businesses.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

Real Time Tele-payment System

This invention is a mobile payment system that relates to payments of bills of the mobile users, or providing the mobile users with the information about their bank account balance, the statement of account, or the movement on the account in a real time basis, by using their portable telephones under any wireless telecommunications systems.

BACKGROUND OF THE INVENTION

There are several mechanical and electronical payment systems for retail business operations like, for example, what is introduced by US patent US-A-5 202 825, in which a hand-held data terminal generates a record of purchases made by a customer for charging a customer in accordance with customer-indicated payment preferences. In these systems the waiter sends by use of a portable data terminal the customer's order to a customer service station which is a typical cash register based in the restaurant. These systems reduces the time requirements for processing customers at check-out counters in comparison with those of more traditional check-out procedures of the recent past. These systems are only for sending the customer order to the cash register in the retail business.

On the other hand in the fixed telecommunications networks a user (subscriber) can be connected from his personal computer to his/her bank via telephone lines and thereby pay his/her bills. In such systems user must use a data modem between his/her computer and the telephone wire. Another disadvantage of such systems is that in order to pay his/her bills, user must have access to a personal computer connected to the fixed telephone infrastructure, therefore user mobility in such systems is completely limited. Before this invention, there was no solution that provides the portable/mobile telephone users with the possibilities to pay their bills by using their personal portable telephones. There was also no payment system, based on use of portable or mobile telephones, that could provide the retail or trading businesses with the possibility to charge their customers in a real time basis; transferring the charges from the customer's account to the account of the retail businessman. There continues to exist a need to further improve the efficiencies of payment systems.

DESCRIPTION OF THE INVENTION

In order to serve such current need, the present invention provides a new and unique mobile payment system. In the inventive system a portable telephone can be used in order to pay bills or transfer money from a bank account to other, or request the bank

for account information. Certain features of the invention are intended as an expansion of value-added services of currently existing mobile communications systems. This invention addresses needs created by users mobility. For example, suppose that you are travelling and you want to pay a certain bill or transfer some amount of money from your bank account to another person's account but you do not have time for going to the bank or the bank may be closed and you may neither have access to your personal computer (which can be connected to the bank via telephone wire). This invention provides you the possibility to pay your bills, by using your portable telephone while you are in move, regardless of are banks closed or not, regardless of if it is night or weekend etc. This invention also provides the retail businesses (for example restaurants) the possibility to charge their customers, via wireless telecommunications networks, by using only the portable telephones. For example, a waiter in a restaurant, after having entered the amount of payment and customer's information (like account number etc.) to his/her portable terminal can send the payment information to the inventive computing station, which is located in the bank. In the computing station the customer's bank account will be charged in accordance with the payment amount received from the waiter's portable telephone. The most important advantage gained by the inventive system is that all mobile telephone subscribers can pay their bills by using only their normal mobile telephones (in which the mobile payment part is included) and their subscriber identity or codes, without requiring any additional data modem, personal computer, and credit cards etc. In this invention the subscriber identity and codes function as the credit card or bank card of the portable terminal's user.

By implementing the inventive mobile payment system a mobile user (subscriber) can pay all his/her bills and handle all his/her banking issues by only using his/her mobile telephone and subscriber identity or codes, where ever under the coverage of a wireless communications network. These and other improvements and advantages are realised by providing a portable telephone (hereafter called portable terminal) including the inventive mobile payment part, and a computing station which is based in the bank. The present invention will now be described by way of examples with reference to the accompanying drawings, in which:

Fig. 1 is a schematic representation of the inventive Real Time Tele-payment System.

Fig. 2 represents, as an general example, a payment flow diagram between the portable terminal and the computing station, which is located in the bank.

Fig. 3 represents, as an general example, a payment flow diagram in which a mobile user pays his/her bills or request the statement of his/her account by using his/her own

portable telephone. In this figure also the payee is informed about the reception of a payment.

Fig. 4 is a schematic representation of two type of portable terminal: one is a normal portable telephone that includes the inventive mobile payment part, and the other is a portable telephone that includes the inventive mobile payment part, a charge slip printer and a user-friendly SIM card reader (SIM: Subscriber Identity Module).

When a mobile user wants to pay a bill or transfer money from an account to other, he/she enters all information required for payment (like his/her account number, the payee's account number, payment's due date, bill's reference number, etc.) to the mobile payment part of his/her portable terminal **1** (for example through the keypad). As it is the object of this invention, the user's own account information dose not need to be entered into the mobile payment part if the computing station **2**, based in the bank **3**, can identify the calling party. This needs that the user information (identity) should be confirmed by his/her telephone operator or service provider in a wireless communications network **4** and then be sent to the bank as a confirmation of user (subscriber) identification. More precisely, user identity can be sent by user's telephone operator or service provider to the computing station **2** when portable terminal **1** set-ups a call or a short message to the computing station **2**. Monitoring a calling party's subscriber number or information at a receiving terminal is a feature provided by today's digital telephone systems. In this invention, in order to implement such procedure, for example the switching systems at the mobile network side can be used so that only when a user set-ups a call or sends a message (by using short message services of the mobile communications systems) to the computing station **2** his/her identity can be monitored in the computing station **2** in order to identify who is the calling party. Therefore, in this invention the computing station **2** receives at least the confirmed user identity from the user's telephone operator or service provider of a wireless communications network (WCN) **4** in order to identify who is in charge for payment of bills sent by portable terminal **1**. Other required information like passwords or access codes to the user's bank account will be sent by user through his/her portable terminal **1**. In today's mobile telecommunications systems the user identity, included in his/her SIM card, is checked and confirmed by network **4** every time his/her portable terminal **1** is turned on and attached to the telephone network **4**. Since the user identity, transmitted from the portable terminal **1** to the network **4**, is completely encrypted and secured therefore the payment messages between portable terminal **1** and computing station **2** are also quite secured because of: first, the security algorithms used in the today's digital wireless telecommunications systems and mobile telephones, and secondly, because of the user's password or access codes used for payment messages in the inventive mobile payment system. All kind of wireless

communications networks can be used in order to communicate the payment messages between the portable terminal and computing station. For example if in the restaurants there is a cordless network like DECT (Digital European Cordless Telephony) **5** then the portable terminal **6** can be connected through such network and PSTN (Public Switched Telephone Network) or ISDN (Integrated Services Digital Network) **7** to the computing station **2**.

The payment question-answering procedure between the user and portable terminal **8** is entered by using the user interface **9** and received and handled by the inventive mobile payment part **10**. The payment information entering procedure **11** is an interactive procedure between the mobile payment part **10** and the user through user interface **9**. Then, the computing station's telephone number will be dialled **12** (either automatically or by user) which after the portable terminal **8** sends the required information for call set-up to the wireless communications network **15** and then payment messages **13** to the computing station **14** via the same network **15**. If the portable terminal **8** dose not send the user (telephone subscriber) identity to the computing station **14**, then the wireless communications network **15** confirms and sends the user identity to the computing station **14** either directly or through the fixed public network **16**. The computing station **14** checks the calling party's account and account number of payee (the account to which the payment should be transferred) and then transfers the required amount of payment from the payer's account to the account of payee **17**. After that the payment has been completed the computing station **14** sends a message **18** to the portable terminal **8** indicating "payment completed" or if there is not enough credit (money) in the payer's account a "No effects" message **19** will be sent to the portable terminal **8**, meaning that the payment can not be accepted. For retail businesses, portable terminal includes also a charge slip printer **20**. If the portable terminal receives a "payment completed" command **18**, the charge slip printer **20** prints a receipt for the customer. In this invention for the retail and trading businesses, the customer's SIM card **39** is entered in the SIM card reader **36** of the portable terminal **1** (of a waiter in a restaurant, for example) temporary in order to pay the bill. Then the portable terminal **8** will be connected to the wireless communications network **15**. The account number of payee (for example account number of the restaurant) can be saved in the memory of his/her portable terminal in order to reduce the information entering procedure of the mobile payment part. This means that only the payment amount should be entered to the mobile payment part. After that the payment amount has been entered to the mobile payment part **10** and the computing station's **14** telephone number has been dialled **12**, the wireless network **15** sends the customer's identity, which can be the subscriber identity or a different code,

to the computing station **14**. The computing station **14** can identify the calling party (the payer) because it has received the calling party's identity from the wireless network **15** and compared with the calling party's identity based in the computing station **14**. Therefore the calling party will be charged for the payment amount received from the portable terminal **8**. The subscriber identity sent from the wireless network **15** to the computing station **14** can be different than the payer's identity sent by the portable terminal **8** to the wireless network **15** but both of these identities belong to one user (subscriber). Alternatively the payer's identity, included in his/her SIM card **39** or entered to the portable terminal by using user interface **9**, can be sent directly from the portable terminal **8** to the computing station **14**. It should be understood that for the simplicity of the description, messages for outgoing call set-up and incoming call or short message services procedures are not explained with details since these procedures are already well known in the mobile communications systems.

Following is an example, in which a mobile user pays his/her bills or transfers money form his/her bank account to other, or ask the bank for statement of account, by using his/her own portable telephone.

First, the payer enters the bill's information **22** (for example: account number of payee, the amount of money which should be transferred, due date of the bill, reference number **11**) to the mobile payment part **21** of his/her portable terminal **41**. Then, after activating an OK function by user, the mobile payment part dials **23** the telephone number of the computing station located in the bank **24**, which after the mobile payment part **21** sends the payment information **25** to the computing station **24**, via a wireless communications network (WCN) **26** and fixed network **27** (PSTN/ISDN). Then, computing station **24** transfers the amount of payment, mentioned on the bill, from the payer's account to the payee's account **28**. Then, computing station **24** sends a "Payment Completed" message **29** to the portable terminal's mobile payment part **21**. If the payee has also a portable terminal **37**, then also his/her mobile payment part **42** would receive a "Payment Reception message" **30**, from computing station **24**, indicating the amount of payment, the payer and the payment date. However, before dialling the number of computing station, the mobile payment part may ask the payer (the user of portable terminal) "Any other payment ?" **31**. The answer can be respond by activating "Yes/No" function **32** or OK function of the mobile payment part **21**. Then the user can enter another bill information to the mobile payment part **21** and when all information required by mobile payment part has been provided, the telephone number of computing station **24** will be dialled **23**. After this, all bills information (payment messages) will be sent to the computing

station in the bank **24** as explained above. Furthermore, there is a command **33** "Send the Statement of Account" in the mobile payment part **21** for requesting the account balance, the statement of account, or the movement on the account from the computing station **24**. When a user selects such command **33**, the mobile payment part **21** sends this message **33**, either by setting up a call or by using the short message facilities of mobile communications networks **26** to the computing station **24**. Then computing station **24** sends the required account balance or the statement of account **34** to the mobile payment part **21** of the portable terminal **41**. The computing station **24** also sends a "Monthly Statement of Account" **35**, to the portable terminals **41, 42** once or twice per month. Then portable terminal's printer **38** can print it for the user to be filed as a record, if required.

Following is an example in which the payee (for example a restaurant or a retail seller) has a portable terminal by which the payer's (a customer) account can be charged.

Suppose that a customer wants to pay his/her bill in a restaurant for the service he/she has received. The customer can give his/her SIM card **39** or credit card to the waiter to be entered to the waiter's portable telephone **1, 8**. Then waiter dials the telephone number of computing station **14**, or the number will be dialled automatically after the SIM card **39** or credit card has been read by the SIM card or credit card reader **36** of the waiter's portable terminal. For example the telephone number of computing station **14** can be saved in the memory of the portable terminal of waiter, and every time a customer's SIM or credit card **39** is entered to the portable terminal **1**, the portable terminal automatically contact the computing station **14**, after having registered in the network **15**. In the bank, the computing station **14** checks the account information of payer (a customer) and then transfers the transaction amount (the sum on the bill) to the payee's (the restaurant) account **17**. If the payer's account do not have enough credit (money) the portable terminal **8** may receive a "No effects" message **19**, or the bank may pay the transaction's amount on behalf of the payer and then later charge the payer or his/her bank for the prepaid transaction. On the other hand if the payer's account information (account number, account identity) is false the computing station **14** may send a "transfer not accepted" message to the payee's portable terminal, which means that the payer (customer) should pay the amount of transaction in cash. If the portable terminal receives from the computing station **14** a "payment completed" message **18**, then the charge slip printer **20** prints a receipt for the customer, as explained in the first example.

It should be considered that in all above-mentioned examples, payment messages can be sent and received either by setting up a call between the portable terminal and computing station or by using short message services facilities of the wireless communications networks.

In the current mobile communications systems, like GSM, there is a facility called "Short Message Services, (SMS)". In SMS a mobile telephone user can send short messages to another subscriber without setting up an interactive call. In order to send the payment messages by SMS, the software of SMS installed in the portable terminal can be modified so that it can also handle the payment parameters and/or commands of the inventive mobile payment part **10**. Then by using the SMS services of the wireless communications network **15**, the bill's information **13** can be sent to the computing station **14**. When computing station **14** receives such payment message **13** sent by SMS, it also generates a message to be sent to the portable terminal in order to inform it if the payment has been completed **18** or not **19**. However, if a user wants to pay many payments (bills) at once and receive also balance or statement of his/her bank account from the computing stations, such long message can be divided to smaller parts and then be combined at the portable terminal or computing station. This means that each bill information can be sent separately using the short message services. This action is transparent to the user of portable terminal. For example several payment information can be entered to the mobile payment part **10**. Then when user selects the "Send" function **40** on the portable terminal **1**, each bill will be sent by one short message in accordance of short messages length. For example, a short message may not include more than 100 letters. If a payment message or the statement of account (sent by computing station) needs more than the assumed 100 letters, then such long information will be divided into two or several short messages and then will be sent one by one to the portable terminal or computing station.

In this invention computing station can send and receive messages either via PSTN (Public Switched Telephone Network) and ISDN (Integrated Services Digital Network) and other fixed networks or via only a wireless communications network. The computing station includes all means for transmitting and receiving payment and banking messages via the wireless networks.

It is to be understood that various changes and modifications can be made to alter the specifically described structure or methods of operation of the preferred embodiment without departing from the spirit and scope of the invention. This invention is to be defined only by the scope of the claims appended hereto.

Claims

1. A mobile payment system (43), characterised in that it is comprised of:

- at least one portable terminal (1, 6, 8), such terminal including a mobile payment part (10, 21) and other means for entering, transmitting, receiving and printing of information relating to: the payments of bills of the telephone subscriber or the user of said portable terminal; transferring of money from the bank account of the subscriber or user to the others account; sending and receiving payment messages (13, 18, 19, 25, 29) or messages including the account balance, the statement of account, or the movement on the bank account (33, 34, 35) of the telephone subscriber or the user of the portable terminal (41, 37);
- at least one computing station (2, 14, 24) which is located in the bank (3), said computing station including means for communicating with said portable terminal and for transferring the amount of payment (money) from the bank account of portable terminal's user and/or telephone subscriber to another bank account (17, 28), or from a customer's bank account, whose account information is entered into said portable terminal, to the calling party's account; and to receive and send messages about the account balance, the statement of account, or the movement on the bank account (33, 34, 35) of the portable terminal's subscriber or user;
- at least one wireless communications network (4, 15, 26) through which said portable terminal can send and receive to or from said computing station said payment messages or messages about the account balance, the statement of account, or the movement on the bank account of said portable terminal's subscriber or user.

2. A mobile payment system (43) according to claim 1, characterised in that said at least one portable terminal (1, 6) is a first plurality of portable terminals, and in which the number of said portable terminals in said first plurality of portable terminals is greater than said at least one computing station (2).

3. A mobile payment system (43) according to claim 1 and 2, characterised in that the payments or bills of a mobile telephone subscriber can be paid by entering the subscriber identity and codes into said portable terminal (1, 6, 8, 41) and the bill's information, including the payee's bank account number, the amount of payment, bill's due date and reference number into the mobile payment part (10, 21) of said portable terminal, and by setting up a call or a short message to the bank's computing station

(2, 14, 24) and sending the payment (bill's) messages (13, 25) to said computing station (2, 14, 24).

4. A mobile payment system (43) according to claim 1, 2 and 3, characterised in that said at least one portable terminal (1, 6) comprises all means for transmitting and receiving payment messages to or from said computing station (2); and that:

- said portable terminal includes a mobile payment part (10, 21) for handling the payment information (11, 22, 31, 32) entered by user to said portable terminal, and that said payment information can be saved into the memory of said portable terminal and be sent to said computing station, whenever required; and that:
- said portable terminal receives a message (18, 19, 29) from said computing station indicating that either the payment or transferring of the required amount of payment from the payer's to the payee's bank account has been accepted and/or completed or not.

5. A mobile payment system (43) according to claim 1, 2, 3 and 4 characterised in that the user of said portable terminal can enter more than one payment or bill information to the mobile payment part (10, 21) , and that after that telephone number of said computing station based in the bank (2, 14, 24) has been dialled (12, 23) either manually or automatically, all required payment information (13, 25) will be sent to said computing station; and that:

- said portable terminal can send payment (bill's) information (13, 25), handled in mobile payment part (10, 12), to the computing station (14, 24) and receive the required payment messages (18, 19, 29) from said computing station by setting up a call or using the Short Message Services (SMS) of the wireless communications network (4, 15, 26); and that
- said portable terminal's subscriber information can be sent from the user's telephone operating network (4, 15, 26) to the computing station (2, 14, 24); and that
- said portable terminal includes a charge slip printer (20, 38) that can print all payment information and the information received from said computing station for user of said portable terminal, and that,
- said mobile payment part (10, 21) can be included into any kind of digital or analogue portable telephone that is capable of operating in cellular communications systems.

6. A mobile payment system (43) according to claim 1 - 5, characterised in that said computing station (2, 14, 24) after receiving a payment message (13, 25) from said portable terminal (1, 6, 8, 41), checks and charges the payer's account (17, 28) in accordance with the payment amount received from said portable terminal and then sends a message (18, 19, 29) to said portable terminal (8, 41, 37) in order to indicate that payment has been accepted and/or completed or indicating that there is not enough credit in the payer's account; and that:

- said computing station (2, 14, 24) can receive or send payment messages (18, 19, 29, 30) or other banking messages (33, 34, 35) to said portable terminal (1, 6, 8, 41) via either fixed and wireless communications networks (4, 7, 15, 16, 26, 27) or via only wireless communications network (15, 26); and that,

- said computing station (2, 14, 24) can receive the payer's information and identity either from the payer's telephone operator or service provider through wireless communications network (4, 15, 26) when payer telephones or send messages (13, 25) to said computing station (2, 14, 24) or from the payer's portable terminal (1, 6, 8, 41); and that, the payer's information received from said payer's telephone operator or service provider or from said portable terminal may include payer's subscriber information or identity or any other required information; and that,

- said computing station can monitor the subscriber information or other identity, received from said payer's telephone operator or service provider or portable terminal, and based on said subscriber information or other identity and account number transfer the required amount of payment (money) from the payer's account to any other required account; and that,

- said subscriber information or identity will be confirmed by subscriber's telephone operator or service provider (4, 15, 26) and said confirmed information will be sent to said computing station (2, 14, 24) in which the subscriber identity will be checked (17, 28) and based on that, the received payment message (13, 25) can be accepted and a payment completed message (18, 29) will be sent to said portable terminal (1, 6, 8, 41); and that,

- said computing station (2, 14, 24) can send or receive payment messages (13, 18, 19, 25, 29, 30, 33, 34, 35) to or from the portable terminals (1, 6, 8, 41, 37) of both the payer and the payee; and that,

- said computing station (2, 14, 24) is equipped with all means for transmitting and receiving messages via any wireless communications network, to or from said portable terminal (1, 6, 8, 41, 37).

7. A mobile payment system (43) according to claim 1 - 6, characterised in that the mobile payment part (10, 21) may ask the user to enter all payment information

(11) such as payee's account number, bill's reference number, bill's due date, the amount of payment and other required information; and that:

- said mobile payment part (10, 21), after receiving all information about a payment or a bill from the user through user interface (9), may ask the user of said portable terminal "any other payment ?" (13) indicating dose user wants to pay another bill or payment; and that,

8. A mobile payment system (43) according to claim 1 - 7, characterised in that said portable terminal (1, 6, 8, 41) can be used in order to pay the bills of any mobile telephone subscriber by entering each subscriber's identities and codes into said portable terminal either by using the portable terminal's user interface (9) or the SIM card (39) and card reader (36); and that:

- said mobile telephone subscriber's codes can be different than said subscriber's identities; and that said subscriber codes can be included both in the subscriber's SIM card (39) and said computing station (2) located in the bank (3); and that:

- said portable terminal (1, 6) can be used in order to charge customers, in retail or trading businesses, by entering the customers' telephone SIM card (39) into said portable terminal (1, 6) and by using the telephone subscriber identities of each customer as an identification for payment; and that:

- after that said customer's SIM card (39) has been entered to said portable terminal (1, 6, 8), said portable terminal will be re-connected to the wireless communications network (4, 15) in order to check the subscriber identity, which after the customer's (subscriber's) bank account can be charged by sending payment messages (13) to the computing station (2, 14).

AMENDED CLAIMS

[received by the International Bureau on 25 March 1996 (25.03.96); original claims 1 and 3-8 amended; new claims 9 and 10 added; remaining claims unchanged (8 pages)]

1. A mobile payment system (43), utilizing the Short Message Services (SMS) facilities of mobile communication networks such as GSM (Global System for Mobile Communications), and subscriber identity such as SIM card (Subscriber Identity Module), and a new mobile-telephone-based functionality and mobile network architecture **characterized** in that it is comprised of:

- at least one portable terminal (1, 6, 8), such terminal utilizing the inventive Mobile Payment Part (10, 21), which provides a function and SMS-based adaptation and application part integrated into said portable terminal to provide at least an alphanumeric payment (bill) inquiry (e.g. 11), and including other means for entering, transmitting and receiving, and printing of the information mainly related to: the payments of bills of the telephone subscriber (1, 6, 8); transferring of money from the bank account of the subscriber or user to the others account; sending and receiving payment messages (13, 18, 19, 25, 29) or messages including the account balance, the statement of account, or the movement on the bank account (33, 34, 35) etc. of the telephone subscriber of the portable terminal (41, 37) without requiring to use any additional data modem to be used in conjunction with said portable terminal for transmission and reception of said payment etc. messages;

- at least one computing station (2, 14, 24) which is located in the bank (3), as it is the object of the architecture of the inventive payment system (43), said computing station includes all information about the portable telephone subscriber data which is connected to the subscriber's bank account in the same bank wherein computing station is located, and said computing station includes means for communicating with said portable terminal (4) and transferring the amount of payment (money) from the bank account of the payer (i.e. the calling subscriber) to another bank account (17, 28), and to receive and send messages about the payments, account balance, the statement of account, the movement on the bank account (33, 34, 35) or other banking messages etc. of the calling subscriber via SMS facilities of the wireless communication network (4);

- at least one wireless communication network (4, 15, 26) equipped with Short Message Services (SMS) infrastructure through which said portable terminal (1, 6, 8) can send and receive to or from said computing station said payment messages or other banking messages etc. and that said wireless communication network can confirm (i.e. authenticate) the subscriber identification for said computing station, whenever required or transfer the subscriber data received from said portable terminal directly to said computing station, in which the subscriber data can be compared with the subscriber data already recorded there.

2. A mobile payment system (43) according to claim 1, **characterized** in that said portable terminal (1, 6) is a first plurality of portable terminals, and in which the number of said portable terminals in said first plurality of portable terminals is greater than said at least one computing station (2).

3. A mobile payment system (43) according to claim 1, 2, **characterized** in that said portable terminal (1, 6, 41) comprises all means for transmitting and receiving payment etc. messages to or from said computing station (2) or other portable terminal (37); and that,

- said portable terminal includes the inventive Mobile Payment Part (10, 21) which is a short-message-based adaptation and application part for handling, dividing or connecting the payment etc. information (11, 22, 31, 32), and that said payment etc. information can be saved into the memory of said portable terminal and be sent to said computing station, whenever required; and that,

- after that portable terminal has been registered into the mobile network (4), the payments or bills of the mobile telephone subscriber (1, 6) can be paid by entering the bill's information such as the payee's bank account number, the amount of payment, bill's due date and reference number etc. into the Mobile Payment Part (10, 21), and by sending the short messages (e.g. 13, 33) to the bank's computing station (2, 14, 24) via SMS facilities of the mobile network (4) and receiving messages such as (18, 19, 30, 34, 35 etc.).

- said portable terminal receives a message (e.g. 18, 19, 29) from said computing station indicating that either the payment or transferring of the required amount of money from the payer's to the payee's bank account has been accepted and/or completed or not; and that,

- said portable terminal includes a charge slip printer (20, 38) that can print all payment information and the information received from said computing station for user of said portable terminal, whenever required.

4. A mobile payment system (43) according to claims 1, 2, 3, **characterized** in that the computing station (2, 14, 24) after receiving a payment message (13, 25) from said portable terminal (1, 6, 8, 41), checks and charges the subscriber's (payer's) account (17, 28) in accordance with the payment amount received from said portable terminal and then sends back a message (e.g. 18, 19, 29) including all information about the payment (e.g. bill reference, payer, amount etc.) to said portable terminal (8, 41, 37) in order to indicate that the payment has been accepted and/or completed or indicating that there is not enough credit in the payer's account; and that:

- said computing station (2. 14. 24) can receive or send payment messages (e.g. 18. 19. 29. 30) or other banking messages (e.g. 33. 34. 35) or any other message to said portable terminal (1. 6. 8. 41) via SMS of a mobile communication network (4. 5) through either fixed and wireless communication networks (4. 5. 7. 15. 16. 26. 27) or via only wireless communications network (4. 15. 26); and that.

- said computing station (2. 14. 24) can receive the payer's identity either from the payer's telephone operator system (4. 15. 26) when payer sends messages (e.g. 13. 25) to said computing station (2. 14. 24) or from the payer's portable terminal (1. 6. 8. 41); and that said payer's data received from said payer's telephone operator or from said portable terminal may include the payer's subscriber data or identity parameters or any other required information; and that.

- said subscriber data can be confirmed (i.e. authenticated) and secured either in the databases and infrastructure of the subscriber's telephone operator (4), or in said computing station. for example. by utilizing the algorithms used in mobile communication systems such as those of the GSM; and that.

- after that the subscriber data. communicated between said portable terminal and wireless communication network or directly between said portable terminal and computing station has been authenticated. the Mobile Payment Part (10) of the portable terminal or said computing station can send and/or receives payment etc. messages through SMS of a mobile communication network (4); and that.

- said computing station can monitor the subscriber identity. number etc.. received alternatively from said payer's telephone operator or said portable terminal. and based on said subscriber identity and/or number and checking of his/her bank account number transfer the required amount of payment (money) from said payer's account to any other required account; and that,

- said subscriber data can alternatively be confirmed or sent by the subscriber's telephone operating network (4. 15. 26) to the computing station (2. 14. 24); as a confirmation of subscriber identification. enabling said computing station to compare the received subscriber data with the data already recorded in said computing station. and when subscriber data is compared and accepted by said computing station, the portable terminal can send payment messages to said computing station; and that,

- said subscriber data may include the subscriber telephone number. confirmed by mobile operator (4), or it may consist of the subscriber identity incorporated in SIM card, or any other code; and that.

- said computing station (2, 14, 24) can send or receive payment messages (e.g. 13, 18, 19, 25, 29, 30, 33, 34, 35) to or from the portable terminals (1, 6, 8, 41, 37) of both the payer and the payee; and that,

- said computing station (2, 14, 24) is equipped with all means for wired or wireless transmission and reception of messages communicated between said computing station (2), wireless communications network (4 or 5), and said portable terminal (1, 6, 8, 41, 37).

- said computing station may send e.g. a monthly report (e.g. 35) to said portable terminal (1, 6, 37, 41) to be displayed or printed (20, 38), for said subscriber, as a receipt and bank report for payments (bills, etc.) charged from the subscriber/payer account to the other subscriber/payee account, by said computing station.

5. A mobile payment system (43) according to any preceding claims, **characterized** in that the subscriber, for example, a waiter etc. in a restaurant can send the payment messages (e.g. a bill) by using the inventive portable terminal (1, 6, 8) either to the computing station (2, 14) or directly to the customer's portable terminal (e.g. a mobile telephone integrated with the inventive Mobile Payment Part), via SMS facilities of mobile communication network (5, 4), which after the payment can be accepted by said customer and be sent to the computing station (2) in which the payment procedure will be completed and then a message (including the bill's information) will be sent to both customer's and waiter's portable terminals indicating that either the payment has been completed and/or accepted (29, 30) or refused (19); and that:

- said waiter etc. or customer can enter the customer's identity code to said waiter's portable terminal, by using user interface (9), and then send the bill together with the customer's code to the computing station, which after said computing station generates a message and sends it to the customer's portable terminal to be accepted by the customer, and that after that the payment has been completed in the computing station, the computing station can send a message such as "Payment Reception" including all information about the payment (e.g. bill reference, payer, payment amount etc.) to the payee's terminal indicating that the payee has received the payment; and that,

- said portable terminal (1, 6) can be used in order to charge customers, in retail or trading businesses, by entering alternatively the customers' telephone SIM card (39) into said portable terminal's SIM card reader (36) and by using the telephone subscriber identity of each customer as an personal identification for payment; and that:

- after that said customer's SIM card (39) has been entered to said portable terminal (1, 6, 8), said portable terminal will be re-registered to the wireless communications

network (4, 15) and/or said computing station in order to check the subscriber identity, which after the customer's (subscriber's) bank account can be charged by sending payment messages (e.g. 13) to the computing station (2, 14).

6. A mobile payment system (43) according to any preceding claims, **characterized** in that whenever the subscriber turns on his/her portable terminal (1, 6) the Mobile Payment Part (10) sends the subscriber data, that can be included in the SIM card, to the computing station (2, 14) through an available wireless communication network (4, 5), and after that registration process between said computing station, said wireless communication network and said portable terminal (1, 6, 8) has been completed said portable terminal can have access to said wireless communication network through which it can send and/or receive payment, banking etc. messages to/from said computing station, and also be able to use the telecommunications services like voice etc. of said wireless communication network; and that,

- after said portable terminal has been registered in said wireless communication network (4) or computing station (2), the subscriber of said portable terminal can send and/or receive banking messages (e.g. 33, 34, 35) or can pay his/her bills by sending and receiving the payment etc. messages (e.g. 13, 18, 19, 25, 29) to the computing station (2, 14) or to another portable terminal, through the Mobile Payment Part (10) of his/her portable terminal; and that,

- said subscriber data can be a data which is recorded only in the SIM card and in said computing station that is located in the bank; and that,

- said subscriber data can be either similar to or different from that subscriber identity which is incorporated in the subscriber's telephone SIM card provided by mobile operators (4); and that,

- said subscriber data can be alternatively sent to said computer station after that registration of said portable terminal into said wireless communication network (4, 5,) has been completed, which after the subscriber can send and/or receive payment/bill messages to said computing station via SMS of said wireless communication network (4, 5,).

7. A portable terminal (1, 6, 8, 41, 37) according to any preceding claims, **characterized** in that it includes the inventive Mobile Payment Part (10, 21) which for each payment procedure may ask the subscriber (i.e. the payer) to enter all payment information (e.g. 11) such as payee's account number, bill's reference number, bill's due date, the

amount of payment and other information included in the bill or required for payment procedure: and that:

- said Mobile Payment Part (10, 21), after receiving all information about a payment or a bill from the user through user interface (9), may ask the subscriber of said portable terminal e.g. "Any other payment ?" (13) indicating dose subscriber wants to pay another bill or payment, and that after this message subscriber can enter other payment information into said Mobile Payment Part: and that,

- said portable terminal (1, 6, 8, 41) can be used in order to pay the bills of any mobile telephone subscriber by entering each subscriber's identities and codes into said portable terminal either by using the portable terminal's user interface (9) or by entering the SIM card (39) and card reader (36); and that:

- more than one payment or bill etc. data can be entered into said Mobile Payment Part (10, 21) of said portable terminal, and that after that telephone number of said computing station based in the bank (2, 14, 24) has been dialed (12, 23) either manually or automatically, all required payment information (e.g. 13, 25) will be sent to said computing station via SMS facilities of the mobile network (4, 5, 15, 26); and that,

- said portable terminal (1, 6, 8, 37, 41), includes all means of a mobile/cellular/cordless telephone for receiving and transmitting voice and data so that said portable terminal can function both as a mobile payment device and as a mobile/cellular/cordless telephone without requiring any data modem to be used in conjunction with the transmission and reception of payment etc. messages; and that,

- said Mobile Payment Part (10, 21) can be integrated into any kind of portable telephone that is capable of operating in cellular communications systems; and that,

8. A portable terminal (1, 6, 8, 41, 37) according to any preceding claims, **characterized** in that a small printing device (20, 38) is integrated into said portable terminal (1, 6, 37, 41) for printing any data received from computing station (2) or other portable terminals or any other source or the messages entered into said Mobile Payment Part (10, 11, 21) by its user or any other short messages received by said portable terminal.

9. A Mobile Payment Part (10) according to any preceding claims, **characterized** in that it is a component and function integrated into the portable terminal (1, 6), said Mobile Payment Part provides a payment (bill) inquiry (11) procedure including for example questions (such as payment amount, Bill reference, Receiver's account number, Due date, Recipient etc.) which can be displayed on the display (6) and which can be answered by the

user of the portable terminal through the user interface (9) and such payment information can be saved into the memory of the portable terminal or be sent to the computing station (2) or another portable terminal via SMS; and that

- said Mobile Payment Part (10) can be either integrated into said portable terminal as a component including the required soft-ware for providing said bill inquiry (e.g. 11), or said Mobile Payment Part can be integrated into the SIM card (i.e. Subscriber Identity Module) to provide said bill inquiry whenever subscriber wants to pay a bill or perform a payment; and that,

- said Mobile Payment Part is a function and SMS-based adaptation, integrated into said portable terminal or alternatively into said SIM card to provide an alphanumeric payment (bill) inquiry (11) procedure; and that,

- said Mobile Payment Part (10) can divide and split any long data of any length, for example e-mails done in a personal computer etc. which can be connected to said portable terminal into several short messages and send them to other portable terminals/telephones (e.g. 1 or 6) or to said computing station (2) via SMS facilities of a wireless communication network (e.g. 4 or 5) without requiring any data modem to be connected between the portable terminal and said personal computer, so that said Mobile Payment Part divides such e-mail to several short messages in a numbering sequence, for example, first short message, second short message etc.; and that,

- said Mobile Payment Part (10) is able to connect several short messages originated from a long data of any length e.g. an e-mail according to said short messages' numbers defined in the sender's portable terminal (e.g. 1) and their sender's identity (e.g. subscriber number), and put them into the original order and configure said original long data, which can be a long information sent by computing station or another portable terminal (e.g. 6) or any other source equipped with the inventive Mobile Payment Part (10) via SMS facilities of a mobile communication system (4), and then display said original data (e.g. the e-mail) on the display of the portable terminal (1, 6) or forward it to a separate monitor or personal computer without requiring any data modem to be used between said portable terminal and said personal computer; and that,

- all short messages which are resulted from a longer data and received by said portable terminal (1 or 6) or computing station (2) may contain a short message number which is unique for each message and is defined according to their dividing sequence; and that,

- all short messages which are divided from a longer data and received by said portable terminal (1 or 6) or computing station (2) , through SMS infrastructure (4 or 5),

may contain both the sender's and receiver's identity number (e.g. payer's and payee's subscriber numbers), which can be added to each short message either at said message sending portable terminal (e.g. 1) or at the wireless communication network's SMS facilities (4); and that,

- all short messages which are divided from a longer data and received by said portable terminal (1 or 6) or computing station (2), through SMS infrastructure (4 or 5), may be connected according to their sender's identity and their arrival time to the SMS facilities of a mobile communication system (4) or their sending time from the portable terminal or computing station or any other source; and that such sending or arrival time can be defined either at said portable terminal (e.g. 1), which sends the messages, or at the SMS facilities of the wireless communication network (4).

10. A mobile payment system (43) according to any preceding claims, **characterized** in that the telephone calls made by portable terminal (1, 6) can be charged simultaneously after each or several calls, from the subscriber's bank account (i.e. payer's account) to the wireless communication operator's (4 or 5) account, so that said operator can send the bills relevant to the telecommunications services used by said subscriber, directly to said computing station (2); and that,

- said computing station can include either the subscribers' data and bank account information or both the subscribers' and said wireless communication operator's data and bank account information so that the subscribers' all telephone calls can be charged directly from the subscriber's account to said wireless communication operator's bank account, by said computing station; and that,

- said computing station may send e.g. a monthly report (e.g. 35) to said portable terminal (1, 6, 37, 41) to be displayed or printed (20, 38), for said subscriber, as a receipt against charged calls or any telecommunications services used by said subscriber and charged from said subscriber bank account to the wireless communication operator's (4 or 5) bank account, by said computing station.

STATEMENT UNDER ARTICLE 19

Hereby we would like to file and publish the attached Amendment together with the above application. The claims filed are amended in order to better define the scope of the claims for the purposes of provisional protection. All claims are amended after that International Searching Report was received by the applicant so that the amended claims define the scope of the claims mainly based on using the second alternative (i.e. Short Message Services facilities. see page 7 of description). Moreover. it was noticed that the filed claims could not cover all objects of the above-mentioned application without applying for amendment. All claims amended here fall into the description of the invention. and go not beyond the disclosure in the above international application as filed. The differences between the claims as filed and as amended are indicated in the next page.

FIG 2

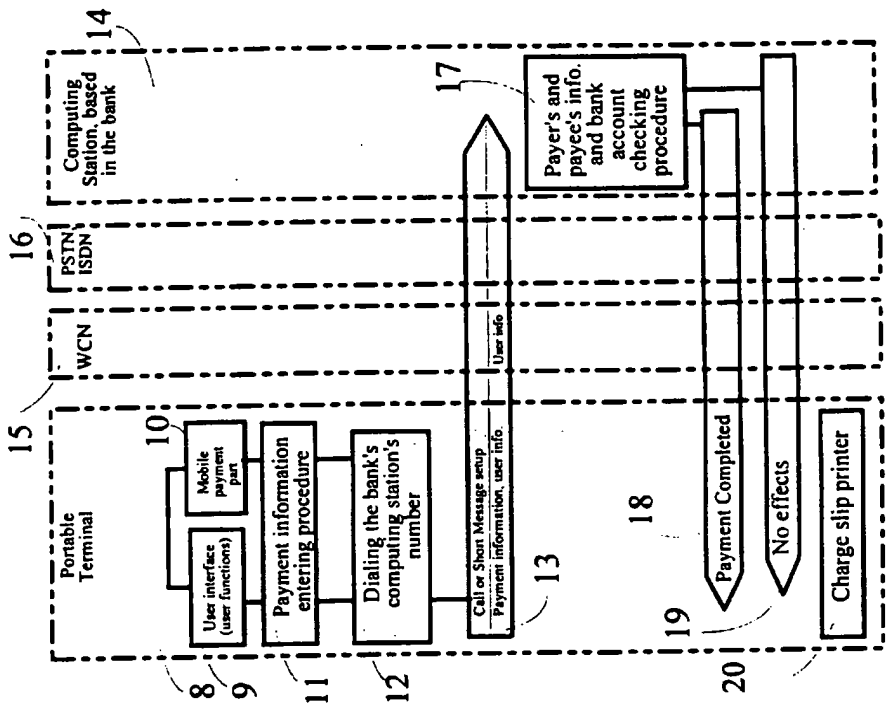

Computing Station, based in the bank
14
PSTN ISDN
16
WCN
15
Portable Terminal
10
Mobile payment part
User interface (user functions)
8
9
Payment information entering procedure
11
Dialing the bank's computing station's number
12
Call or Short Message setup
Payment information, user info.
User info
13
17
Payer's and payee's info. and bank account checking procedure
18
Payment Completed
19
No effects
20
Charge slip printer

FIG 1

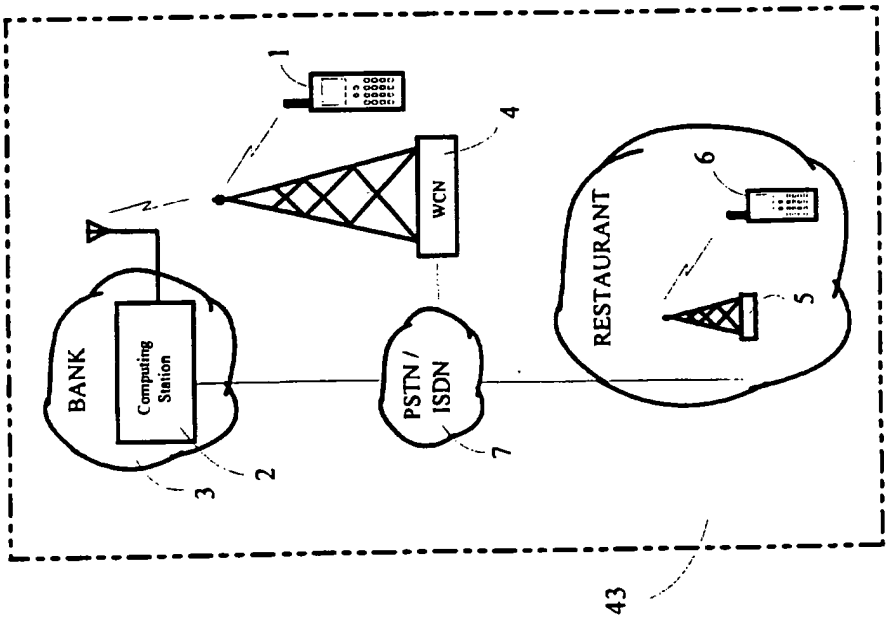

1
4
WCN
BANK
Computing Station
3
2
PSTN / ISDN
7
RESTAURANT
6
5
43

FIG 4

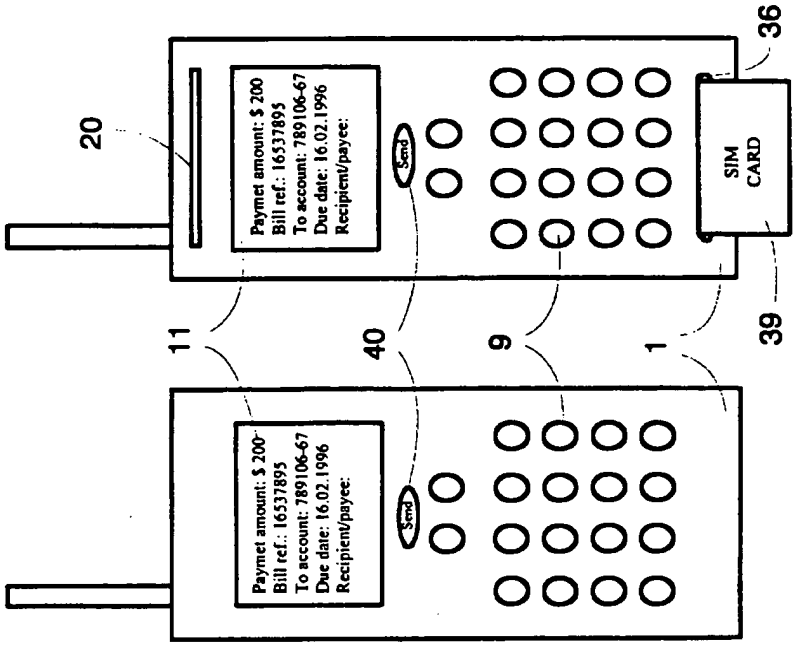
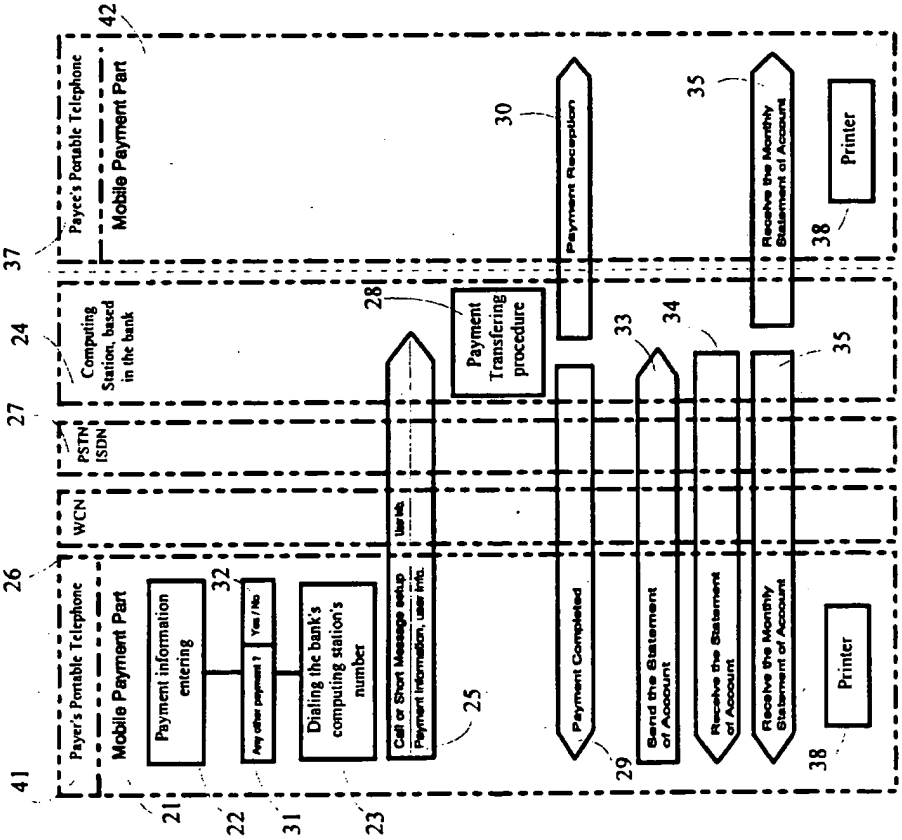

Paymet amount: $ 200
Bill ref.: 16537895
To account: 789106-67
Due date: 16.02.1996
Recipient/payee:
Send
SIM CARD
20
36
11
40
9
1
39

FIG 3


Payer's Portable Telephone
Mobile Payment Part
Payment information entering
Any other payment ?
Yes / No
Dialing the bank's computing station's number
Call or Short Message setup
Payment information, user info.
Payment Transfering procedure
Payment Completed
Send the Statement of Account
Receive the Statement of Account
Receive the Monthly Statement of Account
Printer
Payment Reception
Receive the Monthly Statement of Account
Printer
WCN
PSTN ISDN
Computing Station, based in the bank
Payer's Portable Telephone
Mobile Payment Part
41
21
22
31
23
26
32
25
27
24
28
29
30
33
34
35
35
37
38
38
42

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 95/00591

A. CLASSIFICATION OF SUBJECT MATTER

IPC6: G07F 7/08, G07F 19/00, G06F 17/60 // G06F 157:00
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC6: G07F, H04M, H04Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 9411849 A1 (VATANEN, H.T.), 26 May 1994 (26.05.94) -- --------	1-8

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" erlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
1 March 1996	04-03-1996
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Jan Silfverling Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

05/02/96

International application No.
PCT/FI 95/00591

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO-A1- 9411849	26/05/94	NONE	

16) Family number: 12389386 (JP11031130 A2) | full-text | status | citations | < | > | ^ |

Title: SERVICE PROVIDING DEVICE

Priority: JP19970184866 19970710
Priority Map

Family: Family Explorer	Publication number	Publication date	Application number	Application date	Link
	JP11031130 A2	19990202	JP19970184866	19970710	

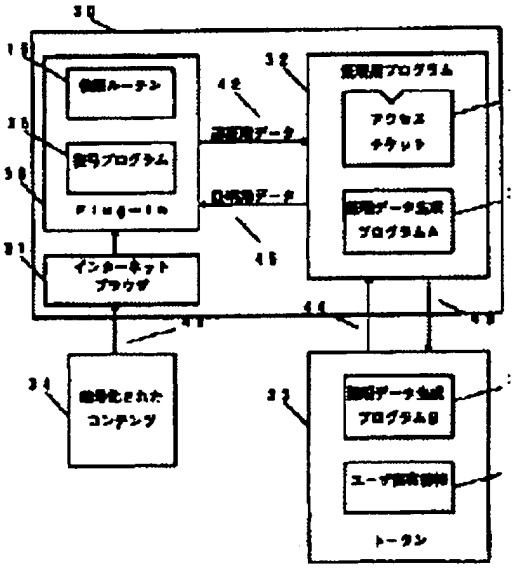
Assignee(s): FUJI XEROX CO LTD

Inventor(s): KOJIMA SHUNICHI ; KONO KENJI ; NAKAGAKI JUHEI

International class (IPC 8): G06F15/00 G09C1/00 H04L9/32 (Advanced/Invention); G06F15/00 G09C1/00 H04L9/32 (Core/Invention)

International class (IPC 1-7): G06F15/00 G09C1/00 H04L9/32

Abstract:
Source: JP11031130A2 PROBLEM TO BE SOLVED: To provide the utilization of service only to a user who has a legal right, minimizing the burden on the user and a service provider. SOLUTION: When a plug-in 38 of an internet browser 31 is started, a verification program 15 in the plug-in 38 is started, communicates with a program 32 for certification and performs user authentication. A certification data generation program A36 of the program 32 cooperates with a certification data generation program B37 in a token 33, calculates based on a user inherent information 16 and an access ticket 13 and communicates with the program 15 in the plug-in 38 based on the calculation. As the result of the communication, the success of authentication by the program 15 is limited to only when the three of the user inherent information, the access ticket and enciphered contents correctly correspond with one another.



8) Family number: 14153892 (JP2000215165 A2) | full-text | status | citations | < | > | ^ |

Title: METHOD AND DEVICE FOR INFORMATION ACCESS CONTROL AND RECORD MEDIUM RECORDII INFORMATION ACCESS CONTROL PROGRAM

Priority: JP19990017401 19990126
Priority Map

Family:	Publication number	Publication date	Application number	Application date	Link
Family Explorer	JP2000215165 A2	20000804	JP19990017401	19990126	

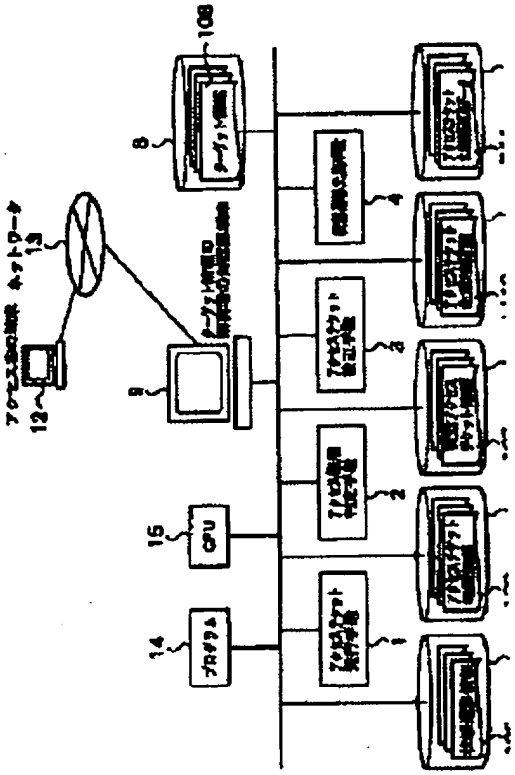
Assignee(s) (std): NIPPON TELEGRAPH AND TELEPHONE

Inventor(s): OHARA YASUHIRO ; OSHIMA YOSHITO

International class (IPC 8): G06F12/14 G06F15/00 G09C1/00 H04L9/32 (Advanced/Invention); G06F12/14 G06F15/00 G09C1/00 H04L9/32 (Core/Invention)

International class (IPC 1-7): G06F12/14 G06F15/00 G06F17/60 G09C1/00 H04L9/32

Abstract:
Source: JP2000215165A2 PROBLEM TO BE SOLVED: To provide the method and device for information access control which can easily change the access authority to be allowed to an accessing person in response to the change of situation of a transaction and also to provide a recording medium which records an information access control program. SOLUTION: An access ticket issuing means 1 issues the access tickets to every accessing person and these tickets prescribe the access authority to the target information for each of plural types and states. Receiving an access request from an accessing person, the means 1 reads the request and the access authority corresponding to the type and state of an inputted access ticket out of an access ticket authority information storing means 6 and decides to permit or not permit the access request based on the access authority. When a state transition request is received from the accessing person, the transition destination state is read out of a state transition information storing means 5 based on the type and state of the access ticket that is inputted together with the state transition request. Based on the transition destination state, the change of the access ticket is updated.



17) Family number: 12393236 (JP11032037 A2) | full-text | status | citations | < | > | ^ |

Title: CERTIFICATION DATA GENERATING DEVICE

Priority: JP19970188801 19970714
Priority Map

Family: Family Explorer	Publication number	Publication date	Application number	Application date	Link
	JP11032037 A2	19990202	JP19970188801	19970714	
	JP3641909 B2	20050427	JP19970188801	19970714	

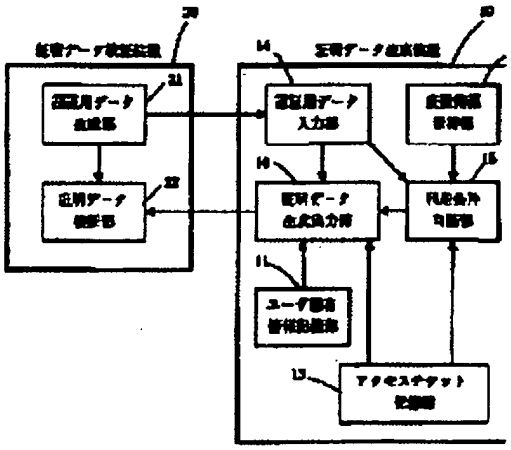
Assignee(s): FUJI XEROX CO LTD

Inventor(s): NAKAGAKI JUHEI ; SHIN YOSHIHIRO

International class (IPC 8): G06F15/00 G06F9/06 G09C1/00 H04L9/32 (Advanced/Invention); G06F15/00 G06F9/06 G09C1/00 H04L9/32 (Core/Invention)

International class (IPC 1-7): G06F15/00 G06F9/06 G09C1/00 H04L9/32

Abstract:
Source: JP11032037A2 PROBLEM TO BE SOLVED: To pre-pay access qualification to purchase or rent without imposing any surplus load on a certification data generating device side. SOLUTION: A pre-paid purchase ticket T_2 is stored in an access ticket storing part 13. Next, (T_1', n_2) is inputted to a certification data-imputting part 14. A use condition judging part 15 extracts a corresponding access ticket (t_2, L_2, n_2), checks whether or not a use condition L_2 is fulfilled, and reduces frequency information V, when the use condition is fulfilled. A certification data generating and outputting part 16 calculates certification data R by using auxiliary certification decision $(t)_2$ and the use condition L_2 extracted by the use condition decision part 15 and (du) read from a user specific information storing part 11, and outputs T_1. A user performs access to a program in a purchase state or a rent state by using the T_1.



12) Family number: 13081077 (JP11205306 A2) | ✉ | 🖶 | full-text | status | citations | < | > | ^ |

Title: AUTHENTICATION SYSTEM AND AUTHENTICATION METHOD

Priority: JP19980006267 19980116
Priority Map

Family:	Publication number	Publication date	Application number	Application date	Link
Family Explorer	JP11205306 A2	19990730	JP19980006267	19980116	

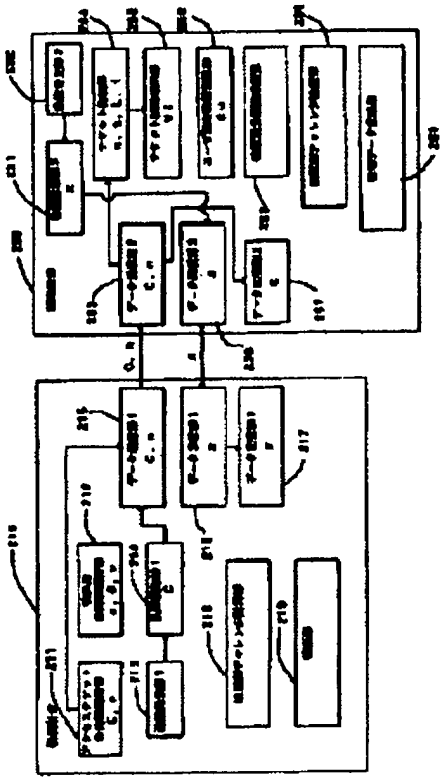
Assignee(s): FUJI XEROX CO LTD

Inventor(s): KOJIMA SHUNICHI ; KONO KENJI ; TAGUCHI MASAHIRO ; TERAO TARO

International class (IPC 8): G09C1/00 H04L9/32 (Advanced/Invention); G09C1/00 H04L9/32 (Core/Invention)

International class (IPC 1-7): G09C1/00 H04L9/32

Abstract:
Source: JP11205306A2 PROBLEM TO BE SOLVED: To provide a system and method that realize diversified services by using an access ticket generated from characteristics information not belonging to a person and information specific to the user, as for the authentification system that authenticates legality of the user. SOLUTION: The authentication device 210 sends authentication data and a ticket identifier to an authentication device 250, the authentification device 250 sends the authentication data to the authentication device 210, which calculates an authentication challenge (ρ) based on a ticket attribute revision request (μ) and an authentication device authentification data (χ). The authentification device 250 receives the ρ and the (&epsi , v) to authenticate an authentication device open key based on input data and an authentication device open key identifier (v '), to authenticate the authentification device challenge and to revise contents of a ticket attribute record (VI) depending on the ticket attribute revision request (μ). Furthermore, an authentification data generating section calculates a response (R) and the authentication device authenticates the legality of the response (R).



11) Family number: 13107360 (JP11215121 A2)

| full-text | status | citations | | > |

Title: DEVICE AND METHOD FOR AUTHENTICATION

Priority: JP19980016710 19980129
Priority Map

Family: Family Explorer	Publication number	Publication date	Application number	Application date	Link
	JP11215121 A2	19990806	JP19980016710	19980129	
	JP3791169 B2	20060628	JP19980016710	19980129	

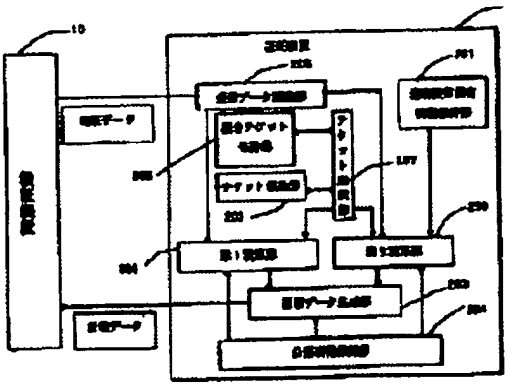
Assignee(s): FUJI XEROX CO LTD

Inventor(s): KIKO KENICHIROU

International class (IPC 8): G09C1/00 H04L9/32 (Advanced/Invention); G09C1/00 H04L9/32 (Core/Invention)

International class (IPC 1-7): G09C1/00 H04L9/32

Abstract:
Source: JP11215121A2 PROBLEM TO BE SOLVED: To perform composite authentication by using the combination of different kinds of issued tickets. SOLUTION: The ticket holding section 202 of a certifying device 20 holds a ticket indicating the specific right of a user while a composite ticket holding section 206 holds a composite ticket for certifying that the user holds a plurality of other effective tickets. A certifying data generating section 203 certifies the presence of a compositely designated right by generating certifying data through executing a prescribed operation by the use of a prescribed access ticket, a composite ticket, and inherent information of the certifying device to authentication information sent from a verifying device 10.



2) Family number: 33529418 (JP2005218143 A2)
extended family

full-text | status | citations | < | > | ^ |

Title: ENCRYPTION DEVICE USED IN A CONDITIONAL ACCESS SYSTEM

Priority: US19970054575P 19970801
Priority Map

Family: Family Explorer	Publication number	Publication date	Application number	Application date	Link
	JP2005218143 A2	20050811	JP20050120426	20050418	
	WO9907150 A1	19990211	WO1998US16145	19980731	

Assignee(s) (std): SCIENTIFIC ATLANTA

Assignee(s): SCIENTIFIC ATLANTA INC

Inventor(s) (std): PALGON MICHAEL S ; PINDER HOWARD G

Designated states: AL AM AT AU AZ BA BB BE BF BG BJ BR BY CA CF CG CH CI CM CN CU CY CZ DE DK EE ES FI F GA GB GE GH GM GN GR GW HR HU ID IE IL IS IT JP KE KG KP KR KZ LC LK LR LS LT LU LV M MD MG MK ML MN MR MW MX NE NL NO NZ PL PT RO RU SD SE SG SI SK SL SN SZ TD TG TJ TR TT UA UG UZ VN YU ZW

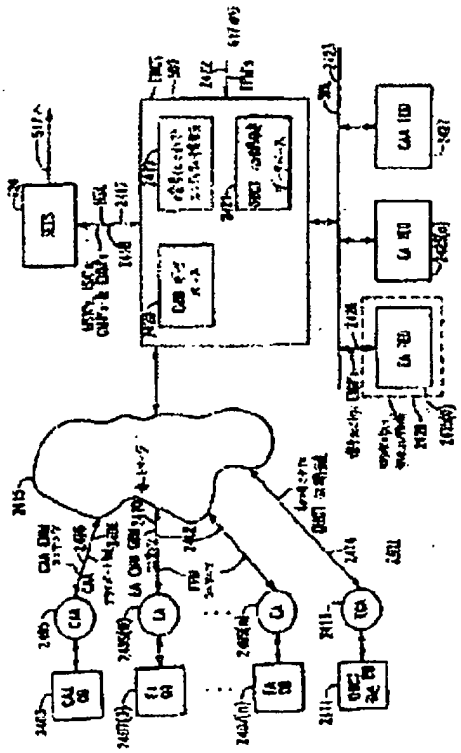
International class (IPC 8): G09C1/00 H04L9/08 H04L9/10 H04N7/10 H04N7/16 H04N7/167 (Advanced/Invention); G09C1/00 H04L9/08 H04L9/10 H04N7/10 H04N7/16 H04N7/167 (Core/Invention)

International class (IPC 1-7): H04L9/10 H04N7/16 H04N7/167

European class: H04N7/167D H04N7/16E2

Cited documents: WO9529560, US5787172, US5592552, US5400401, US5341425, EP0752786,

Abstract:
Source: JP2005218143A2 PROBLEM TO BE SOLVED: To provide a cable television system providing conditional access to a service. SOLUTION: The cable television system includes a headend from which service "instances", or programs, are broadcast and a plurality of set top units for receiving the instances and selectively decrypting these instances for display to system subscribers. The service instances are encrypted, by using public and/or private keys provided by service providers or central authorization agents. Keys, used by the set tops for selective decryption may also be public or private in nature, and these keys may be reassigned at different times, to provide a cable television system in which the anxiety for violation actions is minimized. COPYRIGHT: (C)2005,JPO&NCIPI<



4) Family number: 33529421 (JP2005253109 A2)
extended family

| | full-text | status | citations | < | > | ^ |

Title: CONDITIONAL ACCESS SYSTEM

Priority: US19970054575P 19970801 US19980126921 19980731
Priority Map

Family: Family Explorer

Publication number	Publication date	Application number	Application date	Link
JP2005253109 A2	20050915	JP20050120425	20050418	
WO9909743 A2	19990225	WO1998US16079	19980731	
WO9909743 A3	19990527	WO1998US16079	19980731	

Assignee(s) (std): SCIENTIFIC ATLANTA

Assignee(s): SCIENTIFIC ATLANTA INC

Inventor(s) (std): AKINS GLENDON L III ; PALGON MICHAEL S ; PINDER HOWARD G ; WASILEWSKI ANTHONY J

Inventor(s): AKINS GLENDON L

Designated states: AL AM AT AU AZ BA BB BE BF BG BJ BR BY CA CF CG CH CI CM CN CU CY CZ DE DK EE ES FI F GA GB GE GH GM GN GR GW HR HU ID IE IL IS IT JP KE KG KP KR KZ LC LK LR LS LT LU LV M MD MG MK ML MN MR MW MX NE NL NO NZ PL PT RO RU SD SE SG SI SK SL SN SZ TD TG TJ TR TT UA UG UZ VN YU ZW

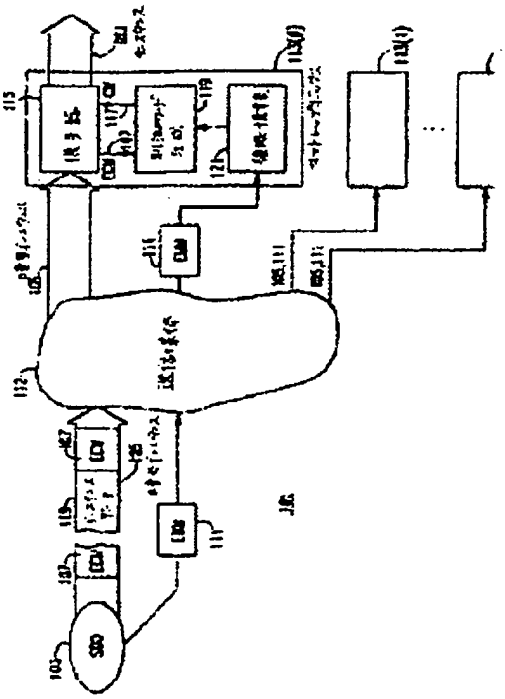
International class (IPC 8): H04H1/00 H04L9/08 H04N5/00 H04N7/16 H04N7/167 H04N7/173 (Advanced/Invention); H04H1/00 H04L9/08 H04N5/00 H04N7/16 H04N7/167 H04N7/173 (Core/Invention)

International class (IPC 1-7): H04L9/08 H04N7/167

European class: H04N5/00M4 H04N7/167D H04N7/16E2

Cited documents: WO9704553, US5381481, US5029207, US4887296, US4864615, US4736422, US4613901,

Abstract:
Source: JP2005253109A2 PROBLEM TO BE SOLVED: To provide a cable television system which provides conditional access to services. SOLUTION: This cable television system includes a headend from which service "instances" or programs are broadcast and a plurality of set top units for receiving the instances and selectively decrypting the instances for display to system subscribers. The service instances are encrypted using public keys and/or private keys provided by service providers or central authorization agents. Keys used by the set tops for a selective decryption may also be public or private in nature, and such keys may be reassigned at different times to provide a cable television system in which piracy concerns are minimized. COPYRIGHT: (C)2005,JPO&NCIPI<



1/9/1
DIALOG(R)File 347: JAPIO

08787202 **Image available**
CRYPTOGRAPHIC KEY SYSTEM

Pub. No.: 2006-180562 [**JP 2006180562** A]
Published: July 06, 2006 (20060706)
Inventor: SAITO MAKOTO
MOMIKI JUNICHI
Applicant: INTARSIA SOFTWARE LLC
Application No.: 2006-082675 [JP 200682675]
Division of 07-346095 [JP 95346095]
Filed: March 24, 2006 (20060324)
Priority: 06-309292 [JP 94309292], JP (Japan), December 13, 1994 (19941213)

International Patent Class (v8 + Attributes)
IPC + Level Value Position Status Version Action Source Office:

H04L-0009/08 A I F B 20060101 20060609 H JP

ABSTRACT

PROBLEM TO BE SOLVED: To provide a concrete structure for applying a cryptographic key system to a television system, a database system or an electronic commercial transaction system or the like.

SOLUTION: This system consists of a broadcasting station, a database, a receiving apparatus, a data communications apparatus and a user terminal. As a cryptographic key system, a secret-key system, a public-key system, and a digital signature system are used. The keys used in the system are either encrypted, or remain uncrypted to be supplied by broadcasting. The system is effective in preventing the unauthorized use of the database system, managing copyrights, and in pay-per-view systems and video-on-demand systems. Further, the system is effective in realizing an electronic market which uses an electronic data information system.

Divertible Protocols and Atomic Proxy Cryptography

Matt Blaze Gerrit Bleumer Martin Strauss

AT&T Labs - Research
Florham Park, NJ 07932 USA
{mab,bleumer,mstrauss}@research.att.com

Abstract. First, we introduce the notion of divertibility as a protocol property as opposed to the existing notion as a language property (see Okamoto, Ohta [OO90]). We give a definition of protocol divertibility that applies to arbitrary 2-party protocols and is compatible with Okamoto and Ohta's definition in the case of interactive zero-knowledge proofs. Other important examples falling under the new definition are blind signature protocols. We propose a sufficiency criterion for divertibility that is satisfied by many existing protocols and which, surprisingly, generalizes to cover several protocols not normally associated with divertibility (e.g., Diffie-Hellman key exchange). Next, we introduce *atomic proxy cryptography,* in which an *atomic proxy function,* in conjunction with a public *proxy key,* converts ciphertexts (messages or signatures) for one key into ciphertexts for another. Proxy keys, once generated, may be made public and proxy functions applied in untrusted environments. We present atomic proxy functions for discrete-log-based encryption, identification, and signature schemes. It is not clear whether atomic proxy functions exist in general for all public-key cryptosystems. Finally, we discuss the relationship between divertibility and proxy cryptography.

1 Introduction

This paper investigates two general ways in which an intermediary sitting between the participants of a 2-party protocol might transform the communication messages without "destroying" the protocol. First, we consider *protocol divertibility,* in which the (honest) intermediary, called a *warden,* randomizes all messages so that the intended underlying protocol succeeds, but information contained in subtle deviations from the protocol (for example, information coded into the values of supposedly random challenges) will be obliterated by the warden's transformation. Next, we introduce *atomic proxy cryptography,* in which two parties publish a *proxy key* that allows an untrusted intermediary to convert ciphertexts encrypted for the first party directly into ciphertexts that can be decrypted by the second. The intermediary learns neither cleartext nor secret keys.

Our paper is organized as follows. In Section 2 we discuss divertible protocols. In Section 2.1 we define protocol divertibility. We propose a slightly stricter

definition than the original one by Okamoto and Ohta [OO90]. In Section 2.2, we present a sufficiency criterion for divertibility. Its usefulness is demonstrated by many examples of known diverted protocols from the literature. Also many known blind signature protocols can be interpreted as diverted proofs of knowledge and in this form they satisfy our criterion (see [Bleu97]). In Section 3, we introduce atomic proxy cryptography and propose a taxonomy for proxy schemes. In Sections 3.1 to 3.3 we give proxy schemes for encryption, identification, and signature. In Section 4, we discuss the deeper relationship between protocol divertibility and proxy cryptography.

2 Divertible Protocols

The idea of divertibility entered the cryptographic literature during the mid 80's with applications to identification protocols. The basic observation was that some 2-party identification protocols could be extended by placing an intermediary—called a warden for historical reasons [Sim84]—between the prover and verifier so that, even if both parties conspire, they cannot distinguish talking to each other through the warden from talking directly to a hypothetical honest verifier and honest prover, respectively. Since identification protocols were developed in close relation to interactive zero-knowledge proofs (ZKP), Okamoto and Ohta [OO90] (and later Desmedt and Burmester [BD91] and Ihto et al [ISS91]) established the notion of divertibility as a *language property*, i.e., a language is considered divertible if it can be recognized by a diverted interactive zero-knowledge proof system. In this paper, we establish divertibility as a *2-party protocol property*, which is orthogonal to zero knowledge or any other particular protocol property.

2.1 Definitions

In order to deal with protocols of more than two parties, we generalize the notion of *interactive Turing machine* (ITM) by Goldwasser et al [GMR89]. Then we define connections of ITMs and finally give the definition of protocol divertibility.

Definition 1 ((m,n)**-Interactive Turing Machine).**
An (m,n)*-Interactive Turing Machine* ((m,n)*-ITM*) is a Turing machine with $m \in \mathbb{N}$ read-only *input tapes*, m write-only *output tapes*, m read-only *random tapes*, a *work tape*, a read-only *auxiliary tape*, and $n \in \mathbb{N}_0$ pairs of *communication tapes*. Each pair consists of one read-only and one write-only tape that serves for reading in-messages from or writing out-messages to another ITM. (The purpose of allowing $n = 0$ will become clear below.) The random tapes each contain an infinite stream of bits chosen uniformly at random. Read-only tapes are readable only from left to right. If the string to the right of a read-only head is empty, then we say the tape is *empty*.

Associated to an ITM is a *security parameter* $k \in \mathbb{N}$, a family $D = \{D_\pi\}_\pi$ of tuples of domains, a probabilistic *picking algorithm* $pick(k)$ and an encoding

scheme S. Each member

$$D_\pi = (In_\pi^{(1)}, \ldots, In_\pi^{(m)}, Out_\pi^{(1)}, \ldots, Out_\pi^{(m)}, \Omega_\pi^{(1)}, \ldots, \Omega_\pi^{(m)}, \\ (IM_\pi^{(1)}, OM_\pi^{(1)}), \ldots, (IM_\pi^{(n)}, OM_\pi^{(n)}))$$

of D contains one input (output, choice, in-message, out-message) domain for each of the m input (output, random) tapes and n (read-only, write-only) communication tapes. The algorithm $pick(k)$ on input some security parameter k outputs a family index π. Finally, there is a polynomial $P(k)$ so that for each π chosen by $pick(k)$, S encodes all elements of all domains in D_π as bitstrings of length $P(k)$.

ITMs proceed in rounds. During each round, an ITM first reads the messages from all its read-only communication tapes, then performs some computations and finally writes a message to each of its write-only communication tapes. It may write an empty string—denoted ε. If, at the beginning of a round, an ITM finds all its input tapes and all its read-only communication tapes empty, then it performs a last computation, writes empty strings to all its write-only communication tapes, writes results to all its output tapes, and then stops. The overall number of reading, writing and computation steps during an execution of an ITM is bound by a polynomial in the security parameter k.

An (m, n)-ITM is an *m-party protocol* if $n = 0$, and linear if $n \leq 2$. The *native functions* of an ITM A are defined as the family

$$nativ_\pi : \prod_{i=1}^m \Omega_{\pi,i} \times \prod_{i=1}^m In_{\pi,i} \times \prod_{j=1}^n IM_{\pi,j} \to \prod_{j=1}^n OM_{\pi,j}$$

of functions that, on input (rnd, in, im), return the respective out-messages that A would write to its write-only communication tapes would it read this data from its random, input and read-only communication tapes.

Let A be an (m_A, n)-ITM and B be an (m_B, n)-ITM, which together make up a protocol $P = \langle A, B \rangle$. Let $m^* \leq \min(m_A, m_B)$ be the number of pairs of communication tapes shared by A and B. Then the *view* of A on B on respective inputs, denoted as,

$$view_B^{(A)} P([in_{A,1}, \ldots, in_{A,m_A}]^A, [in_{B,1}, \ldots, in_{B,m_B}]^B) \ ,$$

is defined as everything that A sees from B, i.e., the probability distribution of all m^*-tuples of pairs of in-messages sent by A to B and out-messages returned from B to A, where the probabilities are taken over the choices of the viewer A.[1]

For m-party protocols P, we adopt the following interface notation:

$$(out_1, \ldots, out_m) \leftarrow P(in_1, \ldots, in_m) \ ,$$

where the left arrow indicates a probabilistic assignment. If the inputs or outputs consist of several components, we delimit them by square brackets.

[1] This is a generalization of the definition given by Goldwasser, Micali and Rackoff [GMR89].

Definition 2 (Connections of ITMs).
Let A be an (m_A, n_A)-ITM and B be an (m_B, n_B)-ITM with equal picking algorithm *pick*. Then a connection $C = \langle A, B \rangle$ is any ITM consisting of A and B sharing $c \leq \min\{n_A, n_B\}$ pairs of their communication tapes. The picking algorithm of C is *pick*, and the domains of C are defined as the cartesian products of the respective domains of A and B. $\diamond$

Obviously, the linear connection operator $\langle \bullet, \bullet \rangle$ is associative and we can therefore omit brackets in the usual way:

$$\langle A, B, C \rangle \stackrel{\text{def}}{=} \langle \langle A, B \rangle, C \rangle = \langle A, \langle B, C \rangle \rangle \ .$$

All connections we consider in the following are linear and have a small constant number of rounds.

Definition 3 (Divertibility of Protocols).
Let $P = \langle A, B \rangle$ be a two-party protocol with interface $P([y, x_A]^A, [y, x_B]^B)$ and input domains $In_\pi = (Y_\pi \times X_{A,\pi}) \times (Y_\pi \times X_{B,\pi})$. Common inputs y are taken from Y_π, whereas private inputs x_A, x_B are taken from $X_{A,\pi}$ and $X_{B,\pi}$, respectively. The product domain of private inputs is denoted $X_\pi = X_{A,\pi} \times X_{B,\pi}$. Furthermore, let $R = \{R_\pi\}_\pi$ be a family of relations $R_\pi \subseteq Y_\pi \times X_\pi$.

The protocol P is called *perfectly (computationally) divertible* over R iff a (1,2)-ITM W exists such that the following properties hold:

EXTENSIBILITY: For all indices π, all common and private inputs $(y, x_A, x_B) \in R_\pi$, the ensembles of views of B on W and on A, i.e.,

$$view_W^{(B)}\langle A, W, B \rangle([y, x_A]^A, [y]^W, [y, x_B]^B), \text{ and}$$
$$view_A^{(B)}\langle A, B \rangle([y, x_A]^A, [y, x_B]^B)$$

as well as the views of A on W and on B, i.e.,

$$view_W^{(A)}\langle A, W, B \rangle([y, x_A]^A, [y]^W, [y, x_B]^B), \text{ and}$$
$$view_B^{(A)}\langle A, B \rangle([y, x_A]^A, [y, x_B]^B)$$

are equal (polynomially indistinguishable).

PERFECT (COMPUTATIONAL) INDISTINGUISHABILITY: For all polynomial-time actively adversary ITMs $\tilde{A}$, $\tilde{B}$, for all indices π, all common and private inputs $(y, x_A, x_B) \in R_\pi$ and all polynomial size strings q representing shared a priori knowledge of $\tilde{A}$ and $\tilde{B}$, the ensembles of simultaneous views of $\tilde{A}$ and $\tilde{B}$ upon W and of their views upon honest B and A, i.e.,

$$view_W^{(\tilde{A},\tilde{B})}\langle \tilde{A}, W, \tilde{B} \rangle([y, x_A, q]^{\tilde{A}}, [y]^W, [y, x_B, q]^{\tilde{B}}) \quad \text{and}$$
$$(view_B^{(\tilde{A})}\langle \tilde{A}, B \rangle([y, x_A, q]^{\tilde{A}}, [y, x_B]^B), view_A^{(\tilde{B})}\langle A, \tilde{B} \rangle([y, x_A]^A, [y, x_B, q]^{\tilde{B}}))$$

are equal (polynomially indistinguishable).[2] [3]

An ITM W that satisfies extensibility and perfect (computational) indistinguishability is said to *perfectly (computationally) divert* protocol P over R. $\diamond$

Divertibility as defined by Okamoto, Ohta [OO90] and almost equivalently by Itoh et al [ISS91] has been introduced as a *language property.* A language L is considered divertible, if there exists a diverted zero knowledge proof system for proving membership in L. In contrast, we define divertibility as a *2-party protocol property.* The main difference between the two definitions is that we ask for a concrete protocol P to be divertible, whereas they ask for existence of a divertible protocol meeting a certain specification S (namely to be a zero-knowledge proof). Consequently, Definition 3 (extensibility) relates the two interfaces of the diverted protocol P' to the interface of the given protocol P, where their definition relates them to S. Another difference is, that we suggest a stronger definition than Okamoto and Ohta's. We require Indistinguishability even for two attackers $\tilde{A}$ and $\tilde{B}$ who *know of each other* (a-priori common knowledge q) and who therefore know which of their views result from the same diverted protocol instance. We discuss this further in Section 2.4.

An immediate consequence of the definition is that if a protocol P is divertible, then we can insert second and third wardens and we, again, obtain a diverted protocol.

2.2 Main Divertibility Result

Theorem 4 (Criterion for Perfect Divertibility).
Let $P = \langle A, B \rangle$ be a two-party protocol with interface $P([y, x_A]^A, [y, x_B]^B)$. Let the input domains be $(Y_\pi \times X_{A,\pi}) \times (Y_\pi \times X_{B,\pi})$, the random domains be $\Omega_{A,\pi} \times \Omega_{B,\pi}$, the out-message domains be $OM_{A,\pi} \times OM_{B,\pi}$, and let the native functions of A and B be

$$nativ_{A,\pi} : \Omega_{A,\pi} \times Y_\pi \times X_{A,\pi} \times OM_{B,\pi} \rightarrow OM_{A,\pi} \ ,$$
$$nativ_{B,\pi} : \Omega_{B,\pi} \times Y_\pi \times X_{B,\pi} \times OM_{A,\pi} \rightarrow OM_{B,\pi} \ .$$

Furthermore, let $R = \{R_\pi\}_\pi$ be a family of relations $R_\pi \subseteq Y_\pi \times (X_{A,\pi} \times X_{B,\pi})$, which capture the correspondence between the private and the public inputs.

Then P is perfectly divertible over R if only there exist:

(i) a family $(\Omega_\pi, \odot, 1)$ of (not necessarily commutative) groups, and

[2] By $view_B^{(A)} P$, we denote the *view* of A on B in a protocol P. This notion as well as that of *polynomial indistinguishability* of families of random variables is defined, e.g., by Goldwasser, Micali and Rackoff [GMR89].

[3] Equality (polynomial indistinguishability) is required only for the views on *complete* runs of the diverted protocol, i.e., runs that the warden has not aborted, for example, because he has detected either $\tilde{A}$ or $\tilde{B}$ cheating.

(ii) three families of functions

$$base_\pi : Y_\pi \times X_{A,\pi} \times X_{B,\pi} \to OM_{A,\pi} \times OM_{B,\pi} \ ,$$
$$join_\pi : \Omega_{A,\pi} \times \Omega_{B,\pi} \times Y_\pi \times X_{A,\pi} \times X_{B,\pi} \to \Omega_\pi \ ,$$
$$divrt_\pi : \Omega_\pi \times Y_\pi \times OM_{A,\pi} \times OM_{B,\pi} \to OM_{A,\pi} \times OM_{B,\pi} \ ,$$

with the following properties:
Function $divrt(\omega, y, o_A, o_B)$ *is defined only for* (o_A, o_B) *that live in the respective image* $OM_{\pi,y}$ *of* $native_A$ *and* $native_B$*, i.e.,*

$$OM_{\pi,y} = native_A(\Omega_A, y, x_A, o_B) \times native_B(\Omega_B, y, x_B, o_A) \ ,$$

where $(y, x_A, x_B) \in R_\pi$.[4]
Second, for each fixed $\alpha, \beta, y, x_A, x_B \in R_\pi$*, the functions,*

$$join_\pi(\alpha', \beta, y, x_A, x_B) \quad and \quad join_\pi(\alpha, \beta', y, x_A, x_B) \ ,$$

are each bijective on Ω_A *and* Ω_B*, respectively.*

(iii) a warden W *that on input two in-messages* o_A, o_B *computes two out-messages* o'_A, o'_B *such that*

$$(o'_A, o_B) = divrt(\omega, y, (o_A, o'_B)) \ .$$

Now, for every π*, for all random choices* $\alpha \in \Omega_{A,\pi}, \beta \in \Omega_{B,\pi}$*, all common and corresponding private inputs* $(y, x_A, x_B) \in R_\pi$*, and all out-messages* $o_A \in OM_{A,\pi}$*,* $o_B \in OM_{B,\pi}$ *the following three conditions must hold:*

DECOMPOSITION:

$$\begin{aligned}&(nativ_A(\alpha, y, x_A, o_B), nativ_B(\beta, y, x_B, o_A)) \\ &= divrt(join(\alpha, \beta, y, x_A, x_B), y, base(y, x_A, x_B)) \ ,\end{aligned}$$

GROUND:

$$divrt(1, y, (o_A, o_B)) = (o_A, o_B) \ ,$$

MIXED ASSOCIATIVITY:

$$divrt(\omega', y, divrt(\omega, y, (o_A, o_B))) = divrt(\omega \odot \omega', y, (o_A, o_B)) \ .$$

◇

Proof. First observe that if *divrt* satisfies the premises GROUND and MIXED ASSOCIATIVITY, then it is injective as a function of ω: For all $(o_A, o_B) \in OM_{\pi,y}$, we have:

$$\begin{aligned}(o_A, o_B) = \ & divrt(1, y, (o_A, o_B)) \\ & divrt(\omega \odot \omega^{-1}, y, (o_A, o_B)) \ \text{(for any } \omega) \\ & divrt(\omega^{-1}, y, divrt(\omega, y, (o_A, o_B))) \ .\end{aligned}$$

[4] Note that the input variables o_A and o_B in the definition of $OM_{\pi,y}$ refer to the output of $nativ_B$ and $nativ_A$, respectively. This recursion is guaranteed to terminate by the following requirement (iii) below.

So, function *divrt* turns out to be bijective on Ω_π for the entire parameter domain $OM_{\pi,y}$. We may thus write: $divrt^{-1}(\omega, \bullet) = divrt(\omega^{-1}, \bullet)$.

In order to infer extensibility and indistinguishability of P, we look separately at the out-messages between $\langle A, W\rangle$ and B and those out-messages between A and $\langle W, B\rangle$. We deal with the former case in detail and argue that the latter case can be handled analogously due to symmetry reasons. Using DECOMPOSITION and MIXED ASSOCIATIVITY, we rewrite the above mentioned out-messages as follows:

$$\begin{aligned}
&(nativ_{\langle A,W\rangle}(\omega, \alpha, y, x_A, o_B), nativ_B(\beta, y, x_B, o_A)) \\
&= divrt(\omega, y, (nativ_A(\alpha, y, x_A, o'_B), nativ_B(\beta, y, x_B, o'_A))) \\
&= divrt(\omega, y, divrt(\omega', y, base(y, x_A, x_B))), \\
&\qquad \text{where } \omega' = join(\alpha, \beta, y, x_A, x_B) \\
&= divrt(\omega' \odot \omega, y, base(y, x_A, x_B)) \\
&= (nativ_A(\alpha', y, x_A, o_B), nativ_B(\omega, \beta', y, x_B, o_A)) \ , \\
&\qquad \text{where } (\alpha', \beta') = join^{-1}(\omega' \odot \omega, y, x_A, x_B) \ .
\end{aligned}$$

It then follows from the bijectiveness of *join* and the fact that $\odot$ is a group operation that the probability of each pair of out-messages is the same over Bob's choices β and over β'. Together with the analogous result for out-messages between A and $\langle W, B\rangle$ (this is where invertibility of *divrt* is needed), this settles extensibility.

For perfect indistinguishability, we need to deal with arbitrary attackers $\tilde{A}, \tilde{B}$, instead. Assume, these attackers produce their out-messages with a certain distribution D that respects the domain of function *divrt*. Otherwise, *divrt* is undefined and the distribution could be ignored according to indistinguishability. Then by decomposition, we see that this given distribution D can also be achieved by honest Alice and Bob if Bob would chose his β according to some appropriate distribution d. Following the above rewriting, and again taking into account that join is bijective and $\odot$ is a group operation, we conclude, that the distribution of $\omega' \odot \omega$ is d because, by presumption, the warden is honest and therefore ω is uniformly distributed. Hence, the out-messages of $\langle A, W\rangle$ and B are also distributed according to D, if the probabilities are taken over β'. Together with the analogous result for out-messages between A and $\langle W, B\rangle$, this in addition settles perfect indistinguishability and therefore perfect divertibility. □

2.3 New Example of Diverted Protocol

The most prominent examples of diverted protocols in the literature are diverted interactive proofs and blind signatures. Since divertibility has been introduced only in the former context, blind signatures are a good example to illustrate the more general concept of divertibility of protocols as proposed in Definition 3. The practical value of Theorem 4 is demonstrated in [Bleu97] by proving many protocols unconditionally divertible; in particular (i) the diverted ZKP that Okamoto and Ohta used to prove their main theorem [OO90] and (ii) a

blind modified ElGamal Signature, which was presented by Horster, Michels and Petersen [HMP95] who built on ideas of Camenisch, Piveteau and Stadler [CPS95].

Here, we consider a new sort of protocol for divertibility, namely key exchange. In Figure 1, we present a diverted Diffie-Hellman key exchange protocol [DH76]. Let p be a k-bit prime ($k \in \mathbb{N}$), q be a large prime divisor of $p-1$ and G_q be the unique (multiplicative) subgroup of order q in $\mathbb{Z}_p^*$. Furthermore, $g \neq 1$ denotes a randomly chosen element of G_q. (The restriction to $g \neq 1$ asserts that g generates G_q). p, q and g are global system parameters and neither Alice nor Bob have private inputs.

$$([a'']^A, [b'']^B) \leftarrow DDH([]^A, []^W, []^B)$$

	Alice		**Warden**		**Bob**
(1)	$\alpha \in_R \mathbb{Z}_q$		$\omega \in_R \mathbb{Z}_q$		$\beta \in_R \mathbb{Z}_q$
(2)	$a \leftarrow g^\alpha$				$b \leftarrow g^\beta$
(3)		$\xrightarrow{a}$	$(a', b') \leftarrow (a^\omega, b^\omega)$	$\xleftarrow{b}$	
(4)	$a'' \leftarrow b'^\alpha$	$\xleftarrow{b'}$		$\xrightarrow{a'}$	$b'' \leftarrow a'^\beta$

Fig. 1. Diverted Diffie-Hellman Key Exchange

Proposition 5. *The warden of protocol DDH computationally diverts the Diffie-Hellman protocol between Alice and Bob over $R = \emptyset$.* ◇

Proof (Sketch). If for given (a, b), an attacker could distinguish valid from invalid diverted out-messages (a', b') with non-negligible probability, i.e., probability $\geq \frac{1}{P(k)}$ for some polynomial P, then he had broken the simultaneous discrete log assumption [CEG88]. □

2.4 Why the Previous Definition is a Little too Weak

The previous definition of divertibility by Okamoto and Ohta [OO90], and by Itoh et al [ISS91] as well, requires that two attackers $\tilde{A}, \tilde{B}$ who on the one hand form a linear 3-party protocol P' with an intermediate warden and on the other hand form 2-party protocols $\langle \tilde{A}, B \rangle$ with an honest B and $\langle A, \tilde{B} \rangle$ with an honest A cannot distinguish their views in $\langle \tilde{A}, B \rangle$ and $\langle A, \tilde{B} \rangle$ from those in *separate* instances of $\langle \tilde{A}, W, \tilde{B} \rangle$. More formally, they require indistinguishability of the two ensembles (protocol inputs exactly as in Definition 3 before):

$$(view_W^{(\tilde{A})}\langle \tilde{A}, W, \tilde{B} \rangle, view_W^{(\tilde{B})}\langle \tilde{A}, W, \tilde{B} \rangle) \tag{1}$$

$$\text{and} \quad (view_B^{(\tilde{A})}\langle \tilde{A}, B \rangle, view_A^{(\tilde{B})}\langle A, \tilde{B} \rangle). \tag{2}$$

However, the attacker model that seems to underly the literature on divertibility is stronger than expressed by the above requirement. The attackers $\tilde{A}$ and $\tilde{B}$ are usually assumed to know when they engage in a protocol with the warden and so they know which of their views result from the same protocol instances.

A good example to illustrate this difference is protocol *DDH* in Section 2.3. The two ensembles according to (1) and (2) above are equal and thus protocol DDH would have to be regarded as perfectly diverted. This is counterintuitive because the warden in DDH uses less random coins than Alice and Bob together. On the other hand, according to Definition 3, DDH is only computationally diverted, which is the most we would expect.

3 Atomic Proxy Cryptography

A basic goal of public-key encryption is to allow only the key or keys selected at the time of encryption to decrypt the ciphertext. To change the ciphertext to a different key requires re-encryption of the message with the new key, which implies access to the original cleartext and to a reliable copy of the new encryption key. Intuitively, this seems a fundamental, and quite desirable, property of good cryptography; it should not be possible for an untrusted party to change the key with which a message can be decrypted.

Here, on the other hand, we investigate the possibility of *atomic proxy functions* that convert ciphertext for one key into ciphertext for another without revealing secret decryption keys or cleartext messages. An atomic proxy function allows an untrusted party to convert ciphertext between keys without access to either the original message or to the secret component of the old key or the new key. In proxy cryptography, the holders of public-key pairs A and B create and publish a *proxy key* $\pi_{A\to B}$ such that $D(\Pi(E(m, e_A), \pi_{A\to B}), d_B) = m$, where $E(m, e)$ is the public encryption function of message m under encryption key e, $D(c, d)$ is the decryption function of ciphertext c under decryption key d, $\Pi(c, \pi)$ is the atomic proxy function that converts ciphertext c according to proxy key π, and e_A, e_B, d_A, d_B are the public encryption and secret decryption component keys for key pairs A and B, respectively. The proxy key gives the owner of B the ability to decrypt "on behalf of" A; B can act as A's "proxy." In other words, the Π function effectively allows the "atomic" computation of $E(D(c, d_A), e_B)$ without revealing the intermediate result $D(c, d_A)$.

We consider atomic proxy schemes for encryption, identification and signatures. An encryption proxy key $\pi_{A\to B}$ allows B to decrypt messages encrypted for A and an identification or signature proxy key $\pi_{A\to B}$ allows A to identify herself as B or to sign for B (i.e., transforms A's signature into B's signature). Generating encryption proxy key $\pi_{A\to B}$ obviously requires knowledge of at least the secret component of A (otherwise the underlying system is not secure) and similarly generating identification or signature proxy key $\pi_{A\to B}$ requires B's secret, but the proxy key itself, once generated, can be published safely.

Categories of proxy schemes Encryption proxy functions (and similarly but contravariantly, identification or signature proxy functions) can be categorized according to the degree of trust they imply between the two key holders. Clearly, A must (unconditionally) trust B, since the encryption proxy function by definition allows B to decrypt on behalf of A. *Symmetric* proxy functions also imply that B trusts A, e.g., because d_B can be feasibly calculated given the proxy key plus d_A. *Asymmetric* proxy functions do not imply this bilateral trust. (Note that this model implies that proxy cryptography probably makes sense only in the context of public-key cryptosystems. Any secret-key cryptosystem with an asymmetric proxy function could be converted into a public-key system by publishing one key along with a proxy key that converts ciphertext for that key into ciphertext for a second key (which is kept secret.))

We can also categorize the asymmetric proxy schemes that might exist according to the convenience in creating the proxy key. In an *active asymmetric* scheme, B has to cooperate to produce the proxy key $\pi_{A\to B}$ feasibly, although the proxy key (even together with A's secret key) might not compromise B's secret key. In a *passive asymmetric* scheme, on the other hand, A's secret key and B's public key suffice to construct the proxy key. Clearly, any passive asymmetric scheme can be used as an active asymmetric scheme, and any asymmetric scheme can be used as a symmetric scheme.

Finally, we can (informally) distinguish proxy schemes according to the "metadata" they reveal about the identity of the secret-public key-pairs being transformed. *Transparent* proxy keys reveal the original two public keys to a third party. *Translucent* proxy keys allow a third party to verify a guess as to which two keys are involved (given their public keys). *Opaque* proxy keys reveal nothing, even to an adversary who correctly guesses the original public keys (but who does not know the secret keys involved).

Proxy schemes in theory and practice The proxy relationship is necessarily transitive. If there are public proxy keys $\pi_{A\to B}$ and $\pi_{B\to C}$, then anyone can compute a proxy function for $A \to C$. Symmetric proxy schemes further establish equivalence classes of keys where the secret component of any key can be used to decrypt messages for any other key in the same class. Note that creating a single symmetric proxy key between a key in one class and a key in another effectively joins the two classes into one.

The notion of proxy cryptography is a rather natural generalization of public-key cryptography and has some nice theoretical properties. The proxy schemes we consider below have the additional property that anyone can use the proxy key $\pi_{A\to B}$ to transform the public key of A to the public key of B. For such proxy schemes, as we will see in the various examples below, certain aspects of the security of publishing a proxy key actually follow from the fact that anyone, trusted or not, can use a proxy key to transform ciphertext and keys.

For example, suppose random messages m and m' are encrypted with random secret keys a and b as $E(m,a)$, $E(m',b)$. Suppose that knowing the proxy key $\pi_{A\to B}$ enables Eve, who knows neither a nor b, to recover m or m'. Then, ignoring

B altogether and starting with just two (presumably secure) ciphertexts $E(m, a)$ and $E(m', a)$, Eve can pick a random proxy key $r = \pi_{A \to Q}$ for some Q, transform $E(m', a)$ to $E(m', q)$ (where q is the unknown secret key of Q), transform A's public key into Q's public key, and proceed with the hypothesized cryptanalysis. We conclude that if it is safe for A to publish k messages then it is safe for A and B to publish a total of k messages *and* to publish a proxy key, provided only that Eve can successfully *apply* the proxy key to transform ciphertext and public keys.

Because proxy keys are tied to specific key pairs, it is not necessary in many applications to certify or otherwise take special care in distributing them (except to prevent denial-of-service). In particular, it is generally sufficient to rely on the certification and trust established in A (for encryption) or B (for signatures) when using proxy key $\pi_{A \to B}$, since a valid proxy key can by definition only be generated with the cooperation of the owner. Furthermore, the proxy function can be safely applied at any convenient time or place, by the message's sender or receiver, or at any intermediate (and possibly untrusted) point in the network.

Proxy functions potentially also have practical utility for key management in real systems. For example, some pieces of secure hardware (*e.g.,* smartcards) limit the number of secret keys that can be stored in secure memory, while some applications might require the ability to decrypt messages for more keys than the hardware can accommodate. With proxy cryptography, once a new key is created and a corresponding proxy key generated, the secret component of the old (or new) key can be destroyed, with the (public and externally-applied) proxy key maintaining the ability to decrypt for both. In effect, proxy functions allow us to increase the number of public keys without also increasing the number of secret bits or the amount of secret computation. Because proxy functions can be computed anywhere, messaging systems, such as electronic mail, can proxy "forward" messages encrypted with one key to a recipient who holds a different key. Proxy functions make it possible to associate a single key with a network or physical address but still decrypt messages forwarded (and proxied) from other addresses. Finally, proxy functions effectively allow changing or adding a key without obtaining new certificates or altering the distribution channel for the previous public key; this could be useful when it is difficult to distribute or certify new keys (e.g., old keys were published in widely-distributed advertisements or embedded in published software, or the certification authority charges high fees for new certificates).

Security of proxy schemes and ad hoc substitutes If Alice wants Bob to be able to read her mail, instead of issuing a proxy key she might just give Bob her secret key (perhaps, obviating the need to involve Bob, by encrypting it in Bob's public key and publishing it). This would be inferior to using a proxy scheme for several reasons. First, as discussed above, Bob's computing environment may be limited and therefore incapable of automatically processing encrypted secret keys; any new software to decrypt and manage such keys would have to run within the environment trusted by Bob. Proxy processing, on the

other hand, can take place entirely outside of Alice's and Bob's trusted environments and without their active involvement. Furthermore, encrypting one's secret key with another's public key is not in general secure. The cryptosystem we present below, a variant of ElGamal [ElG85], is thought to be secure in part because the cryptanalysis problem is random-self-reducible—which allows one to assert mathematically that recovering m from the public information $\langle e_a, E(m, e_a), e_b \rangle$ is hard on average if it is hard at worst. The task of recovering m from $\langle e_a, E(m, e_a), E(d_a, e_b), e_b \rangle$, however, may be considerably easier since $E(d_a, e_b)$, in the context of e_a and e_b, may leak information about d_a—specifically, the new cryptanalysis problem is probably not random-self-reducible and due to the problem's obscurity it is not clear what, if any, mathematical guarantees of security can be given. By contrast, the proxy scheme we give below is just as strong as the underlying cryptosystem.[5]

Related work A natural question to ask is whether there exist atomic proxy functions (and feasible schemes to generate proxy keys) for any public key cryptosystems.

Previous work on delegating the power to decrypt has focused on developing efficient transformations that allow the original recipient to forward *specific ciphertexts* to another recipient. Mambo and Okamoto [MO97] develop this formulation and give efficient transforms (more efficient than decryption and re-encryption) for ElGamal and RSA. Mambo, Usuda and Okamoto [MUO96] apply a similar notion to signature schemes.

While such schemes have value from the standpoint of efficiency, they are not, however, "atomic proxy cryptosystems" by our definition because the transforming function must be kept secret and applied online by the original keyholder on a message-by-message basis (the schemes are not atomic). The security semantics of these systems are essentially the same as a decryption operation followed by a re-encryption operation for the new recipient. Our formulation of proxy cryptography is distinguished from the previous literature by the ability of the keyholder to publish the proxy function and have it applied by untrusted parties without further involvement by the original keyholder.

3.1 Proxy encryption

Although the problem of proxy cryptography seems like a natural extension of public-key cryptography, existing cryptosystems do not lend themselves to obvious proxy functions. RSA [RSA78] with a common modulus is an obvious candidate, but that scheme is known to be insecure [Sim83,DeL84]. Similarly, there

[5] Note that Bob of this example may be a government mandating that Alice provide him with access to her key. It has been argued that such a scheme makes the system as a whole less trustworthy due to the extra engineering effort involved; we argue here that in the case of random-self-reducible cryptosystems such as ElGamal variants, requiring Alice to encrypt her secret key using the government's public key may also weaken the underlying cryptosystem in the precise mathematical sense of spoiling the random-self-reducibility.

do not appear to be obvious proxy functions for many of the previous discrete-log-based cryptosystems. This is not to say, of course, that proxy functions for existing systems do not exist.

We now describe a new secure discrete-log-based public-key cryptosystem that does have a simple proxy function. The scheme is similar in structure to ElGamal encryption [ElG85], but with the parameters used differently and the inverse of the secret used to recover the message.[6] (This approach has merit beyond proxy encryption; [Hug94] proposed a Diffie-Hellman-like key agreement protocol based on the inverse of the secret, which allows a message's sender to determine the key prior to identifying its recipient).

Cryptosystem $\mathcal{X}$ (encryption) Let p be a prime of the form $2q + 1$ for a prime q and let g be a generator in $\mathbb{Z}_p^*$; p and g are global parameters shared by all users. A's secret key a, $0 < a < p - 1$, is selected at random and must be in $\mathbb{Z}_{2q}^*$, *i.e.*, relatively prime to $p - 1$. (A also calculates the inverse $a^{-1} \bmod 2q$). A publishes the public key $g^a \bmod p$. Message encryption requires a unique randomly-selected secret parameter $k \in \mathbb{Z}_{2q}^*$. To encrypt m with A's key, the sender computes and sends two ciphertext values (c_1, c_2):

$$c_1 = mg^k \bmod p$$
$$c_2 = (g^a)^k \bmod p$$

Decryption reverses the process; since

$$c_2^{(a^{-1})} = g^k \pmod p$$

it is easy for A (who knows a^{-1}) to calculate g^k and recover m:

$$m = c_1((c_2^{(a^{-1})})^{-1}) \bmod p$$

The efficiency of this scheme is comparable to standard ElGamal encryption.

Symmetric proxy function for $\mathcal{X}$ Observe that the c_1 ciphertext component produced by Cryptosystem $\mathcal{X}$ is independent of the recipient's public key. Recipient A's key is embedded only in the c_2 exponent; it is sufficient for a proxy function to convert ciphertext for A into ciphertext for B to remove A's key a from c_2 and replace it with B's key b. Part of what a proxy function must do, then, is similar to the first step of the decryption function, raising c_2 to a^{-1} to remove a. The proxy function must also contribute a factor of b to the exponent. Clearly, simply rasing c_2 to a^{-1} and then to b would accomplish this, but obviously such a scheme would not qualify as a secure proxy function; anyone who examines the proxy key learns the secret keys for both A and B.

This problem is avoided, of course, by combining the two steps into one. Hence, the proxy key $\pi_{A \to B}$ is $a^{-1}b$ and the proxy function is simply $c_2^{\pi_{A \to B}}$.

[6] David Wagner notes that this proxy scheme can be extended to work with standard ElGamal encryption.

Note that this is a symmetric proxy function; A and B must trust one another bilaterally. B can learn A's secret (by multiplying the proxy key by b^{-1}), and A can similarly discover B's key. Observe that applying the proxy function is more efficient than decryption and re-encryption, in that only one exponentiation is required.

Security of $\mathcal{X}$ First, we show that $\mathcal{X}$ is secure—that cleartext and secret keys cannot be recovered from ciphertext and public keys. Beyond that, we also show that publishing the proxy key compromises neither messages nor secret keys. Since recovering a secret key enables an adversary to recover a message and since cryptanalysis is easier with more information (i.e., a proxy key), it is sufficient to show that no cleartext is recoverable from ciphertext, public keys, and proxy keys. Specifically, we will show that the problem of recovering m from

$$(g^a, g^b, g^c, \ldots, mg^k, g^{ak}, a^{-1}b, a^{-1}c, \ldots).$$

is at least as hard as Diffie-Hellman.

Theorem 6. *Suppose there exists a randomized algorithm f that with probability $\epsilon > 1/|p|^{O(1)}$ succeeds in recovering m from the public information*

$$(g^a, g^b, \ldots, mg^k, g^{ak}, b/a, \ldots)$$

where the probability is taken over f's random choices as well as over m and the parameters a, b, and k. Then, for each $\eta = 2^{-|p|^{O(1)}}$, there exists a randomized polynomial-time algorithm for Diffie-Hellman that succeeds with probability $1-\eta$.

Proof. The proof is found in [BS98].

Similarly one can show that recovering a from $(g^a, g^b, mg^k, g^{ak}, b/a)$ is as hard as the discrete log, so publishing the proxy key does not compromise a—not even to the level of Diffie-Hellman.

3.2 Proxy identification

In this section we describe a discrete-log-based identification scheme. With p, g, a as before, Alice wishes to convince Charlotte that she controls a; Charlotte will verify using public key g^a. As before, the proxy key $\pi_{A \to B}$ will be a/b—it will be safe to publish a/b and Alice and Charlotte can easily use a/b to transform the protocol so Charlotte is convinced that Alice controls b.

Note that in the case of a secure identification proxy key that transforms identification by A into identification by B, it is B whose secret is required to construct the proxy key because identification as B should not be possible without B's cooperation.

Cryptosystem $\mathcal{Y}$ (identification) Let p and g be a prime and a generator in $\mathbb{Z}_p^*$, respectively. Alice picks random $a \in \mathbb{Z}_{2q}^*$ to be her secret key and publishes g^a as her public key. Each round of the identification protocol is as follows:

- Alice picks a random $k \in \mathbb{Z}_{2q}^*$ and sends Charlotte $s_1 = g^k$.
- Charlotte picks a random bit and sends it to Alice.
- Depending on the bit received, Alice sends Charlotte either $s_2 = k$ or $s_2' = k/a$.
- Depending on the bit, Charlotte checks that $(g^a)^{s_2'} = s_1$ or that $g^{s_2} = g^k$.

This round is repeated as desired. As with existing protocols, there may be ways to perform several rounds in parallel for efficiency [FFS88].

Symmetric proxy function for $\mathcal{Y}$ A symmetric proxy key is a/b. Suppose Charlotte wants to run the protocol with g^b instead of g^a. Either Alice or Charlotte or any intermediary can use the proxy key to convert Alice's responses k/a to k/b.

Security of $\mathcal{Y}$

Theorem 7. *Protocol $\mathcal{Y}$, with or without proxy keys published, is a zero knowledge protocol that convinces the verifier that the prover knows the secret key.*

Proof. The proof is found in [BS98].

3.3 Proxy signature

The concept of proxy cryptography also extends to digital signature schemes. A signature proxy function transforms a message signature so that it will verify with a public key other than that of the original signer. In other words, a signature proxy function $\Pi(s, \pi_{A\to B})$ with proxy key $\pi_{A\to B}$ transforms signature s signed by the secret component of key A such that $V(m, \Pi(S(m, A), \pi_{A\to B}), B)$ returns VALID, where $S(m, k)$ is the signature function for message m by key k and $V(m, s, k)$ is the verify function for message m with signature s by key k.

Again, existing digital signature schemes such as RSA [RSA78], DSA [NIS91], or ElGamal [ElG85], etc. do not have obvious proxy functions (which, again, is not to say that such functions do not exist).

As in the case of proxy identification, in order to construct a proxy key that transforms A's signature into B's signature, B's secret must be required to construct the proxy key because signing for B should not be possible without B's cooperation.

Now we will see how to use the proxy identification scheme to construct a proxy signature scheme. We suppose there exists a hash function h whose exact security requirements will be discussed below. The parameters p, g, a, b are as before.

Cryptosystem $\mathcal{Z}$ (signature) To sign a message m, Alice picks $k_1, k_2, \ldots k_\ell$ at random and computes $g^{k_1}, \ldots g^{k_\ell}$. Next Alice computes $h(g^{k_1}, \ldots g^{k_\ell})$ and extracts ℓ pseudorandom bits $\beta_1, \ldots, \beta_\ell$. For each i, depending on the i'th pseudorandom bit, Alice (who knows a) computes $s_{2,i} = (k_i - m\beta_i)/a$; that is, $s_{2,i} = (k_i - m)/a$ or $s_{2,i} = k_i/a$. The signature consists of two components:

$$s_1 = (g^{k_1}, \ldots, g^{k_\ell})$$
$$s_2 = ((k_1 - m\beta_1)/a, \ldots, (k_\ell - m\beta_\ell)/a)$$

To verify the signature, first the β_i's are recovered using the hash function. The signature is then verified one "round" at a time, where the i'th round is $(g^{k_i}, (k_i - m\beta_i)/a)$. To verify $(g^k, (k - m\beta)/a)$ using public key g^a, the recipient Charlotte raises (g^a) to the power $(k - m\beta)/a$ and checks that it matches $g^k/g^{m\beta}$.

Symmetric proxy function for $\mathcal{Z}$ A symmetric proxy key $\pi_{A \to B}$ for this signature scheme is a/b. The proxy function Π leaves s_1 alone and maps each component $s_{2,i}$ to $s_{2,i}\pi_{A \to B}$.

Security of $\mathcal{Z}$ This scheme relies on the existence of a "hash" function h. Specifically (Hash Assumption), we assume there exists a function h such that:

- On random input (g^a, m), it is difficult to generate $\{r_i\}$ and $\{\beta_i\}$ such that
$$h(g^{ar_1 + m\beta_1}, \ldots, g^{ar_\ell + m\beta_\ell}) = \langle \beta_1 \ldots, \beta_\ell \rangle.$$
- More generally, it is difficult to generate such $\{r_i\}$ and $\{\beta_i\}$ on input g^a, m, and samples of signatures on random messages signed with a.

It is not our intention to conjecture about the existence of such functions h. In particular, we do not know the relationship between the hash assumption and assumptions about collision freedom or hardness to invert.[7] We note that this generic transformation of a protocol to a signature scheme has appeared in the literature [FS87].

We now analyze the hash assumption. Note that in order to produce a legitimate signature on m that verifies with g^a, a signer needs to produce $\langle g^{k_i} \rangle$ and $\langle (k_i - m\beta_i)/a \rangle$. Thus, putting $\langle \beta_i \rangle = h(\langle g^{k_i} \rangle)$ and then $\langle r_i \rangle = \langle (k_i - m\beta_i)/a \rangle$, it is straightforward to see that the signer could actually produce r_i's and β_i's of the stated type in the course of producing the signature.

While we do not address the security of h, we can state that issuing proxy keys does not weaken the system.

Theorem 8. *Suppose h satisfies the hash assumption. Then, for most b, it is also hard to produce $\{r_i\}$ and $\{\beta_i\}$ given additional input $a/b, g^b$, and samples of messages signed with b.*

Proof. The proof is found in [BS98].

[7] The hash assumption *does* imply that, on random input g^a, it is hard to find $\langle r_i \rangle$ making all the β_i's zero, i.e., such that $h(g^{ar_1}, \ldots, g^{ar_\ell}) = 0$.

4 Conclusions

Conceptually, divertibility and proxiability of protocols are both defined in terms of an effectiveness property and one or two security properties. The effectiveness property is basically the same in both cases, namely extensibility as in Definition 3. Our more recent work shows that a proxy key can be naturally incorporated into (and makes sense for) divertibility as well. In the case of divertibility, the security requirement is that Alice and Bob cannot communicate any subliminal message through the warden. In the case of proxiability, the security requirement is that the proxy key releases no more information than what either Alice or Bob would release in the original protocol. A complete unifying framework remains as future work.

We have introduced the notion of perfect and computational protocol divertibility, and have given a sufficiency criterion for the former. All existing diverted protocols we have found in the literature turned out to satisfy this criterion. The first example of a diverted key distribution protocol was given. This is also the first computationally divertible protocol we know of.

Intuitively, atomic proxy cryptography is a fairly natural extension of the basic notion of public-key cryptography. It surely seems plausible, given that there exist cryptosystems that can grant the ability to encrypt without granting the ability to decrypt, that there might also exist cryptosystems that can grant the ability to *re-encrypt* without granting the ability to decrypt. However, it is not at all obvious whether there exist atomic proxy schemes in general.

Indeed, while this paper demonstrates that there do exist efficient and secure public-key encryption and signature schemes with symmetric atomic proxy functions, this observation probably raises more new questions than it answers. In particular, do proxy functions exist for public-key cryptosystems based on problems other than discrete-log? (One possibility is that, for some cryptosystems, proxy functions do exist but it is infeasible to find a proxy key.) More importantly, we have yet to discover a secure *asymmetric* proxy function of any kind; asymmetric proxy functions are probably much more useful in practice, since there are likely many situations where trust is only unidirectional. Are there cryptosystems for which asymmetric proxy functions exist?

5 Acknowledgements

We thank Steve Bellovin, Matthew Franklin, Jack Lacy, Dave Maher, Andrew Odlyzko and David Wagner for helpful discussions and comments.

References

[BS98] Matt Blaze, Martin Strauss. Atomic Proxy Cryptography. AT&T Labs-Research TR98.5.1 <http://www.research.att.com/library/trs>

[Bleu97] Gerrit Bleumer. On Protocol Divertibility. AT&T Labs-Research TR97.34.2 <http://www.research.att.com/library/trs>

[BD91] Mike V. D. Burmester, Yvo Desmedt. All languages in NP have divertible zero-knowledge proofs and arguments under cryptographic assumptions. *Eurocrypt '90* LNCS 473, Springer-Verlag 1991, 1–10.

[CEG88] David Chaum, Jan-Hendrik Evertse, Jeroen van de Graaf. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. *Eurocrypt '87*. LNCS 304, Springer-Verlag 1988, 127–141.

[CPS95] Jan L. Camenisch, Jean-Marc Piveteau, Markus A. Stadler. Blind Signatures Based on the Discrete Logarithm Problem. *Eurocrypt '94*. LNCS 950, Springer-Verlag 1995, 428–432.

[DeL84] John M. DeLaurentis. A Further Weakness in the Common Modulus Protocol for the RSA Cryptoalgorithm. *Cryptologia* 8/3 (1984) 253–259.

[DH76] Whitfield Diffie, Martin E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*. 22/6 (1976) 644–654.

[ElG85] Taher ElGamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Transactions on Information Theory*. 31/4 (1985) 469–472.

[FFS88] Uriel Feige, Amos Fiat, Adi Shamir. Zero-Knowledge Proofs of Identity. *Journal of Cryptology* 1/2 (1988) 77–94.

[FS87] Amos Fiat, Adi Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. *Crypto '86*. LNCS 263, Springer-Verlag 1987, 186–194.

[GMR89] Shafi Goldwasser, Silvio Micali, Charles Rackoff. The Knowledge Complexity of Interactive Proof Systems. *SIAM J. Computing*. 18/1 (1989) 186–207.

[HMP95] Patrick Horster, Markus Michels, Holger Petersen. Meta-Message Recovery and Meta-Blind Signature Schemes Based on the Discrete Logarithm Problem and Their Applications. *Asiacrypt '94*. LNCS 917, Springer-Verlag 1995, 224–237.

[Hug94] Eric Hughes. An encrypted key transmission protocol. *Crypto '94 Rump Session* presentation, August 1994.

[ISS91] Toshija Itoh, Kouichi Sakurai, Hiroki Shizuya. Any Language in IP has a Divertible ZKIP. *AsiaCrypt '91*. Springer-Verlag 1993, 382–396.

[MO97] Masahiro Mambo, Eiji Okamoto. Proxy cryptosystems: delegation of the power to decrypt ciphertexts. *IEICE Trans. Fund. Electronics Communications and Comp Sci.* E80-A/1 (1997) 54–63.

[MUO96] Masahiro Mambo, Keisuke Usuda, and Eiji Okamoto. Proxy signatures: delegation of the power to sign messages. *IEICE Trans. Fund. of Electronic Communications and Comp Sci.* E79-A/9 (1996) 1338–1354.

[NIS91] NIST. A proposed federal information processing standard for digital signature standard (DSS). *Draft Tech. Rep. FIPS PUB XXX*, August 1991. Standards Publication (FIPS

[OO90] Tatsuaki Okamoto, Kazuo Ohta. Divertible zero-knowledge interactive proofs and commutative random self-reducibility. *Eurocrypt '89* LNCS 434, Springer-Verlag 1990, 134–149.

[RSA78] Ronald L. Rivest, Adi Shamir, Leonhard Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *CACM* 21/2 (1978) 120-126, reprinted: 26/1 (1983) 96–99.

[Sim83] Gustavus J. Simmons. A "Weak" Privacy Protocol Using the RSA Crypto Algorithm. *Cryptologia* 7/2 (1983) 180–182.

[Sim84] Gustavus J. Simmons. The Prisoners' Problem and the Subliminal Channel. *Crypto '83*. Plenum Press, New York 1984, 51–67.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 April 2004 (22.04.2004)

PCT

(10) International Publication Number
WO 2004/034223 A2

(51) International Patent Classification[7]: G06F

(21) International Application Number:
PCT/US2003/032153

(22) International Filing Date: 8 October 2003 (08.10.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/417,913 9 October 2002 (09.10.2002) US

(71) Applicant *(for all designated States except US)*: **LEGAL IGAMING, INC.** [US/US]; 200 Ultra Drive, Henderson, NV 89074 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SAUNDERS, Michael, W.** [US/US]; 200 Ultra Drive, Henderson, NV 89074 (US). **MILLER, William, D., III** [US/US]; 8740 Country Pines Avenue, Las Vegas, NV 89129 (US). **CARLSON, Rolf, E.** [US/US]; 211 Dartmouth Avenue S.E., Albuquerque, NM 87106 (US).

(74) Agent: **MALLON, Joseph, J.**; Knobbe, Martens, Olson & Bear, LLP, 2040 Main Street, 14th Floor, Irvine, CA 92614 (US).

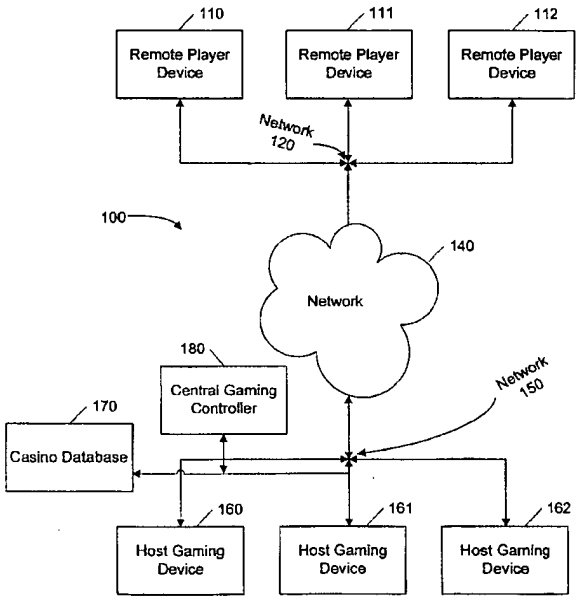
(81) Designated States *(national)*: AE, AG, AL, AM, AT (utility model), AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ (utility model), CZ, DE (utility model), DE, DK (utility model), DK, DM, DZ, EC, EE (utility model), EE, EG, ES, FI (utility model), FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK (utility model), SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declaration under Rule 4.17:
— *of inventorship (Rule 4.17(iv)) for US only*

[Continued on next page]

(54) Title: SYSTEM AND METHOD FOR CONNECTING GAMING DEVICES TO A NETWORK FOR REMOTE PLAY



(57) Abstract: A system (100) and method for connecting remote player devices (110) to regulated host gaming devices (160) in a network to provide remote game play. A host gaming device (160) is configured to provide game information to a plurality of remote player devices (110) to allow remote play of the host game device (160). Whether each remote player device (110) is permitted to receive gaming data is based upon, at least in part, the geographic location of the remote player device (110).

WO 2004/034223 A2

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND METHOD FOR CONNECTING GAMING DEVICES TO A NETWORK FOR REMOTE PLAY

Background of the Invention

Field of the Invention

[0001] The present invention generally relates to electronic devices. In particular, the invention relates to methods and systems of interactive gaming.

Description of the Related Technology

[0002] Traditionally, the way for a gaming operator to increase revenue from gaming devices is to increase the number of gaming devices available for play. In order for casinos to increase the number of gaming devices available for play, casino floor space must be added to house the additional gaming devices. The floor space allocated to house additional gaming devices must meet specific criteria as defined by the gaming authority for the jurisdiction in which the gaming devices are to be located. Providing additional floor space is an expensive process for casino operators and often requires constructing new casino properties. Also, adding gaming devices typically requires payment of additional licensing fees for each additional game.

[0003] A trend in the gaming industry has been to provide Internet gaming. Internet gaming allows players to make wagers on the outcome of casino style games similar to that described above, except that the player does not have to be physically located in a casino to do so. Internet players make wagers and play casino games using a personal computer and wager on games running on computers connected to the Internet.

[0004] More broadly, interactive gaming is the conduct of gambling games through the use of electronic devices. The popularity of Internet gambling sites has indicated a strong market for remotely accessible gaming, or other interactive gaming. Regulated casino operators strongly desire to provide interactive gaming while capitalizing on existing infrastructure. Thus there is a need for improved electronic devices that support regulated remote gaming.

Summary of the Invention

[0005] The system of the present invention has several aspects, no single one of which is solely responsible for its desirable attributes. Without limiting the scope of this invention as expressed by the claims which follow, its more prominent features will now be discussed briefly. After considering this discussion, and particularly after reading the section entitled "Detailed Description of the Invention" one will understand how the features of this invention provide advantages which include providing remote gaming in regulated environment.

[0006] A gaming system and method of using the same to allow a host gaming device to be played from remote player devices to allow casino operators to obtain maximum advantage from their gaming licenses.

[0007] More particularly, in one embodiment gaming system may comprise a data network, a host gaming device connected to the data network, the gaming device configured to execute at least one game and a plurality of remote player devices connected to the data network. Each of the remote player devices is configured to receive game information provided by the host gaming device. Whether each remote player device is permitted to receive gaming data may be based upon, at least in part, the geographic location of the remote player device.

[0008] The host gaming device may be configured to allow no more than a predetermined number of remote player devices to concurrently receive game information provided by the host gaming device during the gaming session. This predetermined number may be determined by a gaming agency.

[0009] In another embodiment of a gaming system, at least one of the plurality of remote player devices may be permitted to receive game data based upon, at least in part, the geographic location of the remote player device, an age of a user of the remote player device.

[0010] A gaming system according to the invention may also include a central gaming controller configured to record gaming transactions on the host gaming device and on each remote gaming device.

[0011] The data network may be, in part, the Internet, and be comprised of one or more logical segment, which may include closed-loop networks. The host gaming device may be configured to identify the geographic location of a remote player device based, at least in part, on a logical segment corresponding to the remote player device. A mobile communications network, or a GPS device may also allow identification of the geographic location of the remote player device.

[0012] The host gaming device may be in a location approved by a gaming agency and include at least one game control configured to provide local use. This game control may be disabled when the host gaming device is providing game information to a remote player device. A host gaming device may also be configured to save an encrypted game state allowing a game to be resumed following a device or network failure.

[0013] A remote player device may be coupled to a credential device configured to receive information relating to a user of the remote player device. The information relating to a user may include the age of the user, or a password that is input by the user. The credential device is a smart card reader, a biometric device such as a fingerprint reader, or any type of input device. The credentials may be verified against information, such as age, password, or fingerprint in a database configured to provide information associated with each of a plurality of users of the gaming system.

[0014] In another embodiment, a gaming system may be comprised of a means for executing at least one game, the game providing game information during its execution, a local access means provides local access to the game information for a user in a location approved by a gaming agency, player means for receiving game information, presenting the game information to a user and providing at least one game control, a means for providing the game information over a data network to a predetermined number of receiving means, means for determining the location of the receiving means, and means for disabling the local access means. Other similar embodiments may also be comprised of means for creating an auditable record of gaming transactions on the playing means and on the gaming means.

[0015] Another embodiment of a gaming system, in addition to the features of the embodiments discussed above, may also include customized promotional messages to players of gaming devices.

[0016] On a remote player device, an embodiment of a method of remotely accessing a host gaming device may include: establishing access to the host gaming device through a data network, receiving gaming related information from the host gaming device through the data network, presenting the gaming related information to a player, receiving at least one control signal from the player, sending the control signal to the host gaming device through the data network, and disabling local use of the host gaming device. In one embodiment, the method may also include recording each gaming transaction occurring on the remote player device. Another embodiment of the method may include providing a geographic location of the remote player device. In another embodiment of the method, the age of the user of the remote player device is also provided.

[0017] On a host gaming device, an embodiment of a method of providing remote access, including: verifying the geographic location of a remote player device, establishing a gaming session on a host gaming device from a remote player device through a data network, receiving at least one control signal from the remote player device through the data network, and sending gaming related information from the gaming device through the data network. One embodiment of a method may also include recording each gaming transaction occurring on the host gaming device,

[0018] In order to provide tolerance for failures of system components, a method of resuming an interrupted gaming session on a gaming device is provided. One embodiment of a method may include generating a gaming state of the gaming session on the first gaming device, encrypting the gaming state, transporting the encrypted gaming state from the gaming device. The method may also include the converse: transporting the encrypted gaming state from the first gaming device to a second gaming device, decrypting the gaming state on the second gaming device; and loading the game state into a second gaming device to resume the gaming session.

[0019] An embodiment of a gaming system which provides for resuming interrupted gaming sessions across a data network. The system may include a first host gaming device connected to the data network, the gaming device configured to execute at least one game, generate a gaming state based on execution of at least one game, encrypt the gaming state, and send the encrypted gaming state over the data network. A second host gaming device may be connected to the data network, the second gaming device configured to receive the encrypted gaming state over the data network, decrypt the gaming state, and resume executing at least one game from the gaming state. A plurality of remote player devices, configured to receive game information provided by the host gaming device, may be connected to the data network. The gaming state may include user payment or credit information, and game jackpot or payout information.

[0020] Another embodiment of a gaming system providing resumption of interrupted gaming sessions may include means for executing at least one game, means for generating a gaming state based on execution of at least one game, means for encrypting the gaming state, and means for sending the encrypted gaming state. The system may also include means for receiving the encrypted gaming state, means for decrypting the gaming state and means for resuming executing at least one game from the gaming state.

[0021] To enable gaming regulatory compliance, methods authenticating gaming system users are also provide. An embodiment of a method of authenticating a user of a host gaming device may include receiving a security certificate from the smart card, sending the security certificate from the gaming device to an authenticator device, receiving an authentication reply from the authenticator, and playing a game in response to the authentication reply.

[0022] An embodiment of the method may also include presenting the security certificate from the gaming device to a certificate authority for authentication over a data network.

[0023] An embodiment of a method of authenticating a user of a remote player device for playing a host gaming device may include receiving an indicia of identity for a user, sending the indicia of identity to an authenticator device, receiving an authentication reply from the authenticator device, and authorizing use of a host gaming device based on the indicia of identity. The indicia of identity for a user may be provided by a biometric device, a smart card, or a password provided by the user.

[0024] Another embodiment of a gaming system provides authentication of users. The system may include a data network, a host gaming device interfaced to the data network, a plurality of remote player devices interfaced to the data network, and a security device configured to provide player credentials to at least one remote player device. The each of the remote player devices may be configured to receive game information provided by the host gaming device. The host gaming device may provide game information to a predetermined number of permitted remote

player devices. Whether a remote player device is permitted to receive gaming information may be based upon, at least in part, on player credentials provided by the security device.

[0025] In one embodiment, a method of remotely accessing a gaming device provides for creating records of gaming transactions on both host gaming devices and remote player devices sufficient to provide an auditable record for a gaming authority in the jurisdiction. The method may include establishing a gaming session on a gaming device for a remote player device through a data network, sending gaming related information from the gaming device through the data network, receiving at least one control signal from the remote player device through the data network, creating an auditable gaming session record representing each gaming transaction of a gaming session on the host gaming device and on the remote gaming device. In addition, the record may be sent to a third party, such as a gaming authority, through the data network.

[0026] In another embodiment of a gaming system, the gaming system includes a network comprised of a plurality of logical segments. A security policy controls the flow of data between logical segments. A host gaming device may be connected to the data network, the gaming device configured to execute at least one game. A plurality of remote player devices may be connected to the data network. The plurality of remote player devices are each configured to receive game information provided by the host gaming device, and to control a gaming session established on the gaming device, subject to the security policy. The security policy may be based, at least in part, on the geographic location of a logical segment.

[0027] One embodiment of the gaming system may include a promotional message server to deliver customized promotional messages to users of the gaming system. In this embodiment, a gaming system may include a data network, a promotional message server configured to provide customized promotional messages. Each message may be customized with information associated with a user of the gaming system. In addition, a gaming system may include a host gaming device interfaced to the data network, and a plurality of remote player devices interfaced to the data network. The plurality of remote player devices are each configured to receive game information provided by the host gaming device and to receive and present promotional messages.

[0028] In another embodiment, a gaming system may include a means for data communication, means for executing at least one game, means for providing game information over the data network to a predetermined number of receiving means, a plurality of means for receiving game information over the data communication means. Each means for receiving game information may be coupled to a means for receiving customized promotional messages. A gaming system may also include a means for presenting promotional messages in conjunction with gaming data.

[0029] A related method of displaying information on a remote player device is also provided. The method may include receiving a promotional message on a remote player device, presenting the promotional message in conjunction with gaming information for an amount of time; and removing the promotional message from the remote player device. Information in the promotional message may be used to calculate the amount of time to present the promotional message.

[0030] A remote player interface of a gaming system may have a number of embodiments. In one embodiment of a gaming system, the gaming system includes data network, a host gaming device interfaced to the data network, and at least one remote player device interfaced to the data network. The remote player device is configured to receive game information provided by the host gaming device. The remote player interface of the gaming system may include a video display device in communication with the remote player device and a remote control device in communication with the remote player device. The remote control device is configured to control operation of a game.

[0031] An embodiment of method of remotely accessing a gaming device may include establishing a gaming session on the host gaming device from a remote player device through a data network, receiving gaming related information from the host gaming device through the data network, presenting gaming related information to a player via a video display device, receiving at least one control signal generated by a remote control device for controlling the gaming session, and sending the control signal to the host gaming device through the data network.

Brief Description of the Drawings

[0032] **FIG. 1** depicts a simplified block diagram of a gaming system according to one embodiment of the invention.

[0033] **FIG. 2** depicts a simplified block diagram of system elements relating to a host gaming device of **FIG. 1** according to one embodiment of the invention.

[0034] **FIG. 3** depicts a simplified block diagram of system elements relating to a remote player device of **FIG. 1** according to one embodiment of the invention.

[0035] **FIG. 4** is a flowchart depicting the sequence of events for acknowledging command messages in a gaming system as embodied in **FIG. 1.**

[0036] **FIG. 5** is a flowchart depicting the sequence of events for establishing a remote gaming session, playing a game, and terminating the remote gaming session in a gaming system as embodied in **FIG. 1.**

[0037] **FIG. 6** is a flowchart depicting the sequence of events for transferring funds from a player's source of funds in the gaming system of **FIG. 1.**

[0038] **FIG. 7** is a flowchart depicting the sequence of events for a host gaming device of **FIG. 2** to connect to a network using security certificates and a certificate authority.

[0039] **FIG. 8** is a flowchart depicting the sequence of events for a gaming device of **FIG. 2** to build and deliver an encrypted block of data representing the complete state of the gaming device.

[0040] **FIG. 9** is a flowchart depicting the sequence of events for retrieving a block of data representing the state of a gaming device from a database and loading the block into a gaming device as performed by a gaming system embodiment as in **FIG. 1**.

[0041] **FIG. 10** is a more detailed block diagram of a gaming system as depicted in **FIG. 1**.

[0042] **FIG. 11** is a detailed block network diagram of a portion of a gaming system as depicted in **FIG. 10**.

Detailed Description of the Preferred Embodiment

[0043] The following detailed description is directed to certain specific embodiments of the invention. However, the invention can be embodied in a multitude of different ways as defined and covered by the claims. In this description, reference is made to the drawings wherein like parts are designated with like numerals throughout.

[0044] In a traditional casino environment, gaming devices are generally located on a gaming floor. Gaming devices are subject to regulation by gaming regulatory agencies. Regulations may limit the locations where gaming devices may be placed and by limit users of gaming devices to those of legal age to gamble in the respective jurisdiction. Regulatory agencies for a given jurisdiction may also limit the number of licensed gaming devices provided to a licensee. Where gaming devices are physically located on a casino gaming floor, verification of whether a device is being used in its licensed location within the jurisdiction may be determined by physical inspection of the gaming floor. Further, monitoring of the gaming floor in casinos ensures that players are of legal age as set by the jurisdiction.

[0045] An embodiment of a gaming system according to the present invention allows a licensed host gaming device to be used by one or more remote player devices geographically separated from the host gaming device, but still located within the jurisdiction of a gaming authority. **FIG. 1** depicts a simplified block diagram of an embodiment of a gaming system **100** according to the invention. One or more host gaming devices **160, 161, 162** are licensed gaming devices. Although three host gaming devices are shown on **FIG. 1**, the gaming system **100** may employ any number of host gaming devices ranging from one to thousands. For convenience of discussion, set forth below is a description of certain aspects of the host gaming device **160**. It is to be appreciated that the other gaming devices may contain the following or different aspects.

[0046] A host gaming device may be any device, comprised of electronic, mechanical, or a combination of electronic and mechanical components, which is used for gaming and which affects the result of a wager by determining win or loss. A host gaming device **160** is connected to a data network **150**. In the embodiment depicted in **FIG. 1**, the data network of gaming system **100** is comprised of three logical segments. Gaming network **150** connects each host gaming device **160** and related elements such as the database **170** and central gaming controller **180**. Remote network **120** connects remote player devices **110, 111, 112** to the system. Backbone network **140** provides interconnection between the gaming network **150** and the remote network **120**.

[0047] The database **170** may be computer server running database software, or any other commercially available database solution. In one embodiment, as depicted, the database **170**, is a casino database. In other embodiments, the database may also contain other data related, or unrelated to the casino operation.

[0048] Remote network **120** connects remote player devices **110, 111, 112** to the system. Each remote player device **110** allows a user to play a game executing on a host gaming device **160**. For convenience of discussion, set forth below is a description of certain aspects of the remote player device **110**. It is to be appreciated that the other remote player devices may contain the following or different aspects. Although three remote player devices are shown on **FIG. 1**, the gaming system **100** may employ any number of remote player devices ranging from one to thousands.

[0049] The remote network **120** may be any form of computer network, as discussed below. In one particular embodiment, the remote network **120** is part of a network provided by a cable television system. **FIG. 10** depicts an embodiment of a gaming system where the remote network **120** is provided through a digital home communications terminal (DHCT) **1000**, such as a set-top box.

[0050] Each host gaming device **160** may be located in any location approved by a gaming agency, such as a casino gaming floor. A host gaming device **160** provides a legally regulated random number generator. Once generation of random number has been performed, a game result is determined. Any further interaction through the game's user interface is for the benefit of a user. For example, in one embodiment of a gaming system, the host gaming device may be a slot machine. After payment is made, through a coin, token, credit device, etc, the player pulls a lever arm to execute play. In a mechanical game, for example, a slot machine, a game result may be determined by the interaction of spinning wheels. In a host gaming device **160** of an embodiment of the present invention, however, pulling the arm triggers generation of a random number which determines the game result. Thus any spinning wheels or its electronic equivalent is

purely for entertainment of the user. A host gaming device **160** plays at least one game of chance, including, but not limited to, Slots, Blackjack, Poker, Keno, Bingo, or Lotteries.

[0051] **FIG. 2** depicts a more detailed block diagram of an embodiment of a gaming system **100** showing additional gaming system elements coupled to the host gaming device **160**. The host gaming device **160** may include local controls **220** such as an arm. The host gaming device **160** may have a display **210** to present the results of a game to a user. Further, the gaming device **160** may have a smart card reader **280**. Functions of the smart card reader **280** may include receiving payment for a game, or identifying a user for promotional or loyalty programs. A biometric identity device **290**, such as a fingerprint scanner, may be used for similar functions by the gaming system.

[0052] Networks **120, 140, 150** may include any type of electronically connected group of computers including, for instance, the following networks: Internet, Intranet, Local Area Networks (LAN) or Wide Area Networks (WAN). In addition, the connectivity to the network may be, for example, remote modem, Ethernet (IEEE 802.3), Token Ring (IEEE 802.5), Fiber Distributed Datalink Interface (FDDI) Asynchronous Transfer Mode (ATM), Wireless Ethernet (IEEE 802.11), or Bluetooth (IEEE 802.15.1). Note that computing devices may be desktop, server, portable, hand-held, set-top, or any other desired type of configuration. As used herein, the network includes network variations such as the public Internet, a private network within the Internet, a secure network within the Internet, a private network, a public network, a value-added network, an intranet, and the like. In embodiments of the present invention where the Internet is the backbone network **140**, gaming network **150** and remote network **120** may form a virtual private network (VPN) transported over the Internet.

[0053] In preferred embodiments, the remote network **120** may be a closed-loop network, such as the cable network depicted in **FIG. 10**. A closed-loop network **120** may have a limited geographic scope which allows the geographic location of a remote player device **110** to be identified. For example, a given cable network may be limited to a specific hotel. Each hotel room may be provided with a remote player device **110** which may then be identified with that location. In other embodiments, the remote network **120** may be a mobile telephone network which is capable of identifying a caller's geographic location.

[0054] As depicted in the simplified block diagram of **FIG. 3**, a remote player interface **300** may comprise a remote player device **110**, a display **310** for presenting game information and a control **320** to provide user game control for the remote player device **160**. In one embodiment, a remote player interface **110** may also comprise a remote control **395** to provide game controls. In preferred embodiments of the remote control, the connection **394** between the remote control **395** and the remote player device **160** may be any type of wireless connection,

including infra-red based protocols, or a RF wireless protocol such as Bluetooth (802.15.1). The remote control **395** may also be connected to the remote player device **160** through a wired connection such as Universal Serial Bus (USB), serial, or equivalent connection. The remote control **395** may also include controls customized for gaming. A handheld computer may also comprise a remote control **395**.

[0055] The display **310** may be a television, a personal computer, or a handheld computer device. A fixed or wireless telephone handset may comprise a display **310** and controls **320** of a remote player interface. In some embodiments the controls **320** may be integrated with display **310**, as for instance, in a touch screen.

[0056] In one embodiment, the game information may be a random number which represents the result of the game, information related to gaming device jackpots, or player credits. In another embodiment, the gaming information may be multimedia, sound and images, including, in one embodiment, video, representing the execution of a game. In another embodiment, game information may also be software for execution on a remote player device **110** or on any element of a remote player interface **300**, such as a remote control **395**, which interactively presents the game through the remote player interface **300**.

[0057] To enable regulatory conformance of the gaming system, gaming device users must be geographically within an approved jurisdiction and of legal age in the jurisdiction. In a regulated gaming environment, such as a gaming floor, physical control of the premises allows enforcement of this requirement. For remote player devices **110** not operated in the regulated gaming environment of a gaming floor, the age of the user of a remote player device **110** must be verified before game information is provided by a host gaming device **160**. Credentials may be received from a user using a variety of security devices and compared to records, such as in a database **170** to confirm identity and thus age of the user.

[0058] To ensure compliance with regulatory requirements, a gaming system **100** may identify the geographic location of a remote player device **110**. As discussed above, a network **120** may be a closed-loop network **120** whose devices are thereby identified in geographic location by the location of that network. Other embodiments may employ a GPS system on the remote player device **110** to provide the geographic location of the device **110**. In other embodiments, the remote network **120** may be a mobile communications network which provides the geographic location of network clients, such as a remote player device **110**.

[0059] In one embodiment, a security device may be a smart card reader **380** that is coupled to the remote player device **110**. In embodiments using a smart card reader, a user inserts a smart card into the reader which provides credentials sufficient to verify the age of the user. In

one such embodiment, indicia present on the smart card reader are compared to records in a casino database **170** to verify the age of the user.

[0060] In other embodiments, a remote player device **110** may be coupled to a biometric identity device **390**, such as a fingerprint scanner. In one embodiment, information received from the biometric identity device **390** may be compared to records in a casino database **170** to verify the age of the user. In other embodiments a biometric identity device **390** may be retinal scanner or facial recognition device.

[0061] In some embodiments, the controls **320** may include an input device (not pictured in **FIG. 3**) coupled to a remote player device **110** to receive a password or PIN as a security device. The password or PIN may be compared to information, such as records in a casino database **170** to verify the identity, and thus the age, of the remote player device user. For example, the input device may be a keyboard, rollerball, pen and stylus, mouse, or voice recognition system. The input device may also be a touch screen associated with an output device. The user may respond to prompts on the display by touching the screen. The user may enter textual or graphic information through the input device. The controls **320** may be coupled to a display **310** in the form of a personal computer, a television, a television with a set-top box, a handheld computer, or a telephone, fixed or mobile, handset.

[0062] Embodiments of a remote player device **110** may be a television, a cable interactive set-top box, a remote control, a personal computer, or a mobile or fixed telephone handset. Another embodiment may comprise a handheld computer coupled to a fixed or preferably wireless network. Also, a host gaming device **160** may also be a remote player device **110**.

[0063] In one embodiment, a remote gaming device **110** may be in a location approved by a gaming agency with controls **320** and display **310** which match the appearance of a stand-alone gaming device. For example, a remote gaming device **110** may be appear to be a slot machine with an arm control **320**, a mechanical or electronic "slots" display **310**. In other embodiments, remote gaming devices **110**, regardless of location, may have controls and displays which match the appearance of a host gaming device **160**. This may include control devices coupled to personal computers or set-top boxes which may be customized for one or more games.

[0064] Indicia of identity and age received from a smart card reader **380**, biometric identity device **390**, or user entry of a password may also be compared to records stored on the remote player device **110**. For example, a remote player device **110** in a hotel room may be programmed by hotel staff to store identification information for eligible guests in the room containing the gaming device without the identification information being included in the casino database **170**. In these embodiments, access to the remote player device thus may itself be an indicium of legal age to the central gaming controller **180** or host gaming device **160**.

[0065] A central gaming controller **180** may manage the interaction of remote player devices and host gaming devices. The central gaming controller **180** may comprise one or more server computers or may be integrated with a host gaming device. In the embodiment depicted in **FIG. 10**, the application server **1027** and request processing servers **1023** comprise the central gaming controller **180**.

[0066] One embodiment of a gaming system **100** comprises a single remote player on a remote player device **110** establishing a gaming session on a host gaming device **160** with no local player using the host gaming device **160**. In this embodiment, the local controls **220** of a host gaming device **160** become disabled for local play during the remote gaming session. Correspondingly, a host gaming device **160** in this embodiment also becomes unavailable for remote play while a player uses the local controls **220** to use the host gaming device **160**.

[0067] Another embodiment comprises a single player using the local controls **220** of a host gaming device **160** and a single remote player on remote player device **110** concurrently. Thus in this embodiment, the local game controls **220** on the host gaming device **160** are not disabled during the remote gaming session.

[0068] Another embodiment of the gaming system **100** comprises a single local player of the host gaming device **160** and multiple remote players on a plurality of remote player devices **110** having concurrent gaming sessions. A similar embodiment comprises multiple concurrent remote players and no local players on the host gaming device **160** because the local controls **220** may be disabled during the remote gaming sessions.

[0069] Another embodiment of a gaming system **100** comprises one or more remote player devices **110** which are physically located in a location approved by a gaming agency and networked to a host gaming device **160** that hosts both local and remote player sessions. Players physically located in the casino may occupy a remote player device **110** and play the games provided by the host gaming device **160**. Concurrently, gaming sessions to one or more remote player devices **110** physically located outside the casino may be provided. Thus, in this embodiment, players may concurrently play using the host gaming device **160**, a physically remote player device **110**, or a remote player device **110** in a location approved by a gaming agency.

[0070] Another embodiment of the invention comprises one or more remote player devices **110**, physically located in a location approved by a gaming agency and at least one host gaming device **160**. In this embodiment, player sessions may only be established on a host gaming device **160** from a remote player device **110** if that remote player device **110** is physically located in a location approved by a gaming agency, such as a casino gaming floor. Players may also play the host gaming device **160** using local controls **220** concurrently with remote player sessions.

Thus, in this embodiment, players may concurrently play using the host gaming device **160**, or a remote player device **110** that is located in a location approved by a gaming agency.

[0071] In each of the above disclosed embodiments, the remote player devices **110** that may concurrently receive game information from a host gaming device **160** may be limited to a predetermined number that is determined by a regulatory gaming agency for the jurisdiction.

[0072] A remote player device **110** that is physically located in the casino in a location approved by a gaming agency, such as a casino gaming floor, may differ from a remote player device physically located outside the casino floor. In one embodiment, a remote player device **110** located in a location approved by a gaming agency resembles the appearance of a stand-alone gaming device and may thus be similar in appearance and operation to the host gaming device **160**.

[0073] In one embodiment, a remote player device **110** requests game data from the host gaming device **160** by sending a request for a game to a central gaming controller **180**. The central gaming controller **180** then transmits the request for a game to the host gaming device **160**. The host gaming device **160** receives the request and provides game data to the central gaming controller **180** that passes to the remote player device **110**. That information is then translated into a game by the remote player device **110** and displayed or performed to the player. The remote player device **110** may contain on-board hardware and software that may be required to present a game. The regulated portion of hardware and software required to execute a game, such as a random number generator, is on the host gaming device **160** and the information transmitted to the remote player device **110** each time a game is requested.

[0074] Gaming devices according to an embodiment of the invention may use mixed-protocol delivery systems for game content and game results. Game information and results comprising image and sound data may be delivered by packet based network protocols such as IP datagrams, by connection-oriented network protocols, or by a combination of both. Streaming media protocols may also be employed. During a given gaming session, these communication methods may be used interchangeably or concurrently.

[0075] In one embodiment, communication over the data networks **120**, **140**, or **150**, may use IP datagrams to package image and sound data comprising a host gaming device interface and display, encrypts it, and delivers it to the remote player device.

[0076] Internet Protocol (IP) is a network layer protocol used by many corporations, governments, and the Internet worldwide. IP is a connectionless network layer protocol that performs addressing, routing and control functions for transmitting and receiving datagrams over a network. The network layer routes packets from source to destination. An IP datagram is a data packet comprising a header part and a data part. The header part includes a fixed-length header

segment and a variable-length optional segment. The data part includes the information being transmitted over the network. As a connectionless protocol, IP does not require a predefined path associated with a logical network connection. Hence, IP does not control data path usage. If a network device or line becomes unavailable, IP provides the mechanism needed to route datagrams around the affected area.

[0077] The remote player interacts with a game through a remote player interface **300**. A remote player device **110** may send commands back to the central gaming controller **180** as, in one embodiment, IP datagrams. The IP datagrams are interpreted by the central gaming controller **180** and used to proxy user interface interaction between the gaming device and the remote player. Game results may also be packaged as IP datagrams and delivered to the remote player through this method.

[0078] Alternative embodiments may use connection-oriented protocols such as TCP, or a combination of connection oriented protocols and connectionless packet protocols such as IP. Transmission Control Protocol (TCP) is a transport layer protocol used to provide a reliable, connection-oriented, transport layer link among computer systems. The network layer provides services to the transport layer. Using a two-way handshaking scheme, TCP provides the mechanism for establishing, maintaining, and terminating logical connections among computer systems. TCP transport layer uses IP as its network layer protocol. Additionally, TCP provides protocol ports to distinguish multiple programs executing on a single device by including the destination and source port number with each message. TCP performs functions such as transmission of byte streams, data flow definitions, data acknowledgments, lost or corrupt data re-transmissions, and multiplexing multiple connections through a single network connection. Finally, TCP is responsible for encapsulating information into a datagram structure.

[0079] Static content comprising the game interface or other elements of the game may be delivered to the remote player device **110** and stored on the remote player device. This delivery of content may use a mixed-protocol as described above. A static image may be a fixed image or an animation activated by the remote control device. Such images may further be overlaid with additional game content such as images and sound that is delivered dynamically during game play.

[0080] In an embodiment of the invention, a central gaming controller **180** converts image and sound data comprising the gaming device interface and display from the remote machine into a data stream (for example but not limited to MPEG-2), encrypts it, and delivers it to the remote player device **110**. The remote player interacts with the game using the remote player interface **300** to send commands back to the central gaming controller as IP datagrams. The IP datagrams may be interpreted by the central gaming controller **180** and used to proxy user interface

interaction between the gaming device **160** and the remote player device **110**. Game results may also be packaged as a data stream and delivered to the remote player through this method.

[0081] **FIG. 4** is a flowchart depicting a method employed when a command message is acknowledged by a central gaming controller **180** according to one embodiment of a gaming system **100**. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Note that in some embodiments, not all messages received by the central gaming controller **180** need be acknowledged. Starting at step **401**, a command message is sent to the central gaming controller **180** by a host on the network. The host may be remote player device **110** used for remote play, or other authorized network devices. Next, at step **405**, a qualified request message is received by the central gaming controller **180**. Moving to step **410**, the message is then recorded in a database. The database may be a casino database **170**. Proceeding to step **415**, the message is processed and a response prepared. Next at step **420**, the response is recorded in the database. Moving to step **425**, the response is sent back to the requesting device. At step **430**, a test to determine whether an acknowledgment of the message has been received is made. Continuing at step **435**, if the timeout value has passed control continues to step **440**, if the timeout period has not expired control returns to step **430**. Moving to step **440**, whether the message has not been acknowledged by the originating host is tested. If acknowledgement has been received, control proceeds to **445**, if not control proceeds to step **455**. At step **445**, the message status is recorded as "RECEIVED" and the process moves to the end state. Returning to step **455**, where the process flow continues following an unacknowledged message, the system sends a status request message to the sending host. Next, at step **460**, if the originating device responds to the message then flow continues to step **465**, otherwise control moves to step **480**. Moving to step **465**, a diagnostic message is sent to query whether the originating device is ready to receive the original message. Next at step **470**, if the originating host responds that it is ready to receive the original message, then control transfers to step **425** but if the originating host fails to respond then control moves to step **480**. Moving to step **480**, the status of the originating host is set to offline until such time as the originating host can respond or reinitializes, and the process moves to the end state.

[0082] **FIG. 5** is a flowchart depicting a method used when a request for a remote gaming session is received, when playing a game, and when terminating the remote gaming session. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Starting at **510**, a request for a remote gaming session is received as a request for a secured encrypted connection to the central gaming controller **180**. Included in the request are the remote players security credentials in the form of a security certificate, for example, X.509 certificate. Next at **515**, the security credentials are authenticated.

This authentication may be performed by submitting the security certificate to a certificate authority for authentication. Moving to **520** if the player is not authenticated, control reverts to **515**. Continuing to step **525**, the central gaming controller **180** establishes a secure encrypted connection with the remote player device **110**. Next, at step **530**, if required the player transfers funds to use during the remote gaming session. Continuing to step **535**, the player then chooses a host gaming device **160** to play. Next, at step **540**, in one embodiment, when a host gaming device **160** is chosen for remote access play the local controls of the host gaming device **160** is disabled to prevent local play. Moving on to step **545**, a remote play session is opened on the host gaming device **160**. Continuing at step **550**, after a remote gaming session is established on the host gaming device, the central gaming controller **180** sends a message to the host gaming device **160** instructing it to displace representations of its user controls, graphics and sounds to the remote player interface **300**. The central gaming controller **180** directs the host gaming device **160** controls over the secured encrypted connection and manages the remote gaming session. Next at step **555**, the remote player may transfer funds from a player account to the host gaming device **160** for wagering on the host gaming device **160**. Moving to step **560**, a wager is made. Next at, **656** a game is played. Continuing to step **570**, the central gaming controller **180** delivers the results of the game to the remote player interface **300**. Next at step **571**, the remote player may repeat the sequence from step **560**. Next at step **575**, if there are any credits on the host gaming device **160** when the player terminates the remote gaming session, the central gaming controller **180** automatically transfers those credits back to the players account. Moving to step **580**, the central gaming controller **180** terminates the remote gaming session with the host gaming device **160**. Continuing to step **585**, the central gaming controller **180**, enables local play on the host gaming device **160**, control is then transferred to the end state.

[0083] **FIG. 7** is a flowchart depicting a method for a host gaming device **160** to become connected to a network using security certificates and a certificate authority. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Starting at **705**, a host gaming device **160** starts the process of connecting to a network as part of its initialization mode. Continuing to step **720**, at a point during initialization, the host gaming device **160** submits a security certificate to a certificate authority for authentication. Moving to step **725**, the certificate authority authenticates the certificate. Next at step **730**, if the certificate is authenticated control moves to step **740**, otherwise control moves to step **735**. Continuing on to step **740**, the host gaming device **160** is permitted onto the network and the process moves to its end state. Returning to step **735**, if the certificate is not authenticated then a log entry is generated and the host gaming device **160** is not permitted onto the network.

[0084] Embodiments according to the invention may also use instant messaging and/or email messaging systems. Typical instant messaging systems permit computer users to type text messages and add file attachments into a host program and have the host program automatically deliver the text through a virtual direct connection to a target computer. Public email systems are those available for general use, as over the internet. Examples of public instant messaging systems in use today include but are not limited to chat programs like IRC, MSN Messenger, AOL Instant Messaging and a host of others. Private systems are restricted to a casino or gaming system. Typical email messaging systems permit messages and file attachments to be entered into a host program and addressed to a specific recipient on a network. These messages may not be delivered directly to the addressee, but are sent to a storage area where the recipient may retrieve the message at a time of their own choosing.

[0085] Gaming devices **160** and remote player devices **110** routinely exchange information with a central gaming controller **180** for, typically, but not limited to, account and game tracking functions. In one embodiment of the invention, devices may send and receive data over public and/or private email-type messaging systems. The message body of any particular message may vary, using a proprietary or non-proprietary format, and may be encrypted or in human-readable format. Messages may be sent at a time determined by the message originator, typically, but not exclusively, in response to an event. The recipient of the message may be any device capable of consuming the message. The message recipient may be responsible for checking the prescribed message storage area for messages addressed to it. The message recipient may reply to a received message or may generate a new message to a specific recipient, a group of recipients, or all recipients connected to the system. Remote player devices **110** may periodically check for new messages in the system and process them.

[0086] According to one embodiment of the invention, gaming devices **160** may send and receive data over public and/or private instant messaging systems. The message body of any particular message may vary, using a proprietary or non-proprietary format, and may be encrypted or in human-readable format. Messages may be sent at a time determined by the message originator, typically, but not exclusively, in response to an event. The recipient of the message may be any device capable of consuming the message. Both the gaming device **160** and the message recipient may queue incoming and outgoing messages. Queuing messages permits devices involved in instant message communications to accept new messages while processing received messages and to generate outgoing messages for delivery as system resources permit.

[0087] In another embodiment according to the invention, devices may send and receive data over public and/or private email-type messaging systems. The message body of any particular message may vary, using a proprietary or non-proprietary format, and may be encrypted or in human-readable format. Messages may be sent at a time determined by the message

originator, typically, but not exclusively, in response to an event. The recipient of the message may be any device capable of consuming the message. The message recipient may be responsible for checking the prescribed message storage area for messages addressed to it. The message recipient may reply to a received message or may generate a new message to a specific recipient, a group of recipients, or all recipients connected to the system. Gaming system devices **110** and **160** may periodically check for new messages in the system and process them.

[0088] Embodiments according to the invention may present promotional messages during remote play sessions. Messages sent may comprise instant messages for promotional information, notification of events, or other pieces of information that can be communicated electronically. Promotional messages may also include jackpot and bonus information. A promotional message server may be used to construct and send promotional messages. In one embodiment, a computer server, comprising a central gaming controller **180,** may also comprise the promotional message server.

[0089] A user interface may be provided to construct message templates. These templates are then used to construct a deliverable message. Embodiments of a message template may comprise a timeout value that indicates how long the message is to be displayed, the frequency with which the message displays in relationship to other scheduled messages, a limitation value that prevents the message from being displayed too often and an expiration date after which the message is no longer used in the system. Custom graphics and display modes may also be specified for a message template, such as icons, animations, and various scrolling methods.

[0090] A remote player device **110** may present a promotional message for an amount of time determined from the contents of the promotional message. The promotional message may be presented to a user in conjunction with gaming information. The presentation may contain icons, animations, and various scrolling methods. In addition multimedia such as sound and video may be utilized.

[0091] The promotional message server may also provide a dynamic data insertion function to insert player information such as the player's name or birthday into a message prior to delivery. Dynamic data insertion may be accomplished through the use of specialized tags within the message body. When encountered, the tag characters within the message are replaced with data from a related data source. The specific tag's character sequence is associated with a specific subset of the data in the data source, such as a player's name in a data source of player information. Processing comprises reading the data source and its subsets, parsing the specialized tags from the message template, indexing the data source and replacing the tag characters with data from the data source to create a deliverable message for each item in the data source. This sequence continues until all the data in the data source has been included in messages. The messages may be delivered

as they are created or queued until all items in the data source have been used to create messages, then all messages may be sent at the same time.

[0092] In one embodiment, a gaming system **100** may comprise a card reader installed in a gaming device **280** or remote player device **380**. Promotional messages may be based on information obtained about a player that is either stored on a card inserted into the card reader or by using identifying information from the card to access the casino's proprietary database systems **170**.

[0093] One embodiment of the promotional message server may also provide a dynamic grouping function in which a subset of players currently gaming is selected and collected into a group. Casino operators may address a message template to this dynamic subset of current players and send a specific message or messages exclusively to that subset. These messages may be constructed using the dynamic data function. The dynamic grouping function may use criteria specified by the casino and available in the casino's proprietary database systems **170** and criteria generated by live gaming activity to establish a profile that players must meet to be selected. The criteria may comprise loyalty points the player has earned, a player's birthday, length of current gaming session, or other data that is collected by the casino on players and gaming activity.

[0094] The dynamic grouping function may be scheduled to run at time intervals determined by the casino. Each time the interval is reached the promotional gaming server searches for current players that meet the established criteria and builds a dynamic group then sends the assigned message to that group of players exclusively. The gaming devices **160**, remote player device **110**, card readers installed in gaming devices **280** and remote player device **380**, and casino proprietary database systems **170** may provide data to search for players that meet the specified criteria and assemble them into a dynamic group.

[0095] In one embodiment of the invention, the casino may advertise a casino sponsored event. The casino may use a user interface display to construct the message and schedule its delivery start time, duration of the message e.g. number of hours, days, weeks, or months that the message will run, and specific values that weight the message's delivery interval and frequency amongst other promotional messages scheduled in the system. The style of message may also be specified, including but not limited to flashing, scrolling, scroll direction, and the use of custom graphics. The casino operator may also specify the criteria players must meet to receive the message. Once the casino operator accepts the promotional message configuration, the promotional message server may deliver the message across a network to remote player devices **110** or host gaming systems **160**.

[0096] An embodiment of a gaming system **100** may provide for the electronic transfer of funds to a gaming device for the purpose of making wagers. When a player chooses a gaming device **160** to play remotely, funds are electronically transferred to the gaming device and

appear as credits on the gaming device **160**. The player then uses those credits to make wagers on game outcome. When the player is finished, the system transfers any remaining credits on the gaming device back to the source of funds or to an alternate storage. Limitations on the amount of funds transferred may be set for a minimum or maximum amount transferred, a minimum or maximum amount transferred within a given time period, or a minimum or maximum amount transferred for the life of the account, or a combination of any of these. The limitation may also vary between accounts, permitting one account to have a different limitation on transfers than another. When the limitation set is reached, further transactions are prevented until the limitation is resolved. The limitation may be set voluntarily by the player, by the casino, or by a gaming authority. Limitations may be set for all players within a specific jurisdiction or for selected players only. The source of funds used by a player for remote access play may be maintained in a database located on a computer that is directly or indirectly connected to the casino network **150**.

[0097] **FIG. 6** is a flowchart depicting an embodiment of the invention whereby a player transfers funds from a bank account to a player account for the purpose of wagering on games. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Starting at step **601**, a remote player device **110** initiates an electronic funds transfer. Continuing to step **605**, the central gaming controller **180** verifies the remote players banking information. Next at step **610**, if the banking information is valid, control transfers to step **620**, otherwise control moves to step **615**. Continuing at step **620**, the remote player device **110** prompts the player to enter the amount of the transfer. Moving to step **615**, the central gaming controller **180** verifies fund availability. Next at step **630**, if funds are not available control moves to step **615**. Otherwise, control moves to step **635**, where, in a one embodiment, the central gaming controller **180** may consult a casino database **170** and determine whether the remote players total gaming activity exceed limits placed on that activity. Next at step **640**, if the limit is reached control moves to step **615**. Otherwise, continuing at step **645**, the transfer is completed. Returning to step **615**, if the players banking information is not correct, funds are not available or a transfer limit is reached, then the transaction is canceled and control transferred to the end state.

[0098] An embodiment of a gaming system **100** may record the interaction between remote players and host gaming devices **160** during remote gaming sessions for the purpose of resuming games in-progress after a communications failure. If at anytime the connection between the remote player and a gaming device becomes unavailable, the system has a sufficient record of player positions to restart the game as at the time just prior to the failure. Thus an embodiment of a gaming system may record, transfer, and reinstate on a like device an encrypted block of data representing the precise state of a particular gaming device **160** at the time that the data block is requested. The encrypted block of data is generated by the gaming device **160** and transferred

using a communication protocol. The encrypted block of data may be used to continue a game in-progress that was interrupted by a gaming device **160** failure or other system failure. In addition, the payer's wager and credit data along with gaming payout data may be included in the data block. The data may also be transported to another gaming device **160** for the purpose of completing an interrupted game or resuming a gaming session. The destination gaming device **160** receives the encrypted block of data, decrypts it, and loads the game state into its own systems, allowing a game in-progress to complete or a game session to continue.

[0099] **FIG. 8** is a flowchart depicting a method for a gaming device **160** to build and deliver an encrypted block of data representing the complete state of the gaming device. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Starting at **805**, a central gaming controller **180** sends a message to a host gaming device **160** to initiate the build of the encrypted data block. Continuing to step **10**, the gaming device responds with an acknowledgement. Next, at step **815**, the gaming device **160** begins the build process. When finished with the build and encryption process, at step **820**, the gaming device saves the data block to non-volatile memory in the gaming device. Continuing to step **825**, the gaming device **160** sets an indication that may be queried by the central gaming controller **180** as to the status of the build/encryption process. Moving to step **830**, the central gaming controller **180** checks the gaming device's status. Next at step **835**, if the build/encryption process is complete, control continues to step **840**, otherwise control returns to step **830**. Moving to step **840**, the central gaming controller **180** retrieves the data block from the gaming device **160**. Next, at step **845**, when the central gaming controller **180** has retrieved the data block it saves the data block to a database. Continuing to step **850**, the central gaming controller then checks the validity of the saved data block. If the data block is not verified then the central gaming controller initiates another retrieval by returning control to step **840**.

[0100] **FIG. 9** is a flowchart depicting a method for retrieving an encrypted block of data representing the state of a gaming device from a database and loading the encrypted block into a gaming device. Depending on the embodiment, additional steps may be added, others removed, steps merged, or the order of the steps rearranged. Starting at step **905**, the central gaming controller **180** retrieves a saved encrypted data block from the database. Next at **910**, the controller **180** verifies the integrity of the data block. Continuing to **915**, if the data block is verified, control continues to step **925**, if not control moves to step **920**. Returning to the flow of control at **925**, the central gaming controller **180** notifies a target gaming device **160** of an intent to upload the data block. Next, at step **930**, the target gaming device **160** responds with a message indicating whether it is available for the upload. Moving to step **935**, if the target device is ready control moves to step **940**, if not control is diverted to step **920**. Returning back to step **940**, the encrypted data

block is uploaded to the target gaming device **160**. Next at step **945**, the target gaming device **160** verifies the encrypted data block. Moving on to step **950**, if the data block was verified, the gaming device moves on to step **955**, if not verified, control moves to step **920**. Continuing on to step **955**, the gaming device **160** initializes its state to the new state defined by the received data block and the process moves to the end state. Returning back to step **920**, which is reached on error conditions, an error log entry is generated and the requesting process notified.

[0101] **FIG. 10** is a block diagram depicting one embodiment of a gaming system according to the present invention wherein the host gaming devices **160** are available for remote play over a network that connects to a cable modem termination system. The cable modem termination system **1005** is located at the head-end of a cable television provider who makes broadband network connectivity available as a service to its customers. Cable television customers who subscribe to broadband or digital television services access the remote network **120** through a digital home communications terminal (DHCT) **1000**. The remote player device **110** may be a stand-alone cable modem or a set-top box that includes a cable modem and a digital television broadcast decoder. The DHCT **1000** may, in some embodiments include the remote player device **110**. The remote player interface **300** may be any device or combination of devices that remote players operate to interact with the remote player device **110**, for example, a television with remote control or a personal computer. To connect to the central gaming controller **180**, a remote player uses the remote player device **110** to send messages, using, in one embodiment, IP datagrams, through the DHCT and the cable modem termination system **1005**. The cable modem termination system **1005** uses a network router **1004** to route the IP datagrams over a network connection **140** to the central gaming controller **180**. The backbone network connection **140** can be any type of network connection such as a dedicated T1 or fiber optic over which network traffic can be exchanged. In preferred embodiments the backbone network **140** is part of a closed loop network. However, in other embodiments, a public network such as the Internet may form at least a portion of the backbone network. Encryption of the data may be performed, either at the endpoints such as remote player device **110**, at a host gaming device **160**, at a central gaming controller **180**, over network **120**, or only over network **140**.

[0102] Network traffic from the remote network **120** and backbone network **140** travels over a number of virtual local area networks (VLAN) configured using a multilayer network switch **1022**. Segmenting the internal network into VLANs creates security zones whereby only permitted network traffic appears on a given VLAN.

[0103] IP datagrams are received over the backbone network **140** through network router **1020** and firewall **1021**. Network router **1020** filters IP datagrams that are not coded with the configured port for access to the gaming network **150**. If an IP datagram passes the network

router **1020** it then must pass the firewall **1021** in order for the IP datagram to be processed by the request processing server(s) **1023** which comprise a portion of a central gaming controller **180** in this embodiment.

[0104] The firewall **1021** has two network interfaces **1050, 1051**; the external-facing network interface **1050** is connected to the router **1020** and the internal-facing network interface **1051** is connected to the multilayer network switch **1022**. In this configuration the firewall **1021** acts as a type of network switch that may perform additional security checks on the IP datagram, then move the datagram to the internal-facing network interface **1051** where the multilayer network switch **1022** moves the datagram to the VLAN where request processing server(s) **1023** are located.

[0105] Each request processing server **1023** has two network interfaces **1052, 1053**, both connected to the multilayer network switch **1022**. Each network interface **1052, 1053** may be configured on a different VLAN of the multilayer network switch **1022**. The multilayer network switch **1022** moves IP datagrams between the firewalls **1021** internal-facing network interface **1051** and the request processing server(s) **1023** external-facing network interface **1052**. This embodiment provides a layer of protection for the host gaming devices **160** in the event that the request processing server(s) **1023** are compromised.

[0106] When an IP datagram arrives at a request processing servers **1023** external-facing network interface **1052**, the request processing server **1023** interprets the IP datagram and issues commands over its internal-facing network interface **1053** to the application server **1027**. The request processing server **1023** may reject invalid commands or make other determinations as to the appropriateness of a request that prevent the request from being passed on to the application server **1027**. Likewise, the request processing server **1023** may request data from the application server for use in building its own response to the request, which may or may not require an acknowledgement from the remote player device **110** as described below.

[0107] Command messages received by the application server **1027** may be recorded in a database using the database server **1025**. The application server **1027** then executes the command, which may include any function relevant to the operation of the host gaming device **160** and may or may not return data to the request processing server **1023** for delivery to the remote access player. In one embodiment, the database server **1025** may comprise the casino database **170**. In other embodiments the database server **1025** and the application server **1027** may comprise the casino database **170**.

[0108] Some commands may require the remote player device **110** to acknowledge the receipt of information sent from the central gaming controller **180**. For commands that require acknowledgement, the central gaming controller **180** queues the status of the messages that are sent to the remote player device **110**. The status of messages sent but not acknowledged is stored in a

database as "open" using the database server **1025**. When the remote player device **110** receives the message it sends an acknowledgment message back to the central gaming controller, which in turn marks the message in the database as "closed"; indicating that the message has reached its destination and has been acknowledged. If the message is not acknowledged within a specified timeout, the message is resent. **FIG. 4** depicts the sequence of events for the receipt, queuing and response loop for qualifying messages.

[0109] Recording of messages between the remote player device **110** and a host gaming device **160** by the central gaming controller **180** allows each game or transaction, on both the host gaming device **160** and remote player device **110**, to be recorded. This allows each host gaming device or remote player device to be individually auditable using standard accounting practices in the gaming jurisdiction where the game is located. In one embodiment, a third party, such as a gaming authority may be sent the records of games and transactions online by the gaming system **100**.

[0110] When the application server **1027** receives a command request that requires communication with gaming devices **160, 161, 162** it connects to those devices using terminal server **1035**. Terminal server **1035** provides Ethernet connectivity to the RS232 serial interface **1054** of the game. Through that interface the remote player device **110** communicates to the gaming devices **160, 161, 162** using a communications protocol supplied by the gaming machine manufacturer. The protocol includes commands that permit the remote operation of the gaming devices **160, 161, 162** and the reporting of game results so that the application server **1027** can control remote play.

[0111] **FIG. 11** depicts a more detailed network diagram of one embodiment of network **150** and elements of a gaming system **100** connected to network **150**. This includes a host gaming device **160**, and a database **160**. As in the embodiment of **FIG. 10**, a central gaming controller **180** may be comprised of request processing servers **1027** and an application server **1023** connected to one or more VLANs of network **150**.

[0112] While the above detailed description has shown, described, and pointed out novel features of the invention as applied to various embodiments, it will be understood that various omissions, substitutions, and changes in the form and details of the device or process illustrated may be made by those skilled in the art without departing from the spirit of the invention. The scope of the invention is indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within their scope.

WHAT IS CLAIMED IS:

1. A gaming system comprising:

a data network, wherein the data network is comprised of at least one logical segment, wherein at least one logical segment is a closed-loop network;

a host gaming device connected to the data network, the gaming device configured to execute at least one game wherein the host gaming device in a location approved by a gaming agency;

a plurality of remote player devices connected to the closed-loop network; and

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device and on each of the plurality of remote player devices,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device, and

wherein the host gaming device is configured to provide game information to a predetermined number of permitted remote player devices.

2. A gaming system comprising:

a data network;

a host gaming device connected to the data network, the gaming device configured to execute at least one game; and

a plurality of remote player devices connected to the data network,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device,

wherein the host gaming device is configured to provide game information to a predetermined number of permitted remote player devices, and

wherein at least one of the plurality of remote player devices is permitted based upon, at least in part, the geographic location of the remote player device.

3. The system of Claim 2, wherein the predetermined number is determined by a gaming agency.

4. The system of Claim 2, wherein at least one of the plurality of remote player devices is permitted based upon, at least in part, an age of a user of the remote player device.

5. The system of Claim 2, wherein the data network is, at least in part, the Internet.

6. The system of Claim 2, wherein the data network is comprised of at least one logical segment.

7. The system of Claim 6, wherein at least one logical segment is a closed-loop network.

8. The system of Claim 6, wherein the host gaming device is configured to identify the geographic location of a remote player device based, at least in part, on a logical segment corresponding to the remote player device.

9. The system of Claim 2, wherein the host gaming device is configured to identify the geographic location of a remote player device based, at least in part, on information provided by a mobile communications network.

10. The system of Claim 2, wherein the host gaming device is configured to identify the geographic location of a remote player device based, at least in part, on information provided by a GPS device.

11. The system of Claim 2, wherein the data network is, at least in part, the casino intranet.

12. The system of Claim 2, wherein the data network is, at least in part, the hotel intranet.

13. The system of Claim 2, wherein the data network is, at least in part, a wireless network.

14. The system of Claim 2, wherein the host gaming device is in a location approved by a gaming agency.

15. The system of Claim 2, wherein the host gaming device includes at least one game control configured to provide local use.

16. The system of Claim 15, wherein the host gaming device is configured to disable local use when the host gaming device is providing game information to a remote player device.

17. The system of Claim 2, wherein each of the remote player devices is in a location approved by a gaming agency.

18. The system of Claim 2, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

19. The system of Claim 2, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

20. The system of Claim 2, wherein the gaming information is, at least in part, software.

21. The system of Claim 2, wherein at least one remote player device is coupled to a credential device configured to receive information relating to a user of the remote player device.

22. The system of Claim 21, wherein the information relating to the user is an age of the user.

23. The system of Claim 21, wherein the information relating to a user is a password that is input by the user.

24. The system of Claim 21, wherein the credential device is an input device configured to receive a password from the user.

25. The system of Claim 21, wherein the credential device is a smart card reader.

26. The system of Claim 21, wherein the credential device is a biometric device.

27. The system of Claim 28, wherein the biometric device is a fingerprint reader.

28. The system of Claim 21, further comprising: a database configured to provide information associated with each of a plurality of users of the gaming system.

29. The system of Claim 28, wherein the information associated with a user includes a password.

30. The system of Claim 28, wherein the information associated with a user includes an age of the user.

31. The system of Claim 28, wherein the information associated with a user includes information relating to a fingerprint of the user.

32. The system of Claim 2, wherein the host gaming device is configured to encrypt the game information.

33. The system of Claim 2, wherein the game information is provided via a public email system.

34. The system of Claim 2, wherein the game information is provided via a private email system.

35. The system of Claim 2, wherein the game information is provided through a public messaging system.

36. The system of Claim 2, wherein the game information is provided through a private messaging system.

37. A gaming system comprising:

a data network;

a host gaming device in a location approved by a gaming agency connected to the data network, the gaming device configured to execute at least one game; and

a plurality of remote player devices connected to the data network.

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device, and

wherein the host gaming device is configured to disable local use of the gaming device when providing game information to the remote player devices.

38. The system of Claim 37, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

39. The system of Claim 37, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

40. The system of Claim 37, wherein the host gaming device is configured to allow no more than a predetermined number of remote player devices to concurrently receive game information provided by the host gaming device.

41. A gaming system comprising:

gaming means for executing at least one game, the game providing game information during execution;

local access means for providing local access to the game information for a user in a location approved by a gaming agency;

player means for receiving game information, presenting game information and providing at least one game control;

means for providing the game information over a data network to a predetermined number of receiving means;

means for determining the location of the receiving means; and

means for disabling the local access means.

42. The system of Claim 41, further comprising:

a means for creating an auditable record of gaming transactions on the gaming means.

43. The system of Claim 41, further comprising:

a means for creating an auditable record of gaming transactions on the playing means.

44. The system of Claim 41, wherein the predetermined number is determined by a gaming agency.

45. The system of Claim 41, further comprising:

means for receiving information associated with a user of the gaming system.

46. The system of Claim 45, wherein the information associated with the user includes the age of the user.

47. The system of Claim 45, wherein the means for receiving information associated with a user is a smart card reader.

48. The system of Claim 45, wherein the means for receiving information associated with a user is a biometric identity device.

49. The system of Claim 45, wherein the means for receiving information associated with a user is a keyboard configured to receive a password.

50. The system of Claim 45, wherein the user information includes, at least, a credential for authentication of the user.

51. The system of Claim 50, further comprising:

means for authenticating the credential coupled to means for limiting access to the gaming system.

52. A method of remotely accessing a host gaming device on a remote player device comprising:

establishing access to the host gaming device from the remote player device through a data network;

receiving gaming related information from the host gaming device through the data network;

presenting the gaming related information to a player;

receiving at least one control signal from the player;

sending the control signal to the host gaming device through the data network; and

disabling local use of the host gaming device.

53. The method of Claim 52, further comprising:

recording each gaming transaction occurring on the remote player device.

54. The method of Claim 52, further comprising:

providing a geographic location of the remote player device.

55. The method of Claim 52, further comprising:

providing information relating to a user of the remote player device to the gaming device.

56. The method of Claim 55, wherein the information relating to a user includes, at least, the age of the user.

57. The method of Claim 52, further comprising:

allowing no more than a predetermined number of remote player devices to concurrently establish a gaming session on the gaming device.

58. A method of providing remote access to a host gaming device comprising:

verifying a geographic location of a remote player device;

establishing a gaming session on a host gaming device from a remote player device through a data network;

receiving at least one control signal from the remote player device through the data network;

sending gaming related information from the gaming device through the data network;

59. The method of Claim 58, further comprising:

recording each gaming transaction occurring on the host gaming device.

60. The method of Claim 58, further comprising:

receiving information relating to a user of the remote player device on the gaming device.

61. The method of Claim 60, wherein the information relating to a user includes, at least, the age of the user.

62. The method of Claim 58, further comprising:

disabling local access to the gaming device.

63. The method of Claim 58, further comprising:

allowing no more than a predetermined number of remote player devices to concurrently establish a gaming session on the gaming device.

64. A method of resuming an interrupted gaming session on a first host gaming device comprising:

generating a gaming state of the gaming session on the first gaming device;

encrypting the gaming state;

transporting the encrypted gaming state from the first gaming device;

transporting the encrypted gaming state to a second gaming device;

decrypting the gaming state on the second gaming device; and

loading the game state into a second gaming device to resume the gaming session.

65. A gaming system comprising:

a data network;

a first host gaming device connected to the data network, the gaming device configured to:

execute at least one game,

generate a gaming state based on execution of at least one game;

encrypt the gaming state; and

send the encrypted gaming state over the data network;

a second host gaming device connected to the data network, the gaming device configured to:

receive the encrypted gaming state over the data network;

decrypt the gaming state;

resume executing at least one game from the gaming state; and

a plurality of remote player devices connected to the data network,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device.

66. The system of Claim 65, wherein the remote player devices are each configured to receive an encrypted gaming state from a first gaming device over the data network and to send the encrypted gaming state to the second gaming device.

67. The system of Claim 66, wherein the first gaming device is the second gaming device.

68. The system of Claim 65, wherein the second gaming device is configured to receive an encrypted gaming state from a first gaming device over the data network.

69. The system of Claim 65, wherein the gaming state includes user payment information.

70. The system of Claim 65, wherein the gaming state includes gaming machine payout information.

71. The system of Claim 65, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

72. The system of Claim 65, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

73. A gaming system comprising:

means for executing at least one game;

means for generating a gaming state based on execution of at least one game;

means for encrypting the gaming state;

means for sending the encrypted gaming state;

means for receiving the encrypted gaming state;

means for decrypting the gaming state; and

means for resuming executing at least one game from the gaming state.

74. The system of Claim 73, wherein the gaming state includes user payment information.

75. The system of Claim 73, wherein the gaming state includes gaming machine payout information.

76. The system of Claim 73, further comprising:

a means for creating an auditable record of gaming transactions on the host gaming device.

77. The system of Claim 73, further comprising:

a means for creating an auditable record of gaming transactions on each of the plurality of remote player devices.

78. A method of authenticating a user of a host gaming device comprising:

receiving a security certificate from the smart card;

sending the security certificate to a certificate authority for authentication;

receiving an authentication reply from the authority; and

playing a game in response to the authentication reply.

79. A method of authenticating a user of a remote player device comprising:

receiving an indicia of identity for a user;

sending the indicia of identity to an authenticator device;

receiving an authentication reply from the authenticator device; and

authorizing use of a host gaming device based on the indicia of identity

80. The method of Claim 79, wherein the indicia of identity for a user is provided by a biometric identity device.

81. The method of Claim 79, wherein the indicia of identity for a user is provided by a password input by the user.

82. The method of Claim 79, wherein the indicia of identity for a user is provided by a smart card.

83. A gaming system comprising:

a data network;

a host gaming device interfaced to the data network;

a plurality of remote player devices interfaced to the data network; and

a security device configured to provide player credentials to at least one remote player device,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device,

wherein the host gaming device is configured to provide game information to a predetermined number of permitted remote player devices, and

wherein at least one of the plurality of remote player devices is permitted based upon, at least in part, on player credentials provided by the security device.

84. The system of Claim 83, wherein the security device is a smart card reader.

85. The system of Claim 83, wherein the security device is a biometric device.

86. The system of Claim 83, wherein the security device is an input device.

87. The system of Claim 86, wherein the player credentials are, at least in part, a password.

88. The system of Claim 83, wherein the remote player device is authorized to receive game information provided by the host gaming device based, in part, on the player credentials.

89. The system of Claim 83, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

90. The system of Claim 83, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

91. A method of remotely accessing a gaming device comprising:

establishing a gaming session on a gaming device for a remote player device through a data network;

sending gaming related information from the gaming device through the data network;

receiving at least one control signal from the remote player device through the data network.

creating an auditable gaming session record representing each gaming transaction of a gaming session on the host gaming device;

creating an auditable gaming session record representing each gaming transaction of a gaming session on the remote gaming device; and

sending the record to a third party through the data network.

92. The method of Claim 91 wherein the third party is a gaming authority.

93. A gaming system comprising:

a data network comprised of a plurality of logical segments wherein a security policy controls the flow of data between logical segments;

a host gaming device connected to the data network, the gaming device configured to execute at least one game; and

a plurality of remote player devices connected to the data network,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device, and

wherein the plurality of remote player devices are each configured to control a gaming session established on the gaming device subject to the security policy wherein the security policy is based, at least in part, on the geographic location of a logical segment.

94. The system of Claim 93, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

95. The system of Claim 93, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

96. A gaming system comprising:

a data network;

a promotional message server configured to provide customized promotional messages wherein each message is customized with information associated with a user of the gaming system;

a host gaming device interfaced to the data network; and

a plurality of remote player devices interfaced to the data network,

wherein the plurality of remote player devices are each configured to receive game information provided by the host gaming device and to receive and present promotional messages.

97. The system of Claim 96, wherein the remote player devices are in a location approved by a gaming agency.

98. The system of Claim 96, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

99. The system of Claim 96, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

100. The system of Claim 96, wherein promotional message are comprised of bonus information.

101. The system of Claim 96, wherein promotional message are comprised of jackpot information.

102. The system of Claim 96, further comprising: at least one database configured to provide information associated with a plurality of users of the gaming system.

103. The system of Claim 96, wherein each of the plurality of remote game devices is associated with a user.

104. The system of Claim 96, further comprising a smart card reader configured to provide information associated with a user of the gaming system.

105. The system of Claim 102, wherein the database is configured to provide information which forms, at least in part, the content of the promotional message.

106. The system of Claim 96, wherein each of the plurality of remote player devices is configured to receive and present the promotional message in conjunction with game information provided by the host gaming device.

107. The system of Claim 106, wherein each of the plurality of remote player devices is configured to present the promotional message for an amount of time.

108. The system of Claim 106, wherein the amount of time is based, at least, in part on information associated with the promotional message.

109. The system of Claim 102, wherein the database is configured to provide information which comprises, at least in part, the content of the promotional message.

110. The system of Claim 96, wherein the promotional messages are transported via an instant messaging system.

111. The system of Claim 96, wherein the promotional messages are transported via an email system.

112. A method of displaying information on a remote player device comprising:

receiving a promotional message on a remote player device;

presenting the promotional message in conjunction with gaming information for an amount of time; and

removing the promotional message from the remote player device.

113. The method of Claim 112, further comprising

calculating the amount of time based, at least in part, on information associated with the promotional message.

114. A gaming system comprising:

means for data communication;

means for executing at least one game;

means for providing game information over the data network to a predetermined number of receiving means; and

a plurality of means for receiving game information over the data communication means, each coupled to a means for receiving customized promotional messages.

115. The method of Claim 114, further comprising:

means for presenting customized promotional messages in conjunction with game information.

116. The method of Claim 114, further comprising:

means for sending promotional messages.

117. The method of Claim 114, further comprising:

means for providing data used to select which players receive customized promotional messages.

118. The method of Claim 114, further comprising:

means for providing data which forms, at least in part, the content of promotional messages.

119. The system of Claim 114, further comprising:

a means for creating an auditable record of gaming transactions on the host gaming device.

120. The system of Claim 114, further comprising:

a means for creating an auditable record of gaming transactions on each of the plurality of remote player devices.

121. A gaming system comprising:

a data network;

a host gaming device interfaced to the data network;

at least one remote player device interfaced to the data network;

a video display device in communication with the remote player device; and

a remote control device in communication with the remote player device,

wherein the remote player device is configured to receive game information provided by the host gaming device and the remote control device is configured to control operation of a game.

122. The system of Claim 121, wherein the video display device is a television.

123. The system of Claim 121, wherein the video display device is a computer.

124. The system of Claim 121, wherein the video display device is a control device.

125. The system of Claim 121, wherein the remote player device is coupled to a cable television system.

126. The system of Claim 121, wherein the data network is, at least in part, the Internet.

127. The system of Claim 121, wherein the data network is, at least in part, the casino intranet.

128. The system of Claim 121, wherein the data network is, at least in part, the hotel intranet.

129. The system of Claim 121, wherein the data network is, at least in part, a wireless network.

130. The system of Claim 121, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on the host gaming device.

131. The system of Claim 121, further comprising:

a central gaming controller configured to create an auditable record of gaming transactions on each of the plurality of remote player devices.

132. A method of remotely accessing a host gaming device comprising:

establishing a gaming session on the host gaming device from a remote player device through a data network;

receiving gaming related information from the host gaming device through the data network;

presenting gaming related information to a player via a video display device;

receiving at least one control signal generated by a remote control device for controlling the gaming session; and

sending the control signal to the host gaming device through the data network.

133. The method of Claim 132, further comprising:

recording each gaming transaction occurring on the remote player device.

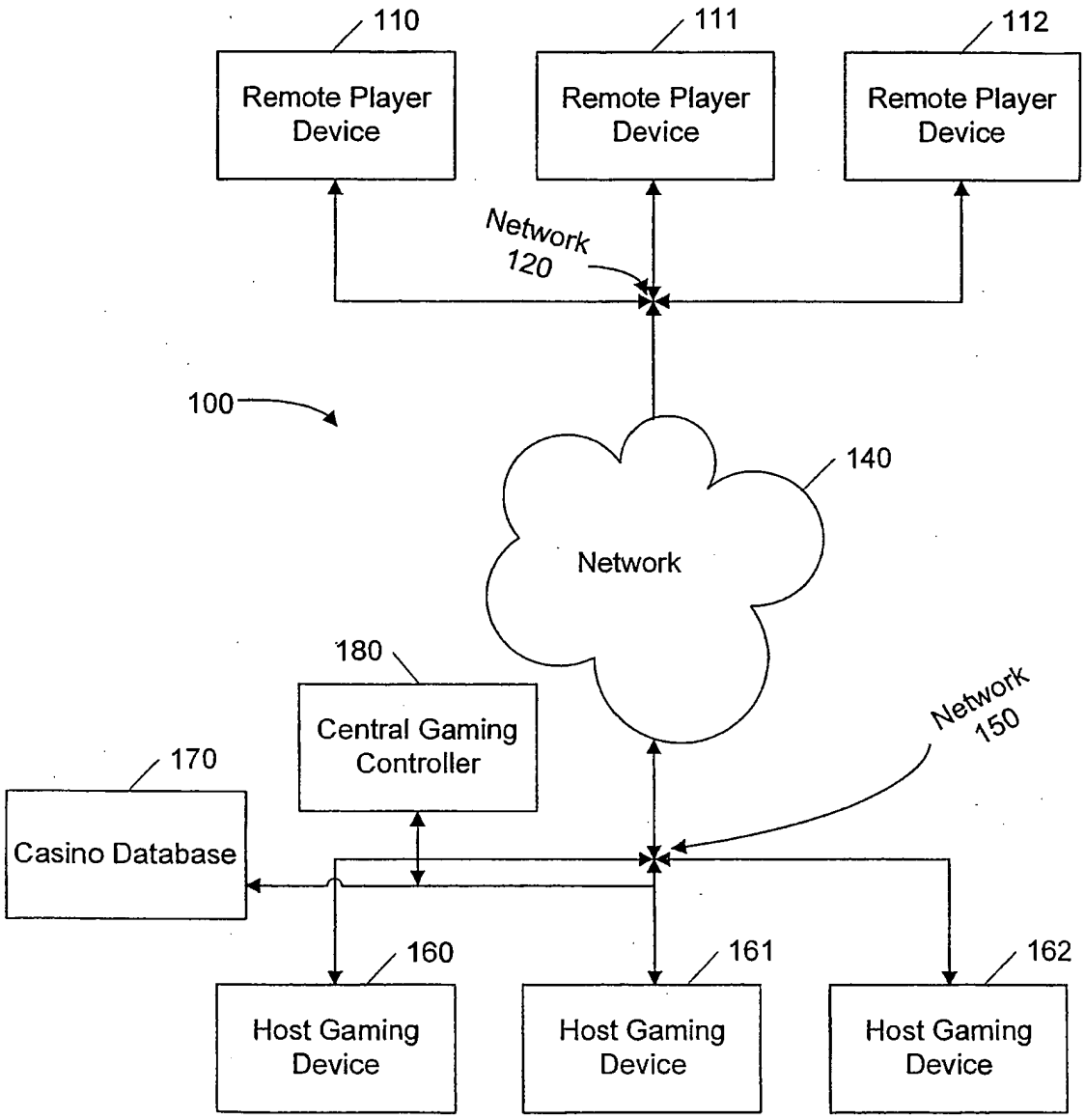

110
111
112
Remote Player Device
Remote Player Device
Remote Player Device
Network 120
100
140
Network
180
Central Gaming Controller
Network 150
170
Casino Database
160
161
162
Host Gaming Device
Host Gaming Device
Host Gaming Device

FIG. 1

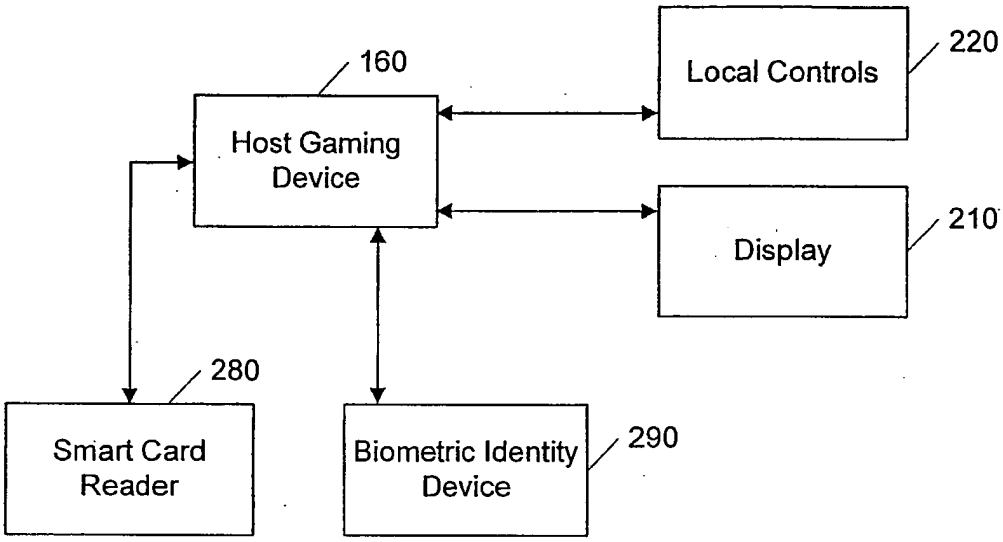

160
Host Gaming Device
Local Controls
220
Display
210
280
Smart Card Reader
Biometric Identity Device
290

FIG. 2

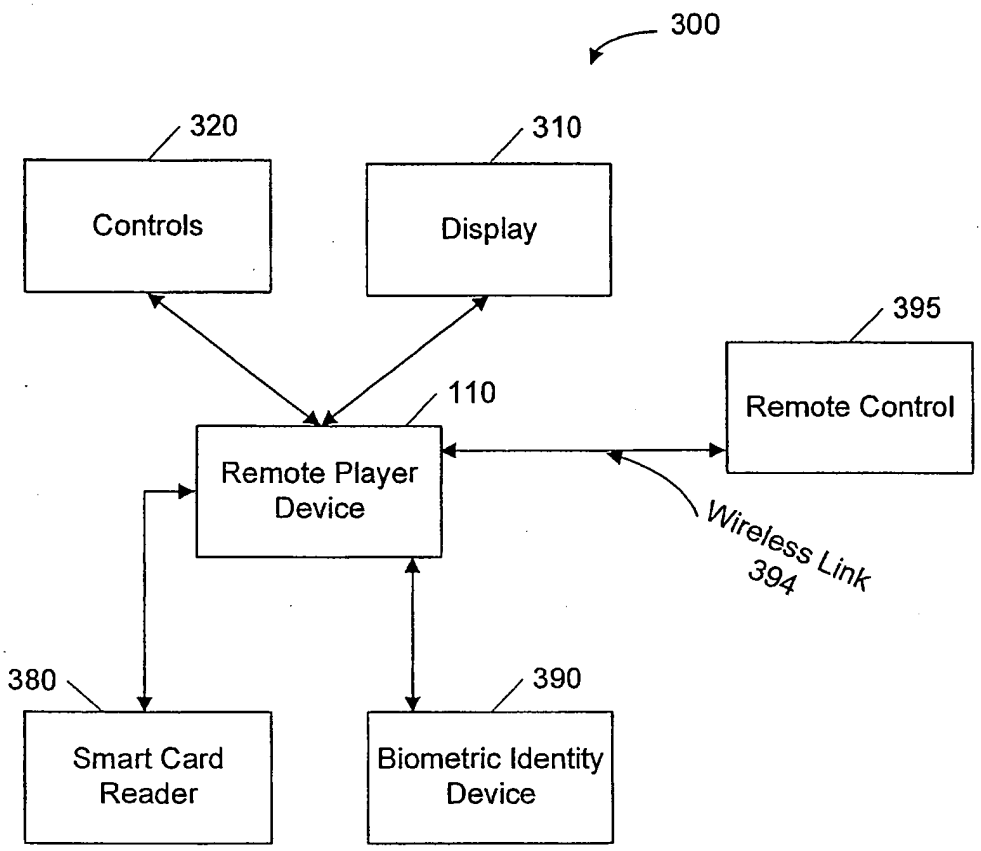

300
320
Controls
310
Display
395
Remote Control
110
Remote Player Device
Wireless Link 394
380
Smart Card Reader
390
Biometric Identity Device

FIG. 3

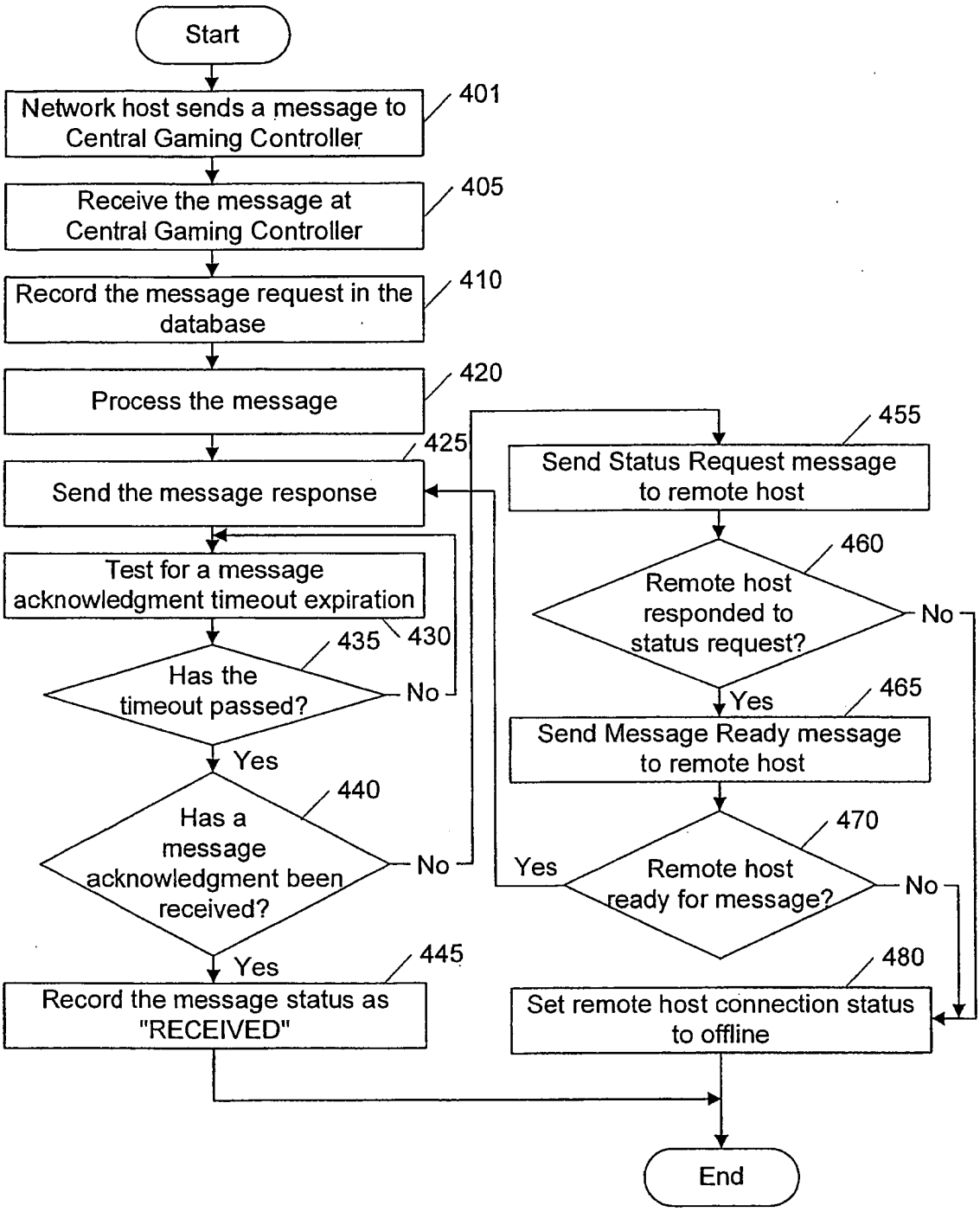

Start
Network host sends a message to Central Gaming Controller
401
Receive the message at Central Gaming Controller
405
Record the message request in the database
410
Process the message
420
Send the message response
425
Test for a message acknowledgment timeout expiration
430
Has the timeout passed?
435
No
Yes
Has a message acknowledgment been received?
440
No
Yes
Record the message status as "RECEIVED"
445
Send Status Request message to remote host
455
Remote host responded to status request?
460
No
Yes
Send Message Ready message to remote host
465
Remote host ready for message?
470
Yes
No
Set remote host connection status to offline
480
End

FIG. 4

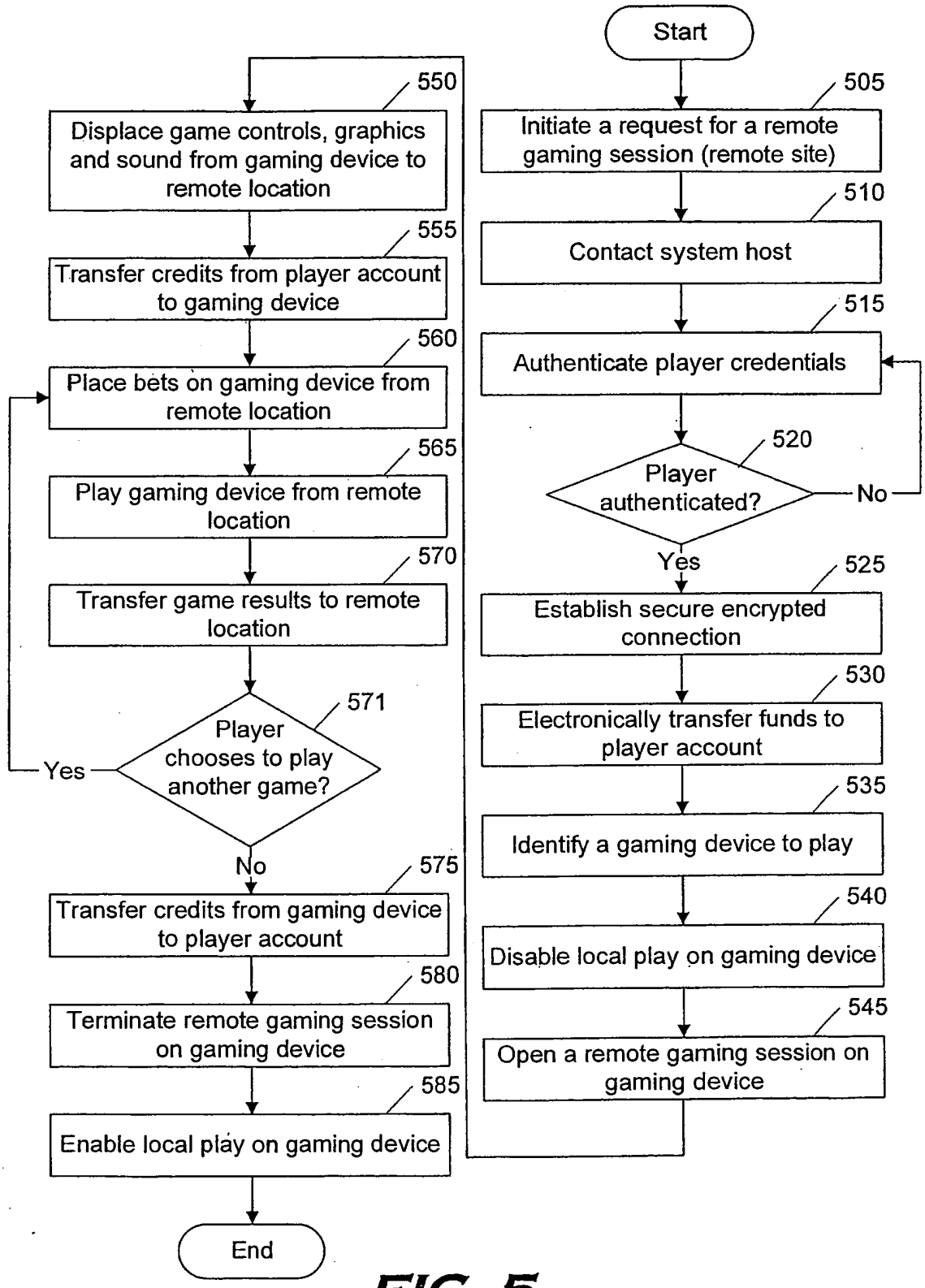

Start
505
Initiate a request for a remote gaming session (remote site)
510
Contact system host
515
Authenticate player credentials
520
Player authenticated?
No
Yes
525
Establish secure encrypted connection
530
Electronically transfer funds to player account
535
Identify a gaming device to play
540
Disable local play on gaming device
545
Open a remote gaming session on gaming device
550
Displace game controls, graphics and sound from gaming device to remote location
555
Transfer credits from player account to gaming device
560
Place bets on gaming device from remote location
565
Play gaming device from remote location
570
Transfer game results to remote location
571
Player chooses to play another game?
Yes
No
575
Transfer credits from gaming device to player account
580
Terminate remote gaming session on gaming device
585
Enable local play on gaming device
End

FIG. 5

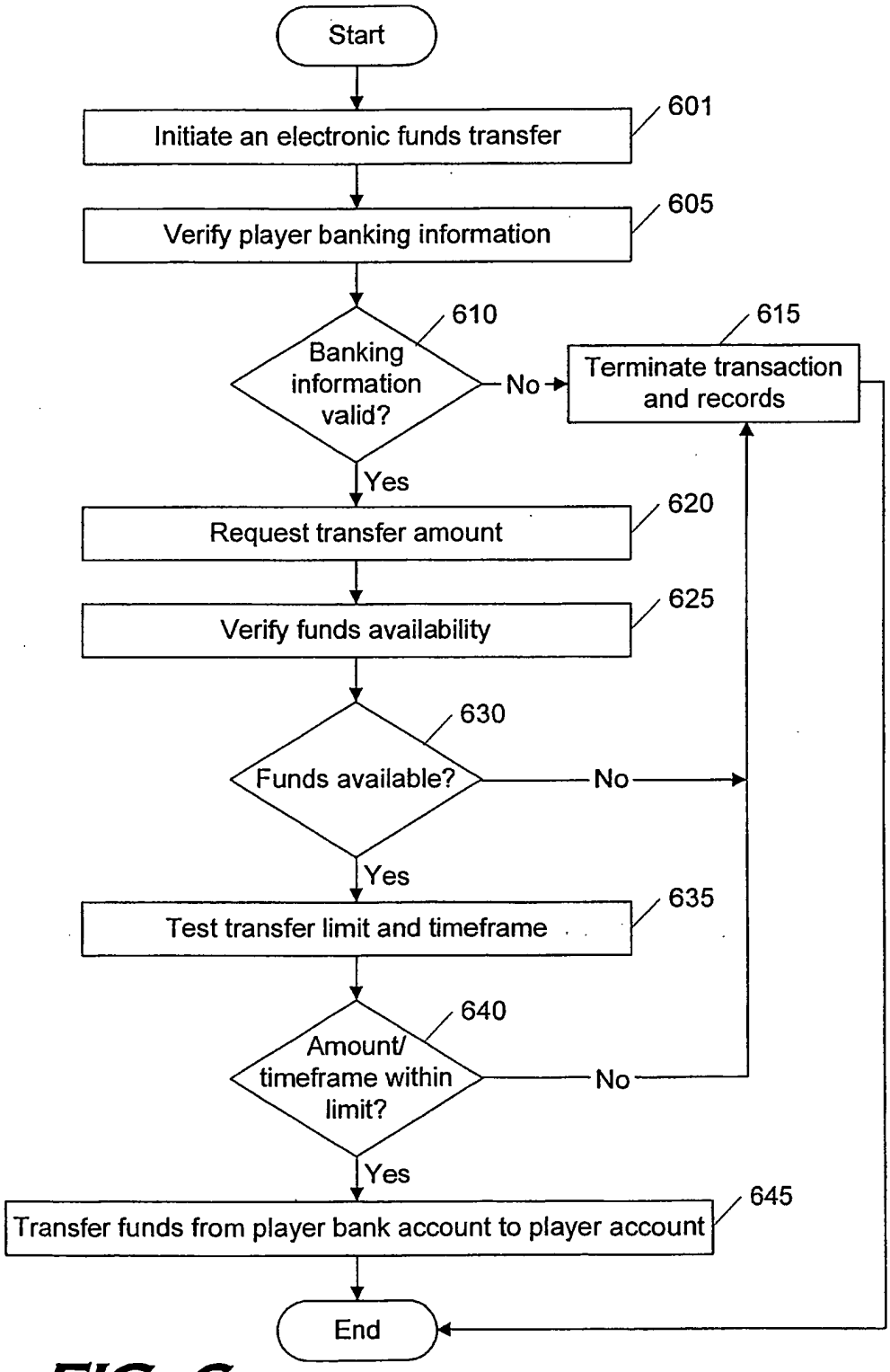

Start
Initiate an electronic funds transfer
601
Verify player banking information
605
Banking information valid?
610
No
Terminate transaction and records
615
Yes
Request transfer amount
620
Verify funds availability
625
Funds available?
630
No
Yes
Test transfer limit and timeframe
635
Amount/ timeframe within limit?
640
No
Yes
Transfer funds from player bank account to player account
645
End

FIG. 6

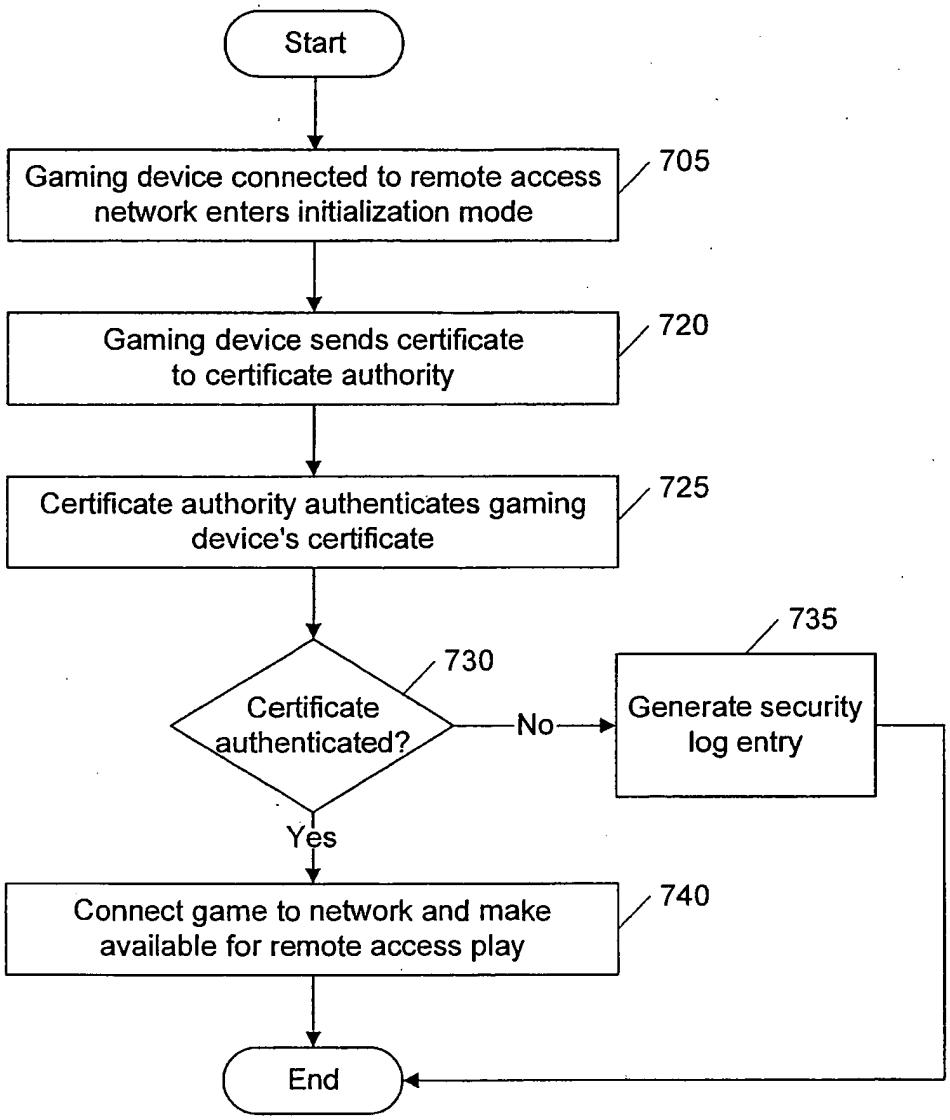

Start
Gaming device connected to remote access network enters initialization mode
705
Gaming device sends certificate to certificate authority
720
Certificate authority authenticates gaming device's certificate
725
Certificate authenticated?
730
No
Generate security log entry
735
Yes
Connect game to network and make available for remote access play
740
End

FIG. 7

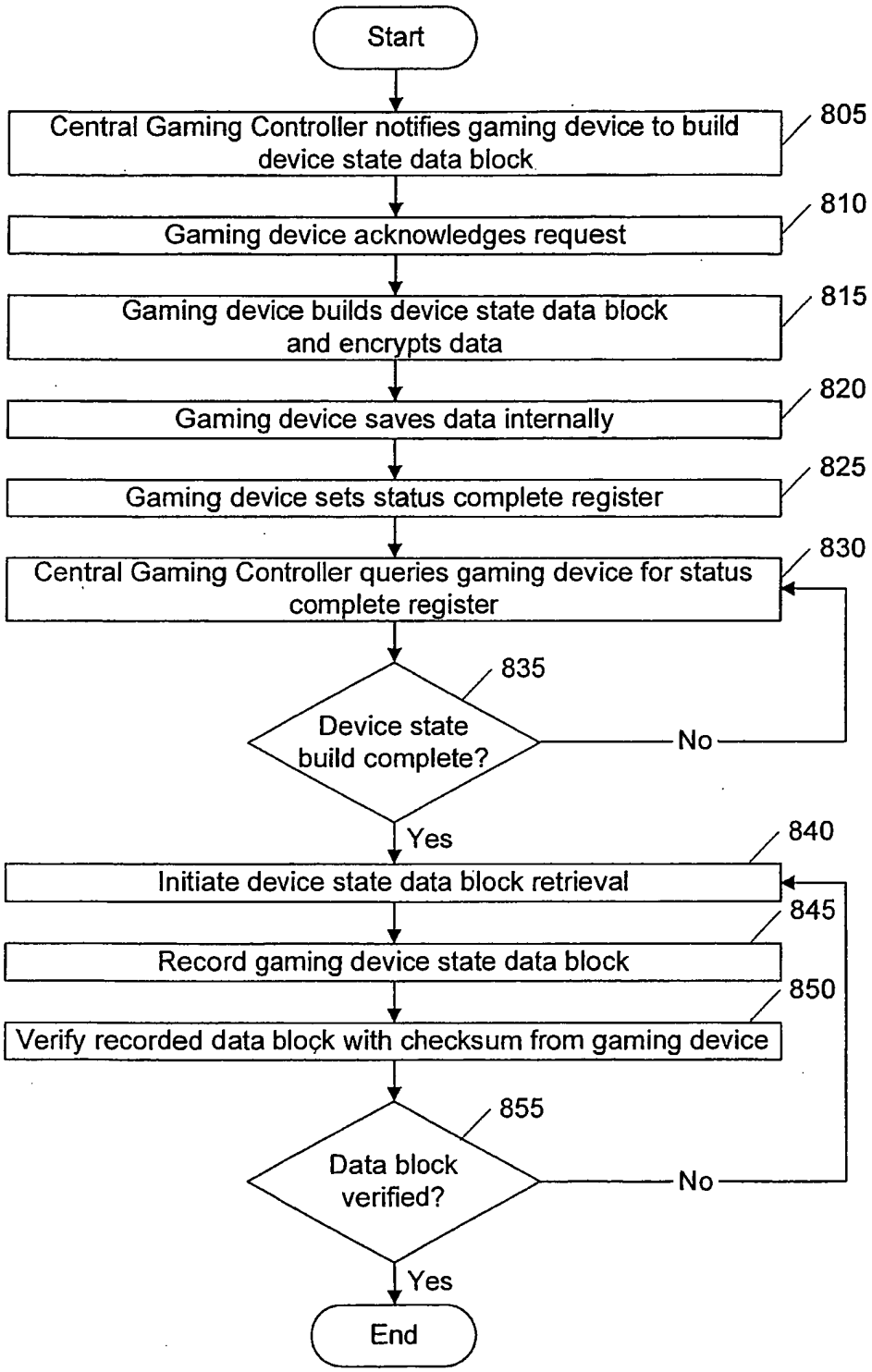

Start
Central Gaming Controller notifies gaming device to build device state data block
805
Gaming device acknowledges request
810
Gaming device builds device state data block and encrypts data
815
Gaming device saves data internally
820
Gaming device sets status complete register
825
Central Gaming Controller queries gaming device for status complete register
830
Device state build complete?
835
No
Yes
Initiate device state data block retrieval
840
Record gaming device state data block
845
Verify recorded data block with checksum from gaming device
850
Data block verified?
855
No
Yes
End

FIG. 8

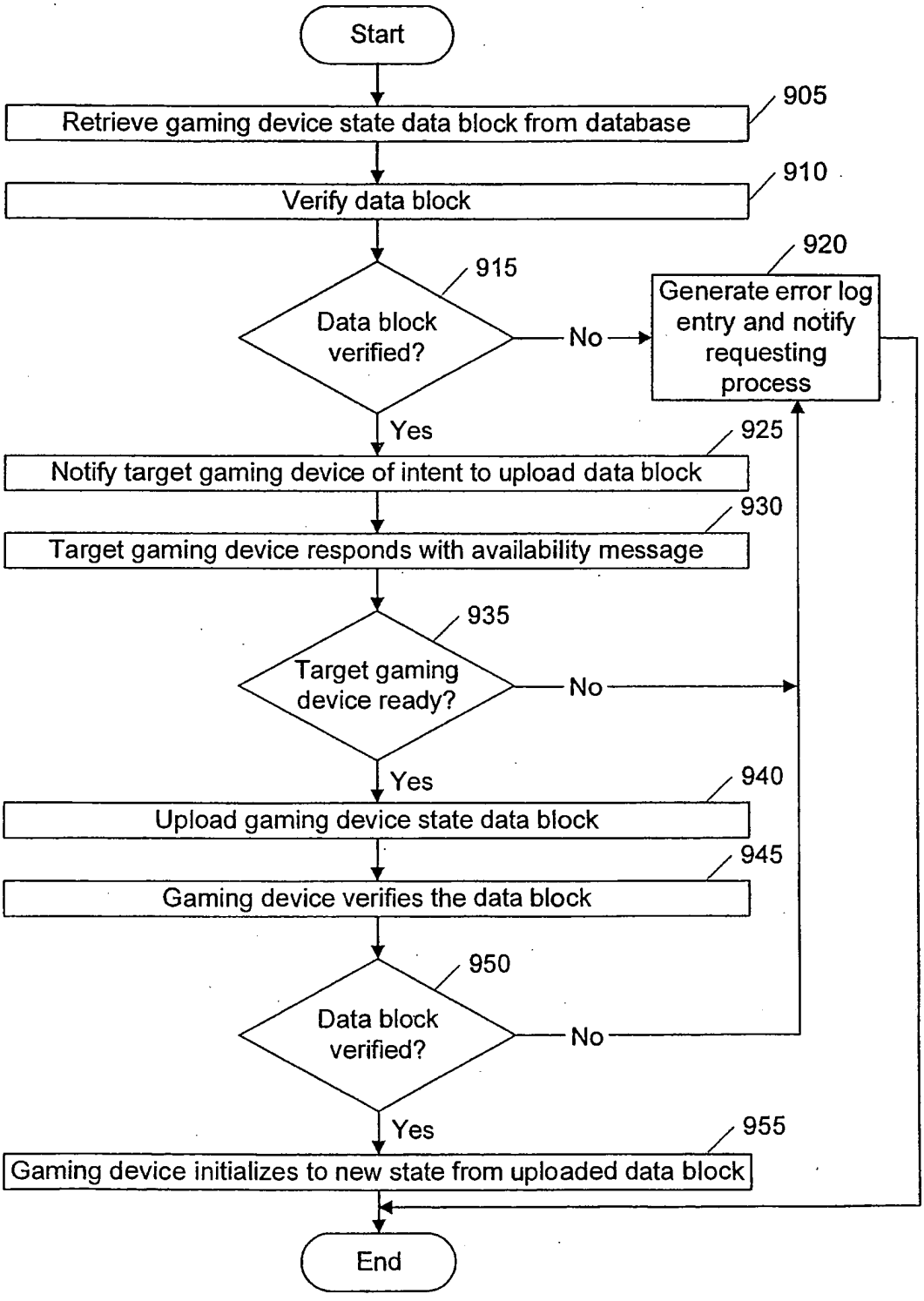

Start
Retrieve gaming device state data block from database
905
Verify data block
910
Data block verified?
915
No
Generate error log entry and notify requesting process
920
Yes
Notify target gaming device of intent to upload data block
925
Target gaming device responds with availability message
930
Target gaming device ready?
935
No
Yes
Upload gaming device state data block
940
Gaming device verifies the data block
945
Data block verified?
950
No
Yes
Gaming device initializes to new state from uploaded data block
955
End

FIG. 9

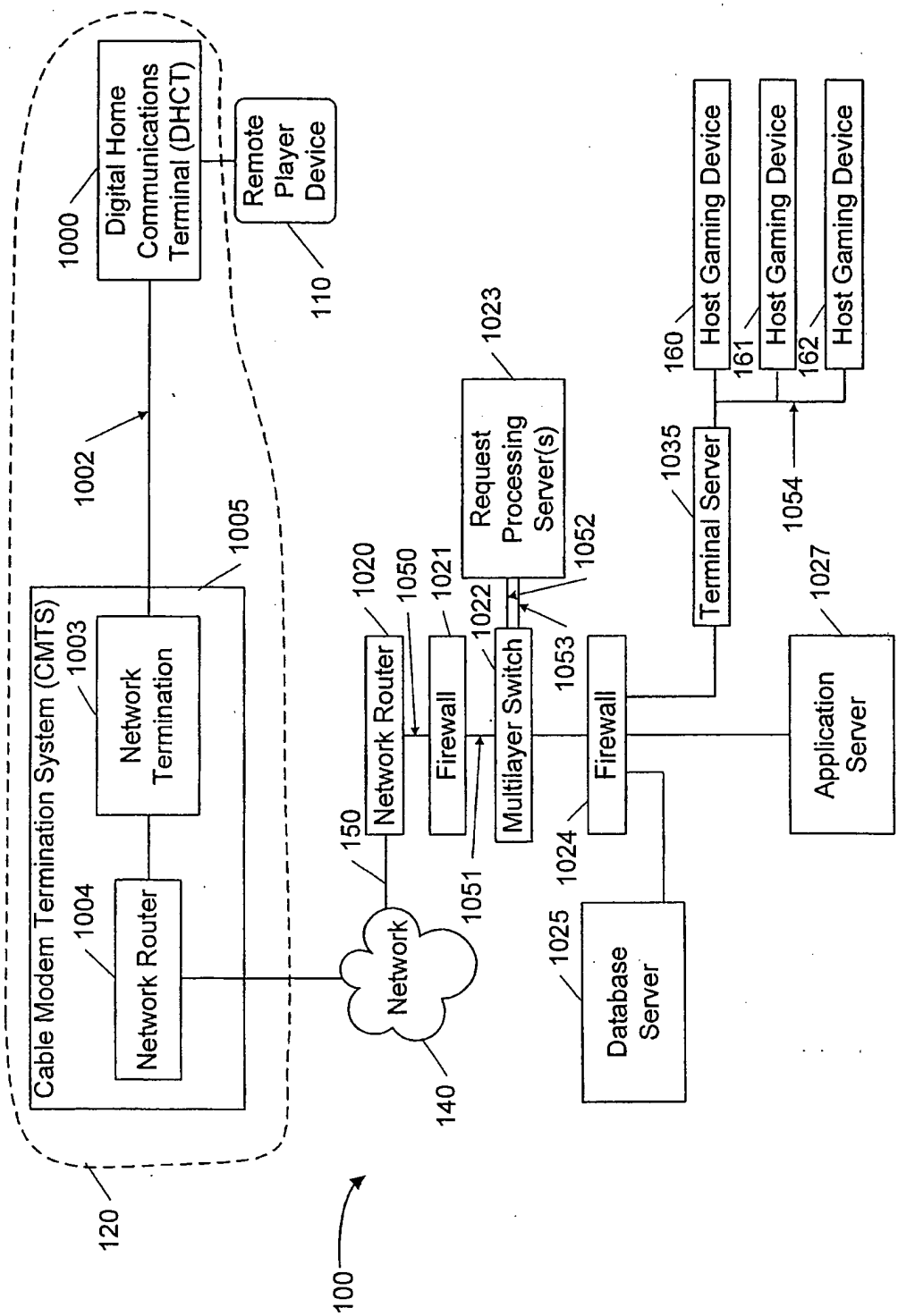

Digital Home Communications Terminal (DHCT)
1000
Remote Player Device
110
1002
1005
Cable Modem Termination System (CMTS)
1003
Network Termination
1004
Network Router
120
100
150
Network Router
1020
1050
1021
Firewall
1022
Multilayer Switch
1051
140
Network
Request Processing Server(s)
1023
1052
1053
Firewall
1024
Database Server
1025
Terminal Server
1035
1054
Application Server
1027
Host Gaming Device
160
Host Gaming Device
161
Host Gaming Device
162

FIG. 10

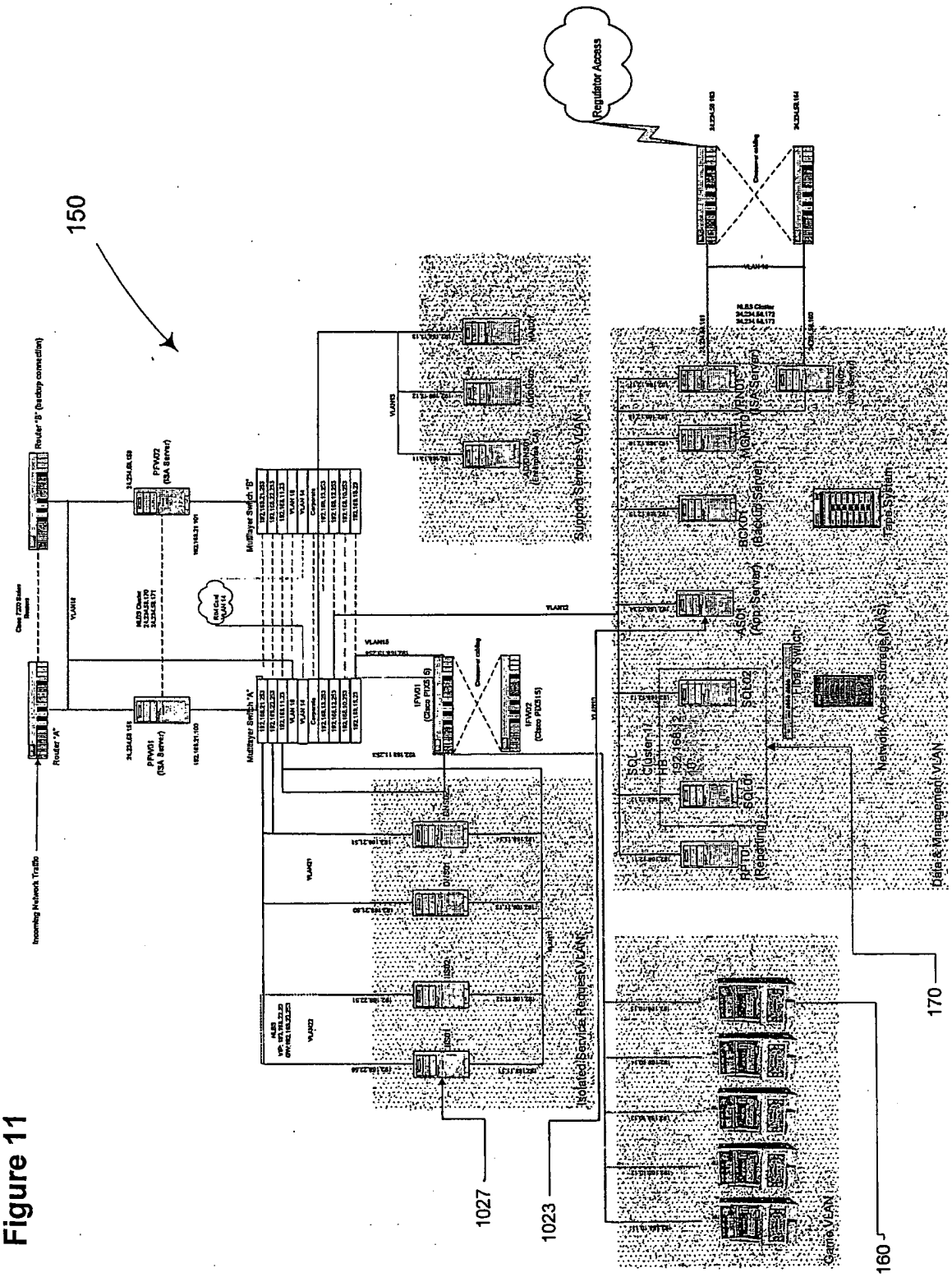

150
Regulator Access
Support Services VLAN
Isolated Service Request VLAN
Data & Management VLAN
Servers VLAN
Network Access Storage (NAS)
Tape System
1027
1023
160
170

Figure 11

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 January 2001 (11.01.2001)

PCT

(10) International Publication Number
WO 01/03044 A1

(51) International Patent Classification[7]: G06F 17/60

(21) International Application Number: PCT/US00/18510

(22) International Filing Date: 6 July 2000 (06.07.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/347,391 6 July 1999 (06.07.1999) US

(71) Applicant *(for all designated States except US)*: TRANSCAST INTERNATIONAL, INC. [US/US]; Regency Plaza, 2350 Mission College Blvd., Suite 190, Santa Clara, CA 95054 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: NARAYAN, Kris [US/US]; 983 Sandalridge Court, Milpitas, CA 95035 (US).

(74) Agent: THAPPETA, Narendra, Reddy; Law Firm of Naren Thappeta, 39899 Balentine Drive #119, Newark, CA 94560 (US).

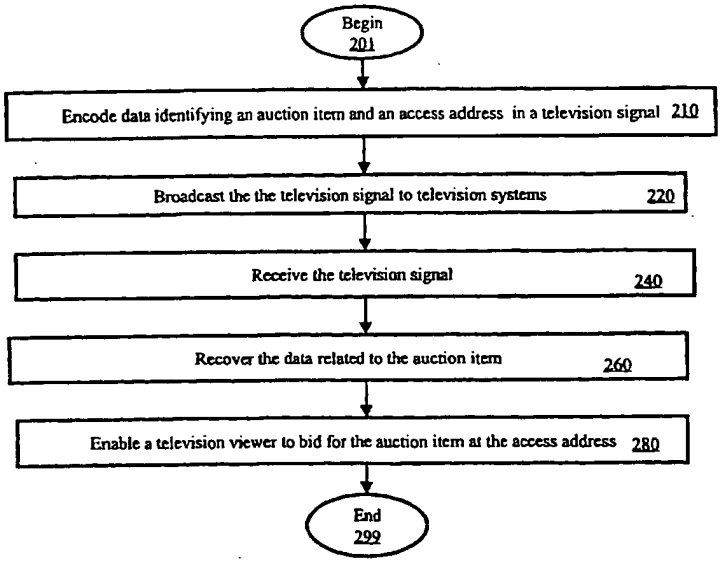
(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: ENABLING VIEWERS OF TELEVISION SYSTEMS TO PARTICIPATE IN AUCTIONS



(57) Abstract: Enabling the viewers of television systems to participate in auctions. Data identifying an item (e.g., description of the auction item and a unique code) offered for sale in an auction and an access address (e.g., universal resource locator of a web site) may be encoded (210) in a television signal and broadcast (220) to various television systems. The data may be recovered (240, 260) by a transaction enabler which enables a viewer to bid for the auction item (280). Other information such as highest present bid price may also be encoded in the television signal and displayed for the viewer.

WO 01/03044 A1

ENABLING VIEWERS OF TELEVISION SYSTEMS TO PARTICIPATE IN AUCTIONS

Related Application

The present invention is related to co-pending U.S. Patent Application Entitled, "Encoding Hot Spots in Television Signals", Serial Number: 09/276,266, Filing Date: March 25, 1999, which is incorporated in its entirety into the present application.

Background of the Invention

Field of the Invention

The present invention relates to television systems, and more specifically to a method and apparatus for using television signals to enable viewers of television systems to participate in auctions.

Related Art

An auction generally refers to a process in which multiple parties are provided the opportunity to bid for an offered item. The offered item can be a process or a service. In a typical bidding process, an a seller offers an item, and a party ("bidder") bids for the offered item usually by specifying a price the party is willing to pay. The seller may specify the minimum acceptable price and a time at which the auction closes.

Typically, an offered item is sold to the highest bidder (i.e., party specifying highest price) in return for the specified highest price. However, criteria other than price (e.g., credit worthiness) of the bidder may also be taken into consideration in determining the bidder to whom to sell an offered item.

Central servers are known in the relevant arts which coordinate the bidding process. For example, web site at URL of http://www.ebay.com enable sellers to offer products

according to various categories (e.g., sports memorabilia, computers), and a bidder may bid on the offered products by using a browser on the world-wide web as is well known in the relevant arts.

Organizations such as those providing the web sites to enable auctions are hereafter referred to as "service providers". Service providers often advertize on various other web sites so that users accessing ("surfing") these web sites may know about the general service. Typically, a user (viewer of the advertisement) can click on an advertisement to access the web sites providing the auction service.

However, these advertisements are typically targeted to the users surfing the world wide web, and may not target at least some of the viewers ("television viewers") of television systems. The television viewers constitutes a big segment of the auction market, and it is therefore desirable to enable television viewers to participate in the auctions.

Such participation may be particularly important as the viewers of a specific television program may be expected to be of certain 'profile', and certain items may be suitable for people of that profile. For example, a person watching Mr. Mark McGuire (a baseball player in United States baseball) hit a record breaking home run may be interested in purchasing a baseball bat signed personally by Mr. Mark McGuire. That is, the auction items can be targeted to the viewers of television programs.

At least for the above-stated reasons, what is needed is a method and apparatus for enabling viewers of television systems to participate in auctions.

Summary of the Invention

The present invention enables viewers ("television viewers") of television systems to participate in auctions. The auctions may be occurring on web sites on the Internet also. In

an embodiment of the invention, data describing an item ("auction item") available for bidding and an access address of a system at which a television viewer may bid are encoded in a television signal.

The user may submit a bid at a system (e.g., a web site) identified by the access address. In case the system is a web server, users ('surfers') of world-wide-web may also submit bids by accessing the web server on the world-wide web. Accordingly, the present invention may be used to draw television viewers to web-sites (e.g., www.ebay.com) dedicated to auctions also.

In an embodiment, the data is encoded in the non-display portion (e.g., vertical blanking interval) of the television signal. However, other portions of a television signal may also be used for encoding the data. Other information of interest to the viewer such as a minimum bid amount specified by a seller and the present maximum may also be encoded in the television signal, and displayed for viewer convenience.

A transaction enabler may recover the data encoded in the television signals, and display the information to the viewer. The viewer may conveniently bid on the auction items, for example, by specifying the bid price (offer) and clicking on a pre-specified portion of a displayed image.

The bid may be automatically sent to a server identified by the access address. In the alternative, the viewer may be first navigated to a web server specified by the access address, and the user may specify the bid price then. A unique code identifying the auction item may also be encoded in the television signal, and the code may be used to identify that the bid price relates to the auction item. In the alternative, the URL itself may contain such identification codes.

The transaction server may also provide updated information on a present highest bid. For example, an end time associated with the auction may be provided to the television viewer, and the viewer may check the present highest bid at a later time before the end time, and then decide whether to submit a bid. In addition, the transaction server may interact with the system providing the auction service, and provide periodic updates at viewer's option. As a result, a viewer may make an informed decision on whether to bid.

Therefore, the present invention enables a television viewer to participate in an auction by encoding in a television signal the data identifying an auction item and an access address.

The present invention enables television viewers to be drawn to web sites providing auction service by specifying the URL of the web site as the access address.

The present invention is useful for broadcasters as the broadcasters may facilitate the joining of additional bidders to a bidding process, and be compensated for such additions.

The present invention is useful for service providers providing auction service as the television viewers are drawn to bid for on-going auctions.

The present invention is useful for service providers providing auction service also because higher commissions may be charged for the auction items sold in accordance with the present invention.

The present invention is useful for television viewers as a television viewer may have non-intrusive access to information on auctions, and purchase the auction items by a convenient user interface.

The present invention is useful for sellers participating in auctions as the sellers may attain greater return for the auction items due to additional pool of bidders participating in accordance with the present invention.

Further features and advantages of the invention, as well as the structure and operation of various embodiments of the invention, are described in detail below with reference to the accompanying drawings. In the drawings, like reference numbers generally indicate identical, functionally similar, and/or structurally similar elements. The drawing in which an element first appears is indicated by the leftmost digit(s) in the corresponding reference number.

Brief Description of the Drawings

The present invention will be described with reference to the accompanying drawings, wherein:.

Figure 1 is a block diagram illustrating an example environment in which the present invention can be implemented;

Figure 2 is a flow-chart illustrating a method in accordance with the present invention;

Figure 3 is a block diagram illustrating an example broadcast system which encodes data related to an auction item in a television signal;

Figure 4 is a block diagram illustrating the details of a transaction enabler in an embodiment of the present invention;

Figure 5 depicts a display screen using which a user may participate in auctions in accordance with the present invention.

Detailed Description of the Preferred Embodiments

1. Overview and Discussion of the Invention

The present invention allows viewers ("television viewers") of television systems to participate in auctions. Typically, the data relating to an item ("auction item") offered for sale in an on-going auction is encoded in a television signal. The encoded information may be displayed while the television viewers watch the images encoded in the television signal. The

viewers may be provided a convenient interface to bid on the auction item.

Auction items consistent with expected viewer profiles may be sold using the present invention. A seller may be able to sell at higher prices as many viewers are likely to bid. For example, a diamond ring may be auctioned towards the end of a romantic movie. The invention is described below with respect to several examples for illustration.

2. Example Environment

Figure 1 is a block diagram illustrating an example environment in which the present invention can be implemented. The environment may include bidding systems 110-A and 110-B, Internet 120, web site 130, broadcast system 150, and television 170. A viewer of television 170 may participate in auctions as described below in further detail.

Web site 130 may provide an auction service. As an illustration, web site 130 may implement the interface of www.ebay.com, well known in the relevant arts. Bidder systems 110-A and 110-B may access Internet 120 to bid on the items offered for sale on web site 130. Bidding systems 110-A and 110-B, Internet 120, and web site 130 may be implemented in a well-known way. Even though the auction service is shown as being provided from web site 130, it should be understood that different other servers using different access technologies (e.g., dial-up) may be used in providing the service.

Broadcast system 150 includes information related to an auction item in a television signal and transmits the television signal on broadcast medium 146 (airwaves, cable, etc.). The data may specify the item offered for sale, the present highest bid, and an access address for enabling the viewer to bid. For example, the access may contain a URL of web site 130. An example embodiment of broadcast system 150 is described below.

The auction may be in progress (on-going) on web site 130, and accordingly broadcast system 150 may access web site 130 to access any data (e.g., present highest bid) for inclusion

in the television signal. Link 134 may be provided on Internet 120 even though a dedicated line is shown in Figure 1.

Viewer bidding system 150 receives the television signal, and enables a viewer to participate in auctions. Viewer bidding system 150 may display the images encoded in the received television signal. In addition, viewer bidding system 150 may recover the data related to the auction item, and display the corresponding information. By appropriate action, the user may indicate a higher bid and transmit the higher bid on virtual link 163 on Internet 120.

In an embodiment, viewer bidding system 150 may include television 170, transaction enabler 160, and remote control 180. Transaction enabler 160 may overlay any images necessary for providing an user interface on top of the images encoded in the television signal ("television signal images"). For example, information identifying the auction item (e.g. Mark McGuire's bat) and the highest bid price may be overlaid on television signal images.

Transaction enabler 160 may encode the overlaid image in a form consistent with conventional television signals for display on television 170. In other words, transaction enabler 160 operates as a 'set-top' box. However, transaction enabler 160 may be integrated into television 170, for example, using embedded chip-sets provided by TeleCruz Technology, Inc. In either case, remote control 180 enables the user to specify the bid price and to transmit the new bid. An example embodiment of transaction enabler 160 is described below in further detail. However, first a method in accordance with the present invention is described first below.

3. Method

Figure 2 is a flow-chart illustrating a method in accordance with the present invention. The method begins in step 201, in which control passes to step 210. In step 210, data identifying an auction item and an access address may be encoded in a television signal. The data identifying an auction item may include both a descriptive component (e.g., "baseball bat signed by Mark McGuire") and a unique code specifying the auction item (or group in case multiple items of the same type are available).

The data may be encoded in one of different formats depending on different criteria, but consistent with an interface at viewer bidding system 150. For example, a unique code identifying an auction item may be encoded as a parameter of a URL (access address) since the web browser's based technology lends well to such encoding and later submission of a bid. The television signal may also be encoded with image frames for display on television signals. Both (images and data related to auction items) encoding may be performed in a known way.

In step 220, the television signal may be broadcasted to television systems covering a large geographic area. In step 240, the television signal may be received at a viewer end (e.g., by transaction enabler 160 of Figure 1). In step 260, the data related to the auction item (encoded in step 210) may be recovered. The recovery generally needs to be consistent with the encoding scheme used by broadcast system. In general, any compatible encoding scheme may be used.

In step 280, the user is provided a convenient user interface to bid on the auction item. Typically, the description of the auction item is displayed, and the user may be provided the option to bid, in which case the bid is submitted to a system identified by the access address. While submitting the bid, the unique code identifying the auction item may be used to specify to the system that the bid relates to that particular item. The access address is used to connect

the user to a central machine (e.g., web site or any server) or person. The user may then submit the bid. The highest bidder is generally entitled to the offered auction item for the submitted bid.

The method and environment described above may be applied in several ways as will be apparent to one skilled in the relevant arts based on the disclosure herein. All such implementations are contemplated to be within the scope and spirit of the present invention. However, it may be desirable to have bidders (television viewers) participation at different points of a broadcast. The manner in which the point can be controlled is described below with respect to broadcast system 140.

4. Broadcast System

Figure 3 is a block diagram illustrating an example embodiment of broadcast system 140. Even though the description of broadcast system is provided substantially with respect to broadcasters producing a television signal, the present invention can be practiced by intermediate broadcasters also. Such advertisements are generally more targeted to the specific geographic profile. Broadcast system 140 may contain production block 310, authoring block 320, broadcast block 330, timing determination block 340, auction data interface 360, and storage 350. Each block is described in further detail below.

Timing determination block 340 may determine the specific time at which to encode data related to an auction item. For example, it may be desirable to broadcast data related to a baseball bat (auction item) when a home run is hit. Timing determination block 340 may be implemented to monitor the scores of the baseball game and generate an indication to auction data interface 360. Several other criteria can be used in determining when to send data related to an auction item.

Timing determination block 340 may also determine when to send updates

corresponding to various auctions. When timing determination block 340 determines to cause update corresponding to an auction to be sent, auction data interface 360 may interact with web site 130 to retrieve a present highest bid from web site 130. The present highest bid may be provided to authoring block 320 for encoding in a broadcast television signal.

Auction data interface 360 receives data on line 134 if a web based auction is on-going for the auction item of interest on web site 130. The data may indicate the present highest bid, bid history, the seller, any comments about the seller. As noted above, auction data interface 360 may provide the data to be encoded in the television signals. The data may contain, in addition to the data retrieved from web site 130, data identifying the auction item (descriptive component and unique code).

Some of the data may be pre-stored in storage 350 also. For example, it may be desirable to display graphic icons on television systems to represent different auction items. Bit maps representing the graphics icons may be stored in storage 350. In general, auction data interface 360 may gather any data which may be of interest to bidders, and pass the data to authoring block 320.

Production block 310 may contain different components such as cameras which are used to film a show/program. The display signal is preferably in a form suitable for eventual transmission as a television signal. In general, production block 310, may encode images in a display data portion of a television signal. The images may be displayed later on a television system for viewing a broadcast program. Production block 310 may be implemented in a known way.

Authoring block 320 encodes data received from auction data interface 360 into television signals. The data may be encoded according to any convention, and transaction enabler 160 may need to be accordingly designed. Several such conventions can be designed

in known way. Authoring block may either store the resulting signal in storage 350 or forward to broadcasting block 330.

In one embodiment, authoring block 320 encodes the data in non-display portion (e.g., vertical blanking interval) of the display signal. Such encoding may be performed in a known way. In an alternative embodiment, the data may be encoded in other portions (e..g, least significant bits of pixel data elements representing an image) as well. This alternative embodiment is described in further detail in co-pending U.S. Patent Application Entitled, "Encoding Hot Spots in Television Signals", Serial Number: 09/276,266, Filing Date: March 25, 1999, which is incorporated in its entirety into the present application.

Even though the encoding is described with reference to analog television signals, it should be understood that the present invention may be practiced in conjunction with digital television signals (e.g., those suitable for HDTV) also. Some of the techniques described in this application may be employed for such encoding in the digital television signals. Many other techniques will be apparent to one skilled in the relevant arts based on the disclosure herein. Such other techniques are also contemplated to be within the scope and spirit of the present invention.

Broadcast block 330 may broadcast television signals (containing the hot spot data in the display data portion) in a known way. It should be noted that the television signal can be in progressive scan format or interlaced format. Production block 310 and authoring block 320 need to be implemented taking into consideration the transmission standard (progressive vs. interlaced, and digital vs. analog) of the television signals. Thus, broadcast block 330 generates television signals containing data which may be used to enable television viewers to bid on the auction items.

Transaction enabler 160 receives the television signals and enables a viewer to bid on

the auction items. Example embodiments of transaction enabler 160 are described below in further detail. Before describing example embodiments of transaction enabler 160 in detail, it is helpful to understand some typical problems with the user interface.

5. Problems and Solutions

In one embodiment, a highest present bid may be encoded in the television signal, and the user may submit a higher bid that the highest present bid. One problem associated in the environments of Figures 1 and 2 is that many bidders may bid for the auction item based on the same highest bid. As the bids are generally marginally more than the present highest bid, the approach may not maximize the return for the seller.

Accordingly, an improvement may be implemented in which an "auction close time" (time at which the auction for the auction item ends) may be associated with the auction item. The auction close time may also be encoded and transmitted in the television signals. Thus, viewers may choose a later convenient time for bidding on the auction item. However, in such a situation, viewer bidding system 150 may need to store the required data.

Yet another problem is, a viewer may wish to know an updated highest bidding price before actually submitting a bid. Thus, the viewer may be provided a convenient user interface to request a 'present highest bid' associated with an auction item of interest. The updated price may also be received on virtual link 163. In this case also, viewer bidding system 150 may need to store the required data.

In yet another scenario, a viewer may wish continuous updates of the highest bidding price. Accordingly, a viewer may be provided an option of initiating a small window in which the updates to the highest bids are provided continuously (e.g., when highest bid changes or every 3 seconds). An embodiment of transaction enabler 160, which provides for at least these features is described below.

6. Transaction Enabler

Figure 4 is a block diagram illustrating the internals of an example embodiment of transaction enabler 160 containing image decoder 410, memory 430, recovery block 420, processor 450, digital to analog converter (DAC) 485, multiplexor 480, infra-red (IR) receiver 460, telephone interface 470 and broadband interface 475. Each component is described below in further detail.

Image decoder 410 generates pixel data elements representing image frames encoded in a television signal received on broadcast channel 146. In response to the operation of remote control unit 180, image decoder 410 may store the pixel data elements representing an image frame in memory 430. Such storage enables overlays. Image decoder 410 may be implemented in a known way. Memory 430 may represent several memory modules such as fast random access memories and relatively slower non-volatile memories. The non-volatile memories may store data and program instructions which enable the operation of the present invention.

Recovery block 420 recovers the data related to auction items encoded in the received television signal. In general, recovery block 420 needs to be implemented consistent with any conventions or protocols used at broadcaster end 380 for encoding the hot spot data. If the data is encoded in non-display portions (e.g., VBI), the data may be recovered in a known way.If the data is encoded in display data portion (i.e., in images), recovery block 420 may examine the pixel data elements stored in memory 430 to recover the data. Further details of recovery are noted in co-pending U.S. Patent Application Entitled, "Encoding Hot Spots in Television Signals", Serial Number: 09/276,266, Filing Date: March 25, 1999, which is incorporated in its entirety into the present application.

Infra-red (IR) receiver 460 receives remote control signals from remote control unit 180, and provides digital data representing the remote control signals to processor 450. The control signals may indicate whether the user wishes to see auction item related data, to enter the bid, to receive an updated present highest bid, etc. Several features of the user interface may be activated by a viewer using IR receiver 460. IR receiver 460 may be implemented in a known way. It may be noted that other receivers which receive control signals from viewers and provide corresponding digital data to processor 450 may be implemented.

Telephone interface 470 enables a telephone call to be initiated. Such telephone calls may be generally initiated either to connect to the Internet via an ISP or to contact a phone with a live-operator. When a telephone call is initiated with a live operation, telephone interface 470 may provide the necessary micro-phone (for a viewer to speak) and receiver for reproducing audible voice. Alternatively, a user may utilize a conventional telephone set that is attached to line 335.

Broadband interface 475 may provide a high speed connection (e.g., using a local area network, digital subscriber loop technology or cable interface) to connect with a web server (corresponding to an URL) or even initiate a voice call (e.g., using voice over Internet Protocol). Telephone interface and broadband interface may be logically viewed as being part of line 163 of Figure 1. In general, broadband interface 475 and telephone interface 470 provide the communication to a system (specified by access address) providing auction service.

Processor 450 receives data related to auction items from recovery block 420, and enables a user to send a bid to a system identified by an access address. The transmission of the bid may be either by broadband interface 475 or telephone interface 470 as specified by the type of access address. Processor 450 may also implement the user interface features noted

in the section above.

For a suitable user-interface, processor 450 may control the images displayed on television system 110. For example, processor 450 may overlay information in the auction items related data on the television signal image. Specifically, the portion to be overlaid on television images may be provided by processor 450, and control line 481 may be controlled to accomplish the overlay function. However, when a user does not wish to bid or when data related to auction items is absent in television signals, processor 450 may control select line 481 to cause the television signal received on line 146 to be passed directly on line 167. In addition, processor 450 may cause auction related data to be displayed in a transparent mode. Typically, techniques such a half-tone control are used for achieving such transparency of display.

If the access address is a URL, transaction enabler 160 may need to operate as a web-browser. Processor 450 may enable such an operation by executing the program instructions provided by memory 430. The web-browser enables transaction enabler 160 to receive different web-pages in a known way. Processor 450 may convert the web pages into image frames, and encode the image frames into a television signal having a format compatible with conventional television signals such that the images can be displayed on television system 110. Well known methods may be employed for such conversion and encoding.

Therefore, transaction enabler 160 may operate in conjunction with broadcast system 140 to enable a television viewer to participate in auctions. As a result, viewers of television systems may be drawn to participate in auctions which are generally accessed mostly by users surfing the world-wide-web. It should be understood that web site 130 and broadcast system 140 may be integrated as one unit depending on the available technologies, and in such a case,

transaction enabler 160 may communicate with such a unit directly. The present invention is described in further detail below with reference to an example user interface considering some of the description of above.

7. User Interface

Figure 5 is a diagram illustrating the manner in which transaction enabler 160 may enable a viewer of television programs to participate in auctions. It should be understood that transaction enabler may use other display devices from which a user can participate in auctions. In addition, other types of systems (such as computers) which display images in television signals may also be used to participate in auctions in accordance with the present invention.

Continuing the description with reference to Figure 5, there is shown television display 500 (for example, on television 170). Auction related data may be received in accordance with the present invention, and the relevant data may be displayed in a small window 540. Window 540 is preferably overlaid on television program images as a transparent window using techniques such as half-toning well known in the relevant arts. By using a transparent display, a viewer may be able to watch the programs encoded in the television signal while participating in the auctions.

Window 540 may be used to display the description of the auction item ("McGuire's 70th Home Run Bat" in the example there), the present highest bid, bidder of the highest bid, and the time at which the auction for this item is expected to close may be displayed. The present highest bid may be periodically updated using the data received on the broadcast television signal. On the other hand, a viewer may select (click on) 'Update' text to cause transaction enabler 160 to initiate a dialogue with web server 130, and retrieve updated information for a presently watched auction item. Thus, in Figure 5, such a selection may

cause transaction enabler 160 to display $4300 (representing an increase in the present highest bid).

The user may select 'Bid History' to view the previous bidders and history. The relevant data may either be displayed based on data stored locally or the data may be retrieved from web site 130 in response to a user request. As is well known in the relevant arts, auction sites such as www.ebay.com provide such bid histories.

The user may specify her/his bid price in the box provided next to text 'Your Bid'. The user may then select the 'Submit' text to cause transaction enabler 160 to submit the bid. As noted above, the submission may be according to any mechanism. The bid can potentially be over a broadband interface to access a web site or to a server accepting over a telephone connection. Once the bid is submitted to a server at the access address, the auction item may be sold to a bidder in a known way. If the user of system 150 has the highest bid, the user may pay the bid amount and receive the auction item.

Thus, an interface such as the one above, a user (or television viewers) may bid for auction items in accordance with the present invention. The bid may be submitted according to any pre-specified protocol between transaction enabler 160 and an auction server (e.g., web site 130). The implementation of auction on web site 130 based on such received bid prices will be apparent to one skilled in the relevant arts.

8. Conclusion

While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. Thus, the breadth and scope of the present invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.

What Is Claimed Is:

1. A method of enabling a viewer of a television system to participate in auctions, said method comprising:

(a) encoding in a television signal a data describing an auction item and an access address of a server at which auction service for said auction item is provided; and

(b) transmitting said television signal,

wherein said data can be used to enable said viewer to bid for said auction item at said server.

2. The method of claim 1, wherein said method further comprises:

(c) receiving said television signal encoded with said data in a transaction enabler;

(d) recovering said data encoded in said television signal;

(e) displaying information describing said auction item on said television system;

(f) enabling said viewer to bid at said server specified by said access address.

3. The method of claim 2, further comprising:

(g) enabling said viewer to specify a bid price for said auction item.

4. The method of claim 3, wherein said enabling said viewer to specify said bid price comprises:

(h) enabling said viewer to indicate said bid price; and

(i) transmitting said bid price to said server at said access address.

5. The method of claim 4, wherein said access address comprises a telephone number

of said server, and said method further comprises:

(j) encoding a unique code identifying said auction item;

(k) recovering said unique code in said transaction enabler; and

(l) transmitting said unique code along with said bid price to said server,

whereby said server can easily associate said bid price with said auction item using said unique code.

6. The method of claim 4, wherein said access address comprises a universal resource locator (URL) of a web site, wherein said web site comprises said server, and wherein steps (h) and (i) comprise the further step of enabling said viewer to indicate said price on a web page provided by said web site.

7. The method of claim 1, further comprising:

(m) encoding a present highest bid in said television signal, wherein said present highest bid may be displayed to said viewer before said viewer decides to submit a bid.

8. The method of claim 7, wherein said server comprises a web site, and said method comprising the further step of retrieving said present highest bid from said web site.

9. The method of claim 1, wherein step (a) comprises the step of encoding said data in non-display portion of said television signal.

10. The method of claim 1, wherein step (a) comprises the further step of encoding said data in a non-display portion of said television signal.

11. The method of claim 10, wherein said non-display portion comprises vertical blanking interval (VBI).

12. The method of claim 1, further comprising:

transmitting an updated highest bid price in said television signal, wherein said updated highest bid price corresponds to a present highest bid for said auction item.

13. The method of claim 12, further comprising:

retrieving said updated bid price from said server,

wherein said step of transmitting said updated highest bid price is performed after said step of retrieving said updated bid price from said server.

14. The method of claim 13, further comprising:

enabling said viewer to request a bid history; and

displaying all of said updated bid prices to said viewer.

15. The method of claim 14, wherein said display corresponding to said bid history further comprises a description of the bidder corresponding to each of said present highest bid.

16. The method of claim 1, wherein said data further comprises a time at which auction for said auction item closes.

17. A method of enabling a viewer of a television system to participate in auctions,

said method comprising:

(a) receiving in a transaction enabler a television signal encoded with a data, said data including a description of an auction item and an access address of a server at which auction service for said auction item is provided;

(b) recovering said data encoded in said television signal;

(c) displaying said description of said auction item on said television system;

(d) enabling said viewer to bid at said server specified by said access address.

18. The method of claim 17, further comprising:

(e) enabling said viewer to indicate said bid price; and

(f) transmitting said bid price to said server at said access address.

19. The method of claim 4, wherein said access address comprises a telephone number of said server, and said method further comprises:

(g) encoding a unique code identifying said auction item;

(h) recovering said unique code in said transaction enabler; and

(i) transmitting said unique code along with said bid price to said server,

whereby said server can easily associate said bid price with said auction item using said said unique code.

20. An environment enabling a viewer of a television system to participate in auctions, said environment comprising:

encoding means for encoding in a television signal a data describing an auction item

and an access address of a server at which auction service for said auction item is provided; and

transmission means for transmitting said television signal,

wherein said data can be used to enable said viewer to bid for said auction item at said server.

21. An environment enabling a viewer of a television system to participate in auctions, said environment comprising:

receiving means for receiving a television signal encoded with a data, said data including a description of an auction item and an access address of a server at which auction service for said auction item is provided;

recovery means for recovering said data encoded in said television signal;

displaying means for displaying said description of said auction item on said television system;

enabling means for enabling said viewer to bid at said server specified by said access address.

22. An environment enabling a viewer of a television system to participate in auctions, said environment comprising:

a broadcast system to encode in a television signal a data describing an auction item and an access address of a server at which auction service for said auction item is provided, said broadcast system being designed also to transmit said television signal,

wherein said data can be used to enable said viewer to bid for said auction item at said server.

23. The environment of claim 22, wherein said broadcast system comprises:

a production block to generate images to encode in a display data portion of said television signal;

an authoring block to encode said data in said television signal; and

a broadcast block to transmit said television signal containing said images and said data.

24. The environment of claim 23, further comprising an auction data interface to receive a present highest bid from a server, said auction data interface to provide said present highest bid to said authoring block, wherein said authoring block encodes said present highest bid in said television signal.

25. The environment of claim 24, further comprising a timing determination block to determine the time at which said authoring block encodes said data including said present highest bid in said television signal.

26. The environment of claim 22, further comprising:

a viewer bidding system to receive said television signal, and enabling said viewer to submit a bid and participate in said auction.

27. The environment of claim 26, wherein said viewer bidding system comprises:

a television system;

a remote control which enables said viewer to submit said bid; and

a transaction enabler coupled to said television system and to receive said commands from said remote control, said transaction enabler to recover said data encoded in said television signal and display information contained in said data on said television,

wherein said viewer can submit said bid using said remote control.

28. The environment of claim 27, wherein said transaction enabler is integrated within said television system.

29. The environment of claim 27, wherein said transaction enabler is provided external to said television system, and wherein said transaction enabler overlays a window with information contained in said data on images encoded in the display data of said television signal.

30. The environment of claim 27, wherein said window is displayed in a transparent mode on said images.

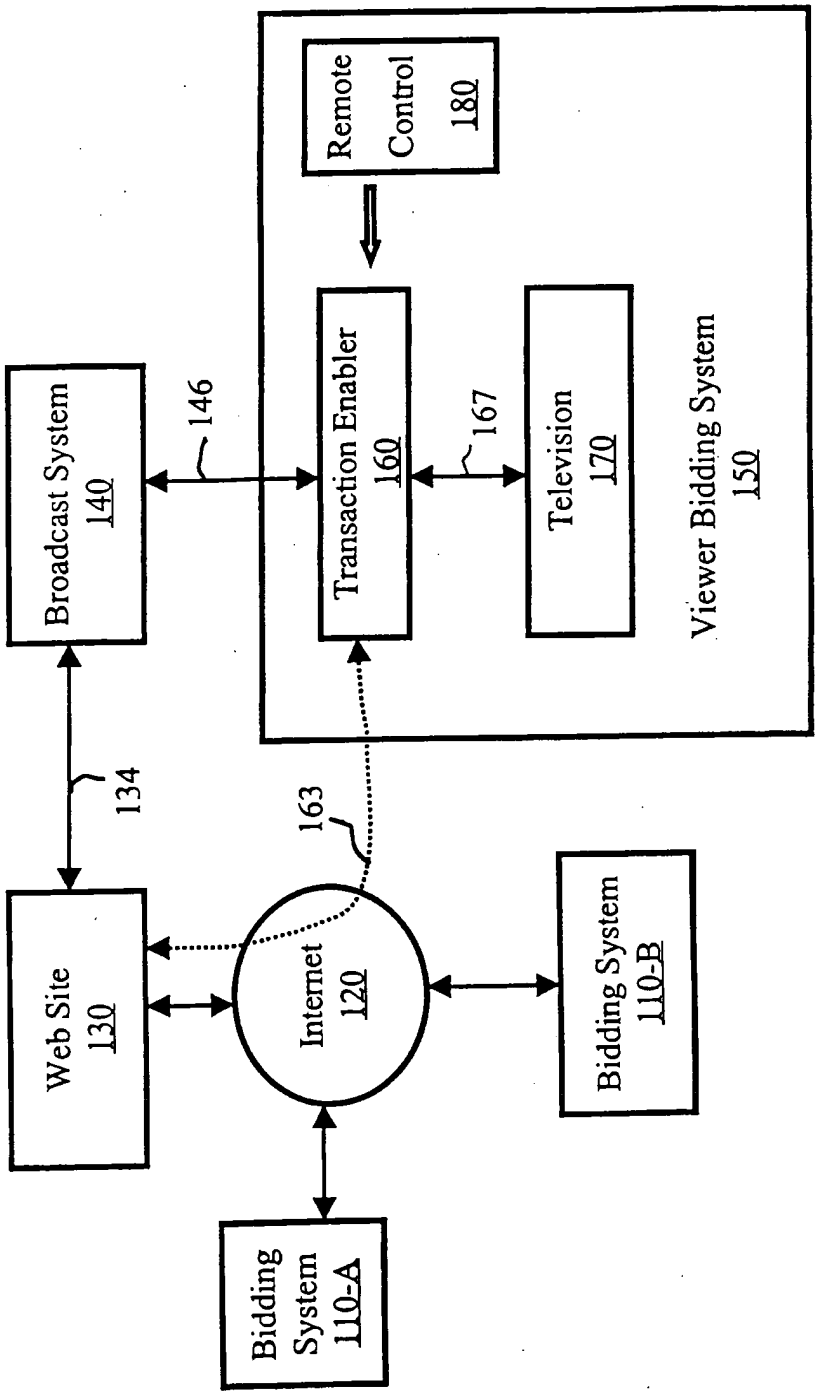

Remote Control 180
Transaction Enabler 160
Television 170
Viewer Bidding System 150
Broadcast System 140
146
167
134
163
Web Site 130
Internet 120
Bidding System 110-B
Bidding System 110-A

Figure 1

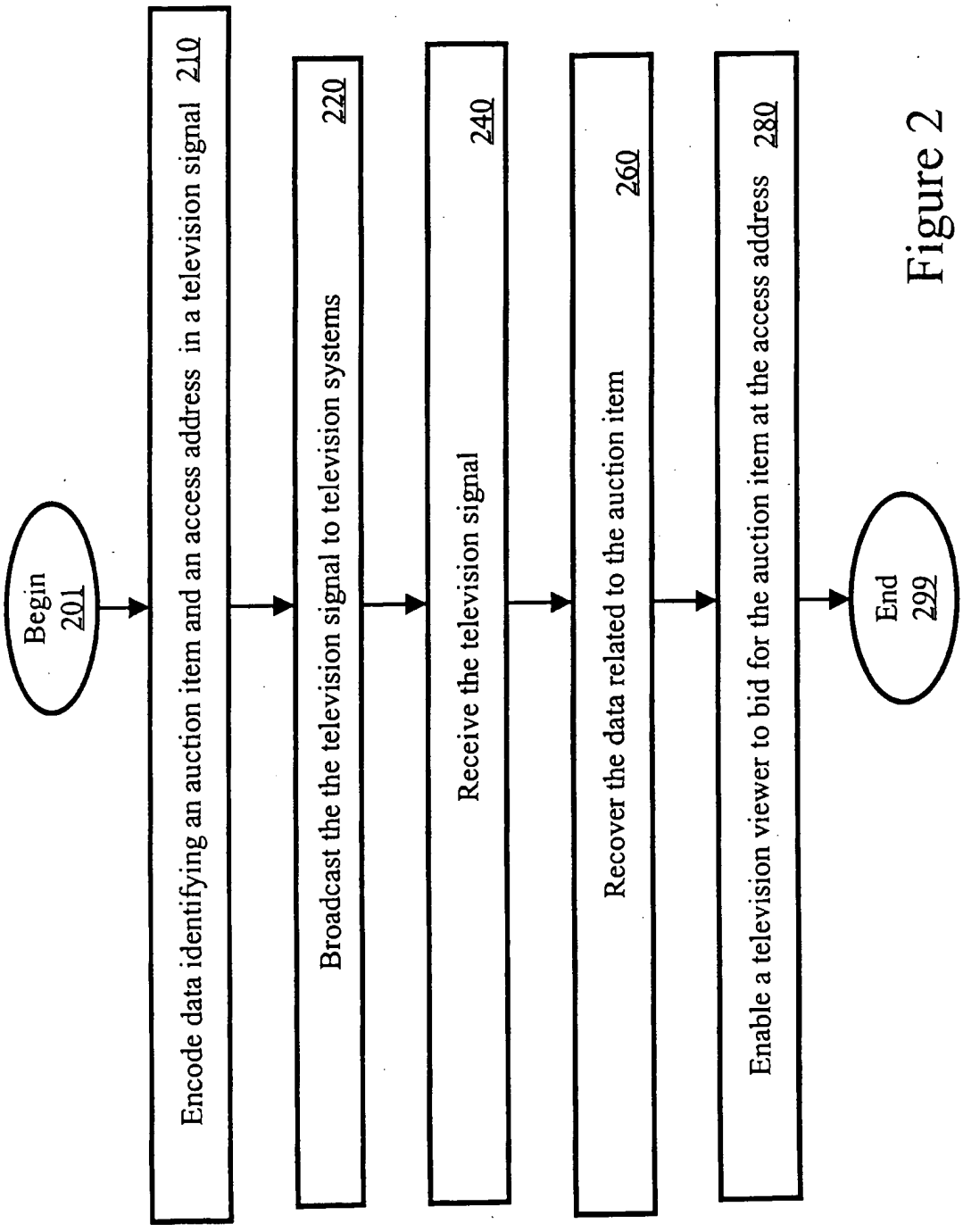

Begin
201
Encode data identifying an auction item and an access address in a television signal 210
Broadcast the the television signal to television systems 220
Receive the television signal 240
Recover the data related to the auction item 260
Enable a television viewer to bid for the auction item at the access address 280
End
299

Figure 2

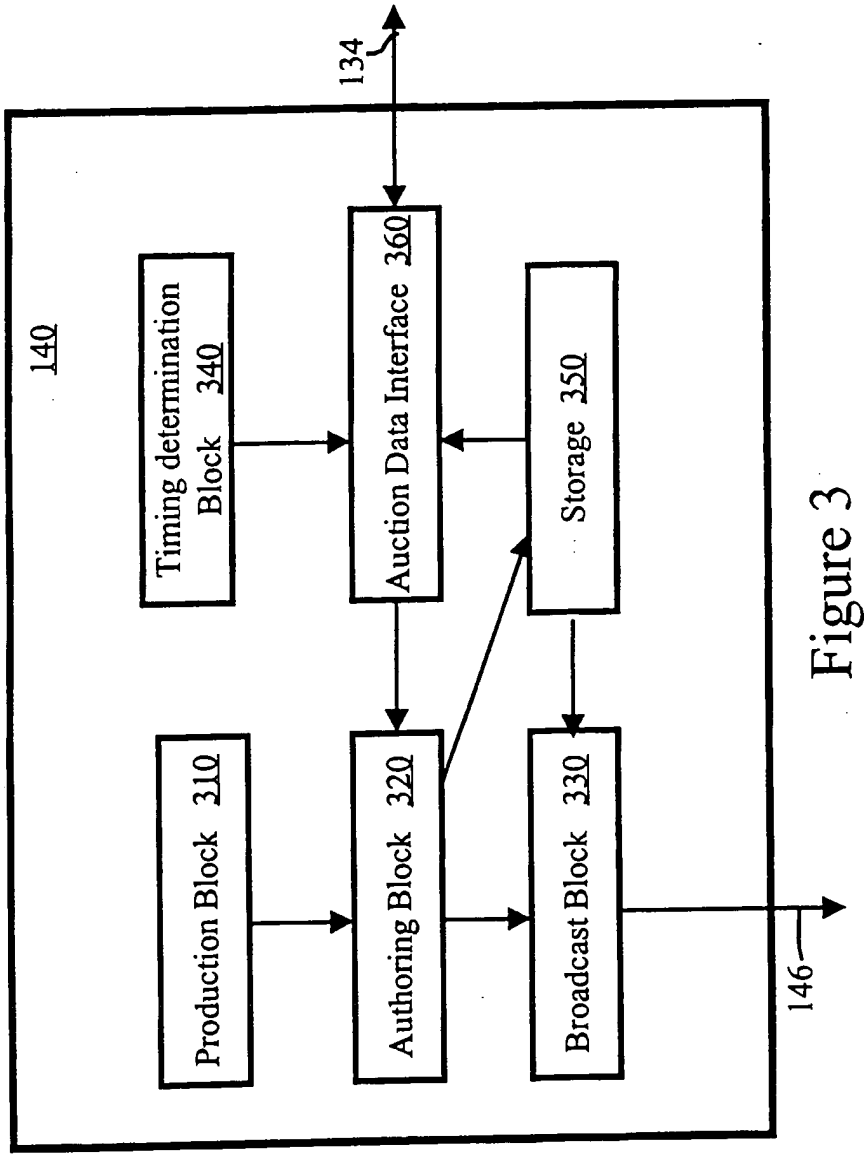

134
140
Timing determination Block 340
Auction Data Interface 360
Storage 350
Production Block 310
Authoring Block 320
Broadcast Block 330
146

Figure 3

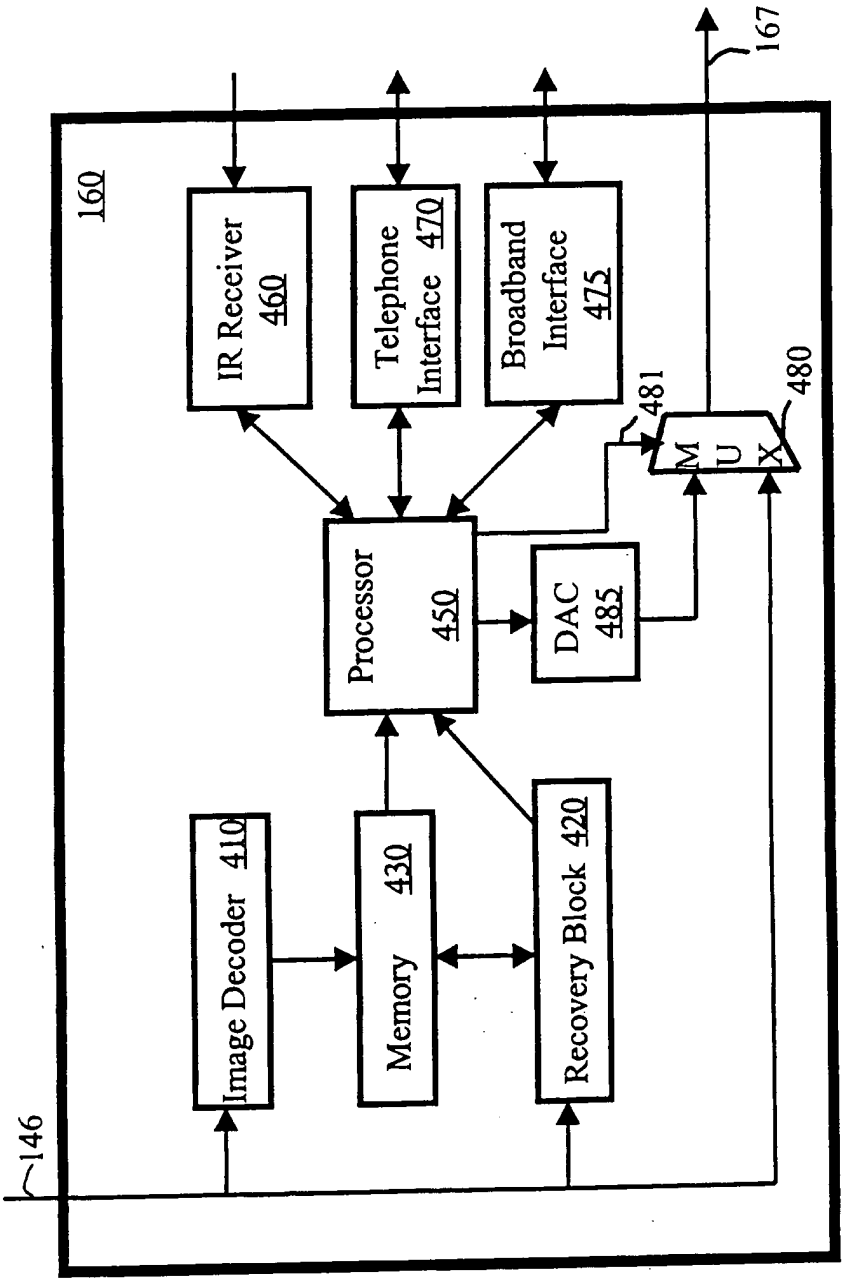

160
IR Receiver 460
Telephone Interface 470
Broadband Interface 475
167
481
480
MUX
Processor 450
DAC 485
Image Decoder 410
Memory 430
Recovery Block 420
146

Figure 4

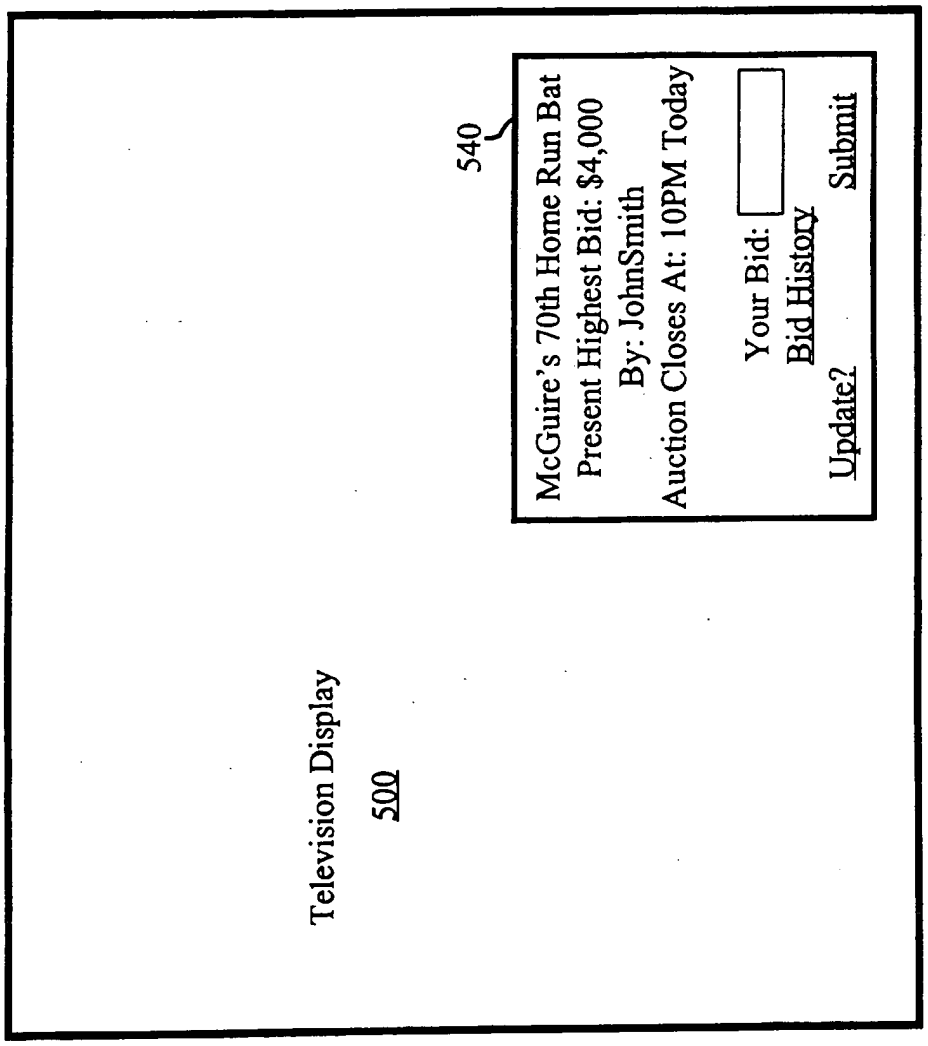

Television Display
500
540
McGuire's 70th Home Run Bat
Present Highest Bid: $4,000
By: JohnSmith
Auction Closes At: 10PM Today
Your Bid:
Bid History
Update?
Submit

Figure 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/18510

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/60
US CL : 705/26, 27, 37

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/26, 27, 37

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Please See Extra Sheet.

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EAST, CORPORATE RESOURCE NET

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	Auction Goes Upscale. Capital District Business Review. April 17, 1995. Vol. 22. Issue 1. page 43.	1-30
Y,E	Strategic Partnership Between ExtraLot.com and The Auction Channel. Business Wire. August 11, 2000.	1-30
Y	Auctioneer Onsale to Broadcast Live Commercials on ZDTV. Electronic Advertising and Marketplace Report. October 6, 1998. Vol 12. Issue 18. page 4.	1-30
Y	Philadelphia Business Journal. Auction Television Does $1 Million Stock Placement. January 29, 1999. Vol. 17. Issue 51. page 36.	1-30

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 22 AUGUST 2000	Date of mailing of the international search report 18 SEP 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer JAMES TRAMMELL Telephone No. (703) 305-9760

Form PCT/ISA/210 (second sheet) (July 1998)*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/18510

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,905,975 A (AUSUBEL) 18 May 1999, col 3, lines 1-30.	1-30
Y	MARQUEZ, RACHELLE. New Dimension For Auction. 15 September 1997. Vol. 15. Issue 20. page 38.	1-30

Form PCT/ISA/210 (continuation of second sheet) (July 1998)*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/18510

B. FIELDS SEARCHED
Documentation other than minimum documentation that are included in the fields searched:

NEWTON'S TELECOM DICTIONARY
McGRAW-HILL ENCYCLOPEDIA OF ELECTRONICS AND COMPUTERS

Search result: 1 of 1

(WO/2004/103843) PACKAGING METHOD AND DEVICE, PACKAGING BAGS

Biblio. Data | Description | Claims | National Phase | Notices | Documents

Latest bibliographic data on file with the International Bureau

Publication Number: WO/2004/103843 **International Application No.:** PCT/FR2004/001185
Publication Date: 02.12.2004 **International Filing Date:** 14.05.2004

Int. Class.: *B65D 33/25* (2006.01), *B65D 85/16* (2006.01)

Applicants: **S2F FLEXICO** [FR/FR]; 1, route de Méru, F-60119 Henonville (FR) *(All Except US)*.
BOIS, Henri, Georges [FR/FR]; 61, boulevard d'Inkermann, F-92200 Neuilly sur Seine (FR) *(US Only)*.

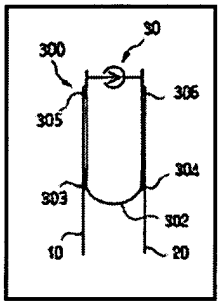
Inventor: **BOIS, Henri, Georges** [FR/FR]; 61, boulevard d'Inkermann, F-92200 Neuilly sur Seine (FR).

Agent: **MARTIN, Jean-Jacques**; Cabinet Regimbeau, 20, rue de Chazelles, F-75847 Paris Cedex 17 (FR).

Priority Data: 03/05887 16.05.2003 FR

Title: PACKAGING METHOD AND DEVICE, PACKAGING BAGS

Abstract: The invention relates to a packaging method comprising the following steps: provision of a bag whose mouth comprises opening/closing means (30) for multiple successive openings and closings and a cleavable linking veil, located at a distance therefrom inside the bag in relation to said opening/closing means (30); introduction of contents (100) to be wrapped in the bag and tightening of said bag in order to close it, tension being applied to the contents (10); the veil (40) enters into contact with the contents (100) avoiding the application of stress on the opening/closing means, guaranteeing free access to the contents (100) via said opening/closing means (30) after tearing, enabling the bag to be relaxed in a closed state as a result of the distance (D) separating the veil (40) and the opening/closing means (30). The invention also relates to a packaging device and to bags thus obtained.



Designated States: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
African Regional Intellectual Property Org. (ARIPO) (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW)
Eurasian Patent Organization (EAPO) (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM)
European Patent Office (EPO) (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR)
African Intellectual Property Organization (OAPI) (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Publication Language: French (FR)

Filing Language: French (FR)

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : H04N 7/16, 7/167 — A1

(11) International Publication Number: **WO 00/46994**

(43) International Publication Date: 10 August 2000 (10.08.00)

(21) International Application Number: PCT/IB00/00163

(22) International Filing Date: 4 February 2000 (04.02.00)

(30) Priority Data:
99400261.6 4 February 1999 (04.02.99) EP

(71) Applicant *(for all designated States except US)*: CANAL+ SOCIETE ANONYME [FR/FR]; 85/89, quai André Citroën, F-75711 Paris (FR).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: MAILLARD, Michel [FR/FR]; 42, avenue du Maréchal Leclerc, F-28130 Maintenon (FR).

(74) Agents: COZENS, Paul, Dennis et al.; Mathys & Squire, 100 Gray's Inn Road, London WC1X 8AL (GB).

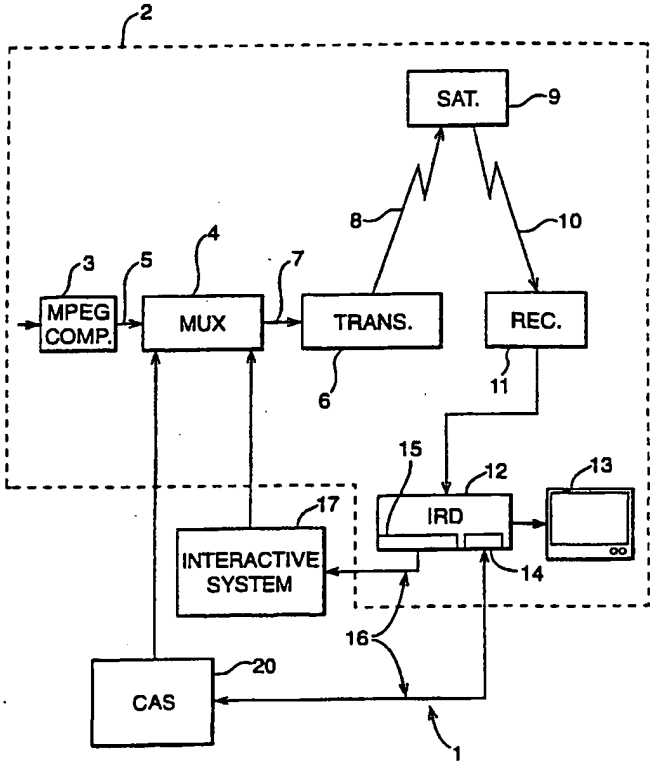
(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD AND APPARATUS FOR ENCRYPTED TRANSMISSION

(57) Abstract

A method and apparatus for encryption of data between a first device (12) and a second device (30), in which one or more precalculated key pairs (41) are stored in a memory of the first device (12), the or each key pair comprising a session key and an encrypted version of the session key. The encrypted version is passed to the second device (30), which decrypts (42) the session key, this session key being thereafter used to encrypt data communicated from the second device (30) to the first device (12) and/or vice versa. The invention is particularly applicable to a digital television system in which data, notably control word data, is to be communicated in encrypted form between a decoder and an associated portable security module.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

METHOD AND APPARATUS FOR ENCRYPTED TRANSMISSION

The present invention relates to a method and apparatus for encryption of messages between two devices, for example a decoder and a portable security module in a digital television system.

Transmission of encrypted data is well-known in the field of pay TV systems, where scrambled audiovisual information is usually broadcast by satellite to a number of subscribers, each subscriber possessing a decoder capable of descrambling the transmitted program for subsequent viewing.

In a typical system, scrambled data is transmitted together with a control word for descrambling of the data, the control word itself being encrypted by a so-called exploitation key and transmitted in encrypted form. The scrambled data and encrypted control word are then received by a decoder having access to an equivalent of the exploitation key stored on a portable security module such as a smart card inserted in the decoder. The encrypted control word is then decrypted on the smart card and subsequently communicated to the decoder for use in descrambling the transmitted data.

In order to try to improve the security of the system, the control word is usually changed every ten seconds or so. This avoids the situation with a static or slowly changing control word where the control word may become publicly known. In such circumstances, it would be relatively simple for a fraudulent user to feed the know control word to the descrambling unit on his decoder to descramble the transmission.

Notwithstanding this security measure, a problem has arisen in recent years where the stream of control words sent during a broadcast becomes known through monitoring of data communicated at the interface between the smart card and decoder. This information may be used by any unauthorised user who has recorded the still-scrambled broadcast on a video recorder. If the film is replayed at the same time as the stream of control words is fed to the decoder, visualisation of the broadcast

becomes possible. This problem has further been exacerbated with the rise of the internet and it is now common to find any number of internet sites that list the stream of control words emitted during a given transmission.

The European patent application PCT WO 97/3530 in the name of Digco addresses this problem by proposing a solution in which the control word stream passed across the interface between the smart card and decoder is itself encrypted with a session key. The session key is generated randomly by the decoder and encrypted with a second key held in the decoder and corresponding to a public key used with a private/public encryption algorithm. The associated smart card possesses the necessary private key to decrypt the session key, which is thereafter used by the smart card to encrypt the control word stream sent from the smart card to the decoder.

As will be appreciated, the use of a locally generated session key to encrypt the control word stream means that the encrypted stream cannot thereafter be fed into another decoder for use in descrambling the data since each decoder will possess a different session key for use in decrypting the control word stream sent from the smart card.

Whilst this solution provides a higher level of security than conventional systems there are nevertheless a number of disadvantages associated with this system.

Notably, the use of a public/private key algorithm is effectively obligatory in such a system since it is not desirable for security reasons to store both a symmetric key and the associated algorithm in the decoder, due to the ease in which this information may be extracted from a decoder memory. This problem does not arise in the case of a public key, since possession of this key does not enable decryption of private key encrypted messages.

It is one object of the present invention to provide a more adaptable alternative to the above known system. However, the invention is not limited to the field of decoder security and, as will be described below, may be applied to a number of other

situations in which secure communication of data is required.

A first aspect of the present invention provides a method of encryption of data communicated between a first and second device, wherein at least one precalculated key pair is stored in a memory of the first device, said at least one key pair comprising a session key and an encrypted version of the session key prepared using a transport key, the encrypted version of the session key being subsequently communicated to the second device which decrypts the encrypted version using an equivalent transport key stored in its memory such that data communicated from at least the second to the first device may thereafter be encrypted and decrypted by the session key in the respective devices.

A preferred embodiment provides a method of encryption of data communicated between a first and second device, characterised in that one or more precalculated key pairs are stored in a memory of the first device, the or each key pair comprising a session key and an encrypted version of this session key prepared using a transport key, the encrypted value of the session key being subsequently communicated to the second device which decrypts this value using an equivalent transport key stored in its memory such that data communicated from at least the second to the first device may thereafter be encrypted and decrypted by the session key in the respective devices.

Unlike the Digco system described above, the use of a precalculated stored pair of values avoids the necessity of having to provide an encryption algorithm within the first device (e.g. the decoder) to encrypt an internally generated session key. As a consequence, the algorithm chosen to encrypt the session key need not be limited to a public/private key algorithm but may correspond to a symmetric type algorithm if desired. Nevertheless, as will be understood, the present invention may also be implemented using public/private key algorithms to encrypt the session key, as will be discussed in further detail below.

Advantageously, a plurality of key pairs are stored in the memory of the first device,

the first device selecting and processing one or more session keys to generate a definitive session key and communicating the associated encrypted value or values to the second device for decryption and processing by the second device to generate the definitive session key.

The provision of a plurality of key pairs within the first device enables the first device to choose and define a different definitive session key for each communication session. In one embodiment, a subset of a plurality of stored session keys is chosen by the first device to generate the definitive session key, the associated encrypted values of these subset session keys being communicated to the second device for decryption and processing.

Depending on the type of operation used, the resulting definitive session key may be dependent on the order of combination of the chosen session keys. In such an embodiment, this order information is communicated to the second device to enable the second device to correctly generate the definitive session key using the associated encrypted values.

For example, an initial session key value known to both the first and second devices may be repeatedly encrypted in both devices by an ordered sequence of session keys using an encryption algorithm sensitive to the order of encryption, such as the DES symmetric algorithm.

Of course, where the first device is using a selected subset of keys to generate the definitive session key, it may not be necessary to also use an order dependent algorithm to generate a changeable definitive session key and the keys may be combined, for example, using a simple arithmetical operation.

In one advantageous embodiment, the one or more precalculated key pair values may be selected from a larger set of precalculated key pairs prior to storage in the first device. For example, the operator or system manager may communicate a large number of precalculated key pairs to the manufacturer of the first device, the device

manufacturer thereafter selecting at random the key pairs to be stored in a given device.

In this way, the key pair or pairs embedded in the first device will be unique to that device, or at least quasi-unique, thereby increasing the level of security for the system. Furthermore, the entity responsible for manufacture of the device need not possess the algorithm or keys used to prepare the encrypted session key values but may be simply supplied with a table of key pairs.

Preferably, the encrypted key value or values communicated to the second device also include a signature value that may be read by the second device to verify the authenticity of the communicated value.

Such a signature value can be generated and verified in accordance with a conventional signature system, for example using combination of hash and public/private key algorithms such as MD5 and RSA, this signature being appended to the key pair values stored in the first device.

Conveniently, the signature value can also be precalculated at the time of calculation of the encrypted key value and thereafter stored in the first device.

In a particularly preferred embodiment, the algorithm and transport key used to encrypt and decrypt the session key or keys correspond to a symmetric algorithm and associated symmetric key. The use of a symmetric algorithm enables an increase in the processing time necessary for the second device to decrypt the session key in comparison with an operation using a public/private key algorithm.

Whilst one of the advantages of the present invention lies in the adaptability of the present system to use a symmetric algorithm, it will be appreciated that this is not obligatory. For example, in an alternative embodiment, the session key or keys may be encrypted by a public key prior to storage in the first device and decrypted by an equivalent private key within the second device.

Further preferably, the encryption algorithm used with the session key to encrypt and decrypt data communicated between the first and second device (or vice versa) corresponds to a symmetric algorithm. The choice of algorithm used may depend on the system requirements such as the need to have bidirectional communication between the devices.

Suitable symmetric algorithms may include DES or even an appropriate proprietary algorithm. Suitable public/private key algorithms may comprise RSA or other similar algorithms.

As mentioned above, the present invention is particularly applicable to the field of digital television and, in one preferred embodiment, the first device corresponds to a decoder and the second device to a portable security module (or vice versa).

The portable security module may conveniently comprise a smart card. If so, the data encrypted with the session key may correspond to simple control word information used by the decoder to descramble broadcast data.

The same principle may also be applied to the case where the descrambling unit in the decoder is implemented as a detachable conditional access module or CAM, broadcast data being descrambled in the conditional access module and communicated to the decoder.

In this embodiment, the first device may thus correspond to a decoder and the second device to a detachable conditional access module. If so, the data encrypted with the session key will normally correspond to the data descrambled by the conditional access module e.g. the broadcast programme itself.

In a conditional access module implementation, a smart card may also form part of the system, this card being inserted in the conditional access module to decrypt the control word, which is then passed to the conditional access module to permit descrambling of the broadcast programme. If so, the first device may then correspond to a

conditional access module, the second device to a smart card and the data encrypted with the session key to control word data.

Within the field of digital television, the invention may also be applied to the communication of data between a decoder and other devices, such as a television or video recorder. In particular, in one embodiment, the first device corresponds to a first decoder and the second device to a second decoder.

In households possessing a first and second decoder, there are often a number of problems associated with maintaining communication between a first or "master" decoder and a second "slave" decoder. The use of a secure encrypted link to communicate audiovisual data, control word data, or even data relating to current subscription rights and exploitation keys, may prove useful in this context.

In yet a further realisation, the present invention may be applied to home network system, the first and second devices corresponding to first and second consumer electronic devices adapted to transfer data via a communication link (e.g. radio, PLC, infra-red etc.).

The above embodiments have been described in relation to a method of encryption of data. Viewed from another aspect, the invention may equally be applied to first and second devices adapted to carry out such a method.

Another aspect of the present invention provides a system for providing secure communication of data between first and second devices, said first device comprising a memory for storing at least one precalculated key pair comprising a session key and an encrypted version of the session key prepared using a transport key, and communication means, such as a communication link, for communicating the encrypted version of the session key to said second device, said second device comprising a memory for storing an equivalent transport key, decryption means, such as a processor, for decrypting said encrypted version of the session key using said equivalent transport key, and means, such as the processor, for encrypting data to be

communicated to said first device using said session key.

Features described above relating to method aspects of the present invention can also be applied to device or system aspects, and vice versa.

As used above, the terms "portable security module", "smart card" and "conditional access module" may be interpreted in their broadest sense as applying to any portable microprocessor and/or memory based card capable of carrying out the described functions.

As particular examples of such devices, a smart card may correspond to a card device constructed in accordance with the known international standards ISO 7816-1, 7816-2 and 7816-3 whilst the conditional access module may be implemented as a PCMCIA or PC card corresponding to the standards fixed by the PCMCIA group. Other physical shapes and forms are of course possible.

The terms "scrambled" and "encrypted" and "control word" and "key" have been used at various parts in the text for the purpose of clarity of language. However, it will be understood that no fundamental distinction is to be made between "scrambled data" and "encrypted data" or between a "control word" and a "key".

Similarly, unless obligatory in view of the context stated or unless otherwise specified, no limitation to either symmetric or public/private algorithms is to be inferred for a given encryption and/or decryption process. In the same way, whilst the matching keys used in encrypting and decrypting information may be referred to by the same name (e.g. "transport key", "session key") it is to be understood that these need not be numerically identical keys as long as they fulfil their functions. For example, the corresponding public and private keys used to encrypt and decrypt data will normally possess numerically different values.

The term "receiver/decoder" or "decoder" as used herein may connote a receiver for receiving either encoded or non-encoded signals, for example, television and/or radio

signals, which may be broadcast or transmitted by any appropriate means. Embodiments of such decoders may also include a decoder integral with the receiver for decoding the received signals, for example, in a "set-top box", a decoder functioning in combination with a physically separate receiver, or such a decoder including additional functions, such as a web browser, integrated with other devices such as a video recorder or a television.

As used herein, the term "digital transmission system" includes any transmission system for transmitting or broadcasting for example primarily audiovisual or multimedia digital data. Whilst the present invention is particularly applicable to a broadcast digital television system, the invention may also be applicable to a fixed telecommunications network for multimedia internet applications, to a closed circuit television, and so on.

As used herein, the term "digital television system" includes for example any satellite, terrestrial, cable and other system.

There will now be described, by way of example only, a number of embodiments of the invention, with reference to the following figures, in which:

Figure 1 shows by way of background the overall architecture of a digital TV system;

Figure 2 shows the architecture of the conditional access system of Figure 1;

Figure 3 shows a method of encryption of data between a smart card and a decoder according to this embodiment of the invention;

Figure 4 shows the generation of a session key in a decoder operating according to the embodiment of Figure 3; and

Figure 5 shows the steps in the preparation of a session key in a smart card interfacing with the decoder of Figure 4.

The present invention describes a method of encryption of data, in particular but not exclusively applicable to the encryption of data across the interface between a portable security module and decoder in a digital television system. By way of background, the architecture of a known digital television system will now be described.

Digital Television System

An overview of a digital television system 1 is shown in Figure 1 comprising a broadcast system 2 which uses the MPEG-2 compression system to transmit compressed digital signals. In more detail, an MPEG-2 compressor 3 in a broadcast centre receives a digital signal stream (for example a stream of audio or video signals). The compressor 3 is connected to a multiplexer and scrambler 4 by linkage 5. The multiplexer 4 receives a plurality of further input signals, assembles one or more transport streams and transmits compressed digital signals to a transmitter 6 of the broadcast centre via linkage 7, which can of course take a wide variety of forms including telecom links.

The transmitter 6 transmits electromagnetic signals via uplink 8 towards a satellite transponder 9, where they are electronically processed and broadcast via a notional downlink 10 to earth receiver 11, conventionally in the form of a dish owned or rented by the end user. The signals received by receiver 11 are transmitted to an integrated receiver/decoder 12 owned or rented by the end user and connected to the end user's television set 13. The receiver/decoder 12 decodes the compressed MPEG-2 signal into a television signal for the television set 13.

A conditional access system 20 is connected to the multiplexer 4 and the receiver/decoder 12, and is located partly in the broadcast centre and partly in the decoder. It enables the end user to access digital television broadcasts from one or more broadcast suppliers. A portable security module in the form of a smartcard capable of decrypting messages relating to broadcast programmes or data can be inserted into the receiver/decoder 12.

An interactive system 17, also connected to the multiplexer 4 and the receiver/decoder 12 and again located partly in the broadcast centre and partly in the decoder, may be provided to enable the end user to interact with various applications via a modemmed back channel 16.

The conditional access system 20 will now be described in more detail. With reference to Figure 2, in overview the conditional access system 20 includes a Subscriber Authorization System (SAS) 21. The SAS 21 is connected to one or more Subscriber Management Systems (SMS) 22, one SMS for each broadcast supplier, for example by a respective TCP-IP linkage 23 (although other types of linkage could alternatively be used). Alternatively, one SMS could be shared between two broadcast suppliers, or one supplier could use two SMSs, and so on.

First encrypting units in the form of ciphering units 24 utilising "mother" smartcards 25 are connected to the SAS by linkage 26. Second encrypting units again in the form of ciphering units 27 utilising mother smartcards 28 are connected to the multiplexer 4 by linkage 29. The receiver/decoder 12 receives a portable security module, for example in the form of "daughter" smartcard 30. It is connected directly to the SAS 21 by Communications Servers 31 via the modemmed back channel 16. The SAS sends, amongst other things, subscription rights to the daughter smartcard on request.

The smartcards contain the secrets of one or more commercial operators. The "mother" smartcard encrypts different kinds of messages and the "daughter" smartcards decrypt the messages, if they have the rights to do so.

The first and second ciphering units 24 and 27 comprise a rack, an electronic VME card with software stored on an EEPROM, up to 20 electronic cards and one smartcard 25 and 28 respectively, for each electronic card, one card 28 for encrypting the ECMs and one card 25 for encrypting the EMMs.

The operation of the conditional access system 20 of the digital television system will now be described in more detail with reference to the various components of the

television system 2 and the conditional access system 20.

Multiplexer and Scrambler

With reference to Figures 1 and 2, in the broadcast centre, the digital audio or video signal is first compressed (or bit rate reduced), using the MPEG-2 compressor 3. This compressed signal is then transmitted to the multiplexer and scrambler 4 via the linkage 5 in order to be multiplexed with other data, such as other compressed data.

The scrambler generates a control word used in the scrambling process and included in the MPEG-2 stream in the multiplexer. The control word is generated internally and enables the end user's integrated receiver/decoder 12 to descramble the programme.

Access criteria, indicating how the programme is commercialised, are also added to the MPEG-2 stream. The programme may be commercialised in either one of a number of "subscription" modes and/or one of a number of "Pay Per View" (PPV) modes or events. In the subscription mode, the end user subscribes to one or more commercial offers, or "bouquets", thus getting the rights to watch every channel inside those bouquets. In the preferred embodiment, up to 960 commercial offers may be selected from a bouquet of channels.

In the Pay Per View mode, the end user is provided with the capability to purchase events as he wishes. This can be achieved by either pre-booking the event in advance ("pre-book mode"), or by purchasing the event as soon as it is broadcast ("impulse mode"). In the preferred embodiment, all users are subscribers, whether or not they watch in subscription or PPV mode, but of course PPV viewers need not necessarily be subscribers.

Entitlement Control Messages

Both the control word and the access criteria are used to build an Entitlement Control

Message (ECM). This is a message sent in relation with a scrambled program; the message contains a control word (which allows for the descrambling of the program) and the access criteria of the broadcast program. The access criteria and control word are transmitted to the second encrypting unit 27 via the linkage 29. In this unit, an ECM is generated, encrypted and transmitted on to the multiplexer and scrambler 4. During a broadcast transmission, the control word typically changes every few seconds, and so ECMs are also periodically transmitted to enable the changing control word to be descrambled. For redundancy purposes, each ECM typically includes two control words; the present control word and the next control word.

Each service broadcast by a broadcast supplier in a data stream comprises a number of distinct components; for example a television programme includes a video component, an audio component, a sub-title component and so on. Each of these components of a service is individually scrambled and encrypted for subsequent broadcast to the transponder 9. In respect of each scrambled component of the service, a separate ECM is required. Alternatively, a single ECM may be required for all of the scrambled components of a service. Multiple ECMs are also generated in the case where multiple conditional access systems control access to the same transmitted program.

Entitlement Management Messages (EMMs)

The EMM is a message dedicated to an individual end user (subscriber), or a group of end users. Each group may contain a given number of end users. This organisation as a group aims at optimising the bandwidth; that is, access to one group can permit the reaching of a great number of end users.

Various specific types of EMM can be used. Individual EMMs are dedicated to individual subscribers, and are typically used in the provision of Pay Per View services; these contain the group identifier and the position of the subscriber in that group.

Group subscription EMMs are dedicated to groups of, say, 256 individual users, and are typically used in the administration of some subscription services. This EMM has a group identifier and a subscribers' group bitmap.

Audience EMMs are dedicated to entire audiences, and might for example be used by a particular operator to provide certain free services. An "audience" is the totality of subscribers having smartcards which bear the same conditional access system identifier (CA ID). Finally, a "unique" EMM is addressed to the unique identifier of the smartcard.

EMMs may be generated by the various operators to control access to rights associated with the programs transmitted by the operators as outlined above. EMMs may also be generated by the conditional access system manager to configure aspects of the conditional access system in general.

The term EMM is also often used to describe specific configuration type messages communicated between the decoder and other elements of the system and, for example, will be used later in this application to refer to a specific message passed from the decoder to a smart card.

Subscriber Management System (SMS)

A Subscriber Management System (SMS) 22 includes a database 32 which manages, amongst others, all of the end user files, commercial offers, subscriptions, PPV details, and data regarding end user consumption and authorization. The SMS may be physically remote from the SAS.

Each SMS 22 transmits messages to the SAS 21 via respective linkage 23 which imply modifications to or creations of Entitlement Management Messages (EMMs) to be transmitted to end users.

The SMS 22 also transmits messages to the SAS 21 which imply no modifications or

creations of EMMs but imply only a change in an end user's state (relating to the authorization granted to the end user when ordering products or to the amount that the end user will be charged).

The SAS 21 sends messages (typically requesting information such as call-back information or billing information) to the SMS 22, so that it will be apparent that communication between the two is two-way.

Subscriber Authorization System (SAS)

The messages generated by the SMS 22 are passed via linkage 23 to the Subscriber Authorization System (SAS) 21, which in turn generates messages acknowledging receipt of the messages generated by the SMS 21 and passes these acknowledgements to the SMS 22.

In overview the SAS comprises a Subscription Chain area to give rights for subscription mode and to renew the rights automatically each month, a Pay Per View Chain area to give rights for PPV events, and an EMM Injector for passing EMMs created by the Subscription and PPV chain areas to the multiplexer and scrambler 4, and hence to feed the MPEG stream with EMMs. If other rights are to be granted, such as Pay Per File (PPF) rights in the case of downloading computer software to a user's Personal Computer, other similar areas are also provided.

One function of the SAS 21 is to manage the access rights to television programmes, available as commercial offers in subscription mode or sold as PPV events according to different modes of commercialisation (pre-book mode, impulse mode). The SAS 21, according to those rights and to information received from the SMS 22, generates EMMs for the subscriber.

The EMMs are passed to the Ciphering Unit (CU) 24 for ciphering with respect to the management and exploitation keys. The CU completes the signature on the EMM and passes the EMM back to a Message Generator (MG) in the SAS 21, where a header

is added. The EMMs are passed to a Message Emitter (ME) as complete EMMs. The Message Generator determines the broadcast start and stop time and the rate of emission of the EMMs, and passes these as appropriate directions along with the EMMs to the Message Emitter. The MG only generates a given EMM once; it is the ME which performs cyclic transmission of the EMMs.

On generation of an EMM, the MG assigns a unique identifier to the EMM. When the MG passes the EMM to the ME, it also passes the EMM ID. This enables identification of a particular EMM at both the MG and the ME.

Programme Transmission

The multiplexer 4 receives electrical signals comprising encrypted EMMs from the SAS 21, encrypted ECMs from the second encrypting unit 27 and compressed programmes from the compressor 3. The multiplexer 4 scrambles the programmes and sends the scrambled programmes, the encrypted EMMs and the encrypted ECMs to a transmitter 6 of the broadcast centre via the linkage 7. The transmitter 6 transmits electromagnetic signals towards the satellite transponder 9 via uplink 8.

Programme Reception

The satellite transponder 9 receives and processes the electromagnetic signals transmitted by the transmitter 6 and transmits the signals on to the earth receiver 11, conventionally in the form of a dish owned or rented by the end user, via downlink 10. The signals received by receiver 11 are transmitted to the integrated receiver/decoder 12 owned or rented by the end user and connected to the end user's television set 13. The receiver/decoder 12 demultiplexes the signals to obtain scrambled programmes with encrypted EMMs and encrypted ECMs.

If the programme is not scrambled, that is, no ECM has been transmitted with the MPEG-2 stream, the receiver/decoder 12 decompresses the data and transforms the signal into a video signal for transmission to television set 13.

If the programme is scrambled, the receiver/decoder 12 extracts the corresponding ECM from the MPEG-2 stream and passes the ECM to the "daughter" smartcard 30 of the end user. This slots into a housing in the receiver/decoder 12. The daughter smartcard 30 controls whether the end user has the right to decrypt the ECM and to access the programme. If the end user does have the rights, the ECM is decrypted within the smart card and the control word extracted.

Thereafter the smart card then communicates the control word to the decoder 12 which then descrambles the programme using this control word. In most conventional systems, the control word is communicated across the smart card interface in a clear or non-encrypted form, leading to the problems of security described in the introduction of the present application. After descrambling by the decoder, the MPEG-2 stream is decompressed and translated into a video signal for onward transmission to television set 13.

In the system described above, the descrambling of the MPEG data is carried out within the decoder using the control word information communicated to the decoder from the smart card. In other systems, the descrambling circuitry may be implemented in a detachable conditional access module or CAM, commonly embodied in the form of a PCMCIA or PC card insertable in a socket in the decoder.

The CAM module may itself further include a slot to receive a smart card. In such systems, control word data is decrypted in the smart card communicated to the CAM module which then descrambles the scrambled MPEG data stream to supply the decoder with a clear MPEG stream for decompression and subsequent display.

In this type of system, sensitive data may be passed between the smart card and CAM (control word data) and/or between the CAM and decoder (descrambled MPEG data) and problems of security may arise at either of these interfaces.

Data Encryption across an Interface

Referring to Figure 3, there will now be described a method of data encryption as applied to the control word data communicated between a smart card and a decoder in one of the simplest embodiments of this invention. However, the same principles may be applied to the encryption of control word data between a smart card and a CAM, audiovisual MPEG data between a CAM and a decoder, or indeed any type of data between two such devices.

In accordance with the present invention, a set of key pairs is stored in a non-volatile memory of the decoder e.g. a FLASH memory. Each key pair corresponds to a key value in clear form and an encrypted version of the key. As will be described, the encrypted version of the key will be eventually communicated in an EMM message sent to a smart card inserted in the decoder.

Thus, within the decoder a set of EMM message/key pairs are stored as follows:

n	EMM (19 octets)	Key (8 octets)
1	EMM(1)	Key(1)
2	EMM(2)	Key(2)
3	EMM(3)	Key(3)
.	.	.
.	.	.
.	.	.
.	.	.
16	EMM(16)	Key(16)

The encrypted value of the key stored in the EMM is calculated external of the decoder using an encryption algorithm not present in the decoder. In the present example the key values Key(1), Key(2) etc. correspond to symmetric keys to be used with a symmetric encryption algorithm such as DES.

The encryption algorithm used to prepare the encrypted DES key values contained with the stored EMM messages may also correspond to a symmetric encryption algorithm. For increased security, a proprietary symmetric algorithm (PSA) different from DES will be used to prepare the encrypted values, although in another

embodiment DES may also be used to encrypt the key values.

In addition to the encrypted value of the associated key, the EMM message may also include a signature value associated with the message and prepared as per any conventional signature preparation method. For example, a message may be subject to a hash function such as MD5 followed by encryption of the hash value by a private key of private/public key algorithm such as RSA. Verification of the signature may then be carried out at the point of reception using a MD5 algorithm and the corresponding public key of the private/public key pair.

The EMM message will additionally include a standard smart card header element (as defined by the international standard ISO 7816-3) to place the message in a format necessary to permit it to be read by a smart card. An EMM associated with an 8 byte key will therefore typically have the following structure:

Header	5 bytes
Encrypted key	10 bytes
Signature	9 bytes

In the present embodiment a set of 16 key/message pairs are implanted in the memory of the decoder. Alternative embodiments are equally possible using more or less key/message pairs and the invention may even be implemented using a single key/message pair. Whilst it may be envisaged that all decoders are equipped with the same key/message pairs it is preferred for security reasons that each decoder has a unique set of key/message pairs. In implementing this embodiment, an operator may supply to a decoder manufacturer a set of ten thousand or more key/message pairs, the decoder manufacturer taking a random selection of 16 pairs during the personalisation of each decoder.

In order to increase the security, a different subset of the message/key pairs stored in the decoder will be used during each session. A session may be defined as corresponding to each time the decoder is switched on and off, or each time the

decoder changes channel, for example.

Referring to Figure 3, a random number generator 40 within the decoder selects 8 out of the 16 message/key pairs to be used in that session. The 8 selected EMM messages 41 of the pairs are then communicated to the smart card 30 to be verified and decrypted and processed as shown at 42 and 43 to obtain the appropriate session key (see below). The same key generation operation is carried out within the decoder at 43 using the corresponding key values of the pairs so as to obtain the same session key value.

The generation of the session key within the decoder will now be described with reference to Figure 4.

A base session key value KeyS Initial shown at 44 and constant for all decoders is encrypted at 45 by the first key 46 of the subset chosen by the random generator 40. The resulting value is then encrypted at 47 using the second key 48 of the session subset and the operation repeated just until the last encryption operation 49 carried out with the last key 50 of the subset so as to obtain the final session key value shown at 51.

The initial session key value KeyS Initial can be a universal value present in all decoders and smart cards, a value linked to a specific decoder/smart card pair or even a value generated at the start of each session in the decoder and thereafter communicated to the smart card.

In the example given above, the session key is prepared by a sequence of repeated operations on the KeyS Initial using the DES algorithm and the selected keys 46, 48, 50 etc. In the case of the DES algorithm, the order in which the keys are applied is important and must be respected to produce the same key each time.

However, whilst the session key S is itself a numerical value that will be used as a DES key in the subsequent decryption operation (see below), the steps used to

generate this key value need not correspond to DES encryption steps. Instead, the subset of keys chosen by the random number generator may be combined together in any number of ways to arise at a suitable session key value KeyS Final. For example, the keys may be combined using a sequence of simple arithmetic operations. Depending on the method chosen, it may not be necessary that the order of the steps in the preparation of the KeyS be respected in order to regenerate the same key.

Referring now to Figure 5, the decryption and processing operations 42 and 43 carried out in the smart card 30 to generate the session key used by the smart card will now be described.

Upon insertion of the smart card in the decoder, the subset of EMM messages matching the selected key values are sent to the smart card. Authentification of each EMM messages is first carried out with reference to the attached signature value, using for example an MD5/RSA type process as described above. For simplicity, this step has been omitted from Figure 5.

The first EMM message 60 is then decrypted at 61 using a transport key 59 embedded in a secure and non-readable manner within the smart card. As mentioned above, for security reasons the algorithm used in the decryption 61 of the EMM message may correspond to a proprietary security algorithm PSA known only to the operator responsible for preparation of the message/key pairs used in the decoder and the personalisation of the smart card.

The transport key KeyT shown at 59 may be a key value common to all smart cards in the system or unique to one such card. The use of a unique key value KeyT requires that the message/key table stored in the decoder be prepared with the same key as that in the card, such that a decoder and card will be irreversibly linked together. In practice, this may not be desirable.

A similar decryption operation using the transport key 59 is then carried out at 62 on the next EMM message 63 in the series and 50 on until the last decryption operation

64 on the final EMM message 65.

In the present embodiment, encryption of each of the EMM messages 60, 63, 65 produces keys 46, 48, 50 identical to those associated in the message/key table present in the decoder and used for generation of the session key as described previously. For this reason, the same reference numbers have been used for these keys and for the key generation operation 43 also carried out in the decoder. Similarly, the same initial session key 44 present in the decoder is also stored in the smart card.

The initial session key KeyS Initial shown at 44 is then encrypted at 45 by the first key 46, the result re-encrypted at 47 by the second key 48 and so on until the final encryption step carried out at 49 using the last key 50 in the series so as to obtain the final session key at 51.

Both the decoder and smart card now possess the same session key KeyS which may thereafter be used in encrypting and decrypting data passed in either direction between the two devices.

Referring back to Figure 3, the smart card 30 receives an encrypted ECM message containing the control word necessary for descrambling an associated segment of MPEG audiovisual or other data. The smart card decrypts the ECM at 71 to obtain the control word value CW.

In passing, we note that the algorithm used to encrypt ECM messages for a user may conveniently correspond to the Proprietary Security Algorithm used for decryption of the EMM messages received from the smart card as described above.

The decrypted control word is then re-encrypted at 72 using the session key KeyS and the encrypted control word value f(CW) transmitted over the decoder/smart card interface as shown. The encrypted value f(CW) is then decrypted at 73 using the session key KeyS held in the decoder and the clear value of the control word CW obtained at 74.

As the session key is symmetric, it may equally be used in the encryption of data transmitted from the decoder to the smart card. Furthermore, the data transmitted from the smart card to the decoder may be data other than simple control word data.

As mentioned above, the same principle may be applied across all interfaces in a system comprising a decoder in which a detachable CAM module is inserted (decoder/CAM interface, CAM/smart card interface etc.). Similarly, the same principle may be applied in the case of a portable module (either a CAM type module or a smart card) inserted in other devices such as a television or video recorder.

In fact, the above method of setting up an encrypted communication channel may be applied to any pair of devices where security of data communication is required. In particular, the same principle may be applied in a home network system where multiple consumer devices (television, video, PC, decoder etc.) transfer data such as audiovisual data or computer files via a communication link. This may be an RF link, an infrared link, a dedicated bus, a power line connection etc. For example, it may be desired to transmit control word in other data in an encrypted form between a decoder and a television or between a master decoder and a slave decoder in the same household.

Other examples of systems of this type where a secure communication link would be desirable will also be apparent to the reader.

CLAIMS

1. A method of encryption of data communicated between a first and second device, wherein at least one precalculated key pair is stored in a memory of the first device, said at least one key pair comprising a session key and an encrypted version of the session key prepared using a transport key, the encrypted version of the session key being subsequently communicated to the second device which decrypts the encrypted version using an equivalent transport key stored in its memory such that data communicated from at least the second to the first device may thereafter be encrypted and decrypted by the session key in the respective devices.

2. A method as claimed in claim 1, in which a plurality of key pairs are stored in the memory of the first device, the first device selecting and processing at least one session key to generate a definitive session key and communicating the associated encrypted version of said at least one session key to the second device for decryption and processing by the second device to generate the definitive session key.

3. A method as claimed in claim 2 in which a subset of a plurality of stored session keys is chosen by the first device to generate the definitive session key, the associated encrypted versions of the subset of session keys being communicated to the second device for decryption and processing.

4. A method as claimed in claim 2 or 3, in which the order of combination of a plurality of session keys used to generate the definitive session key is communicated from the first to the second device.

5. A method as claimed in claim 4 in which an initial session key value known to both the first and second devices is repeatedly encrypted in both devices by an ordered sequence of session keys using an encryption algorithm sensitive to the order of encryption.

6. A method as claimed in any preceding claim in which said at least one

precalculated key pair is selected from a larger set of precalculated key pairs prior to being stored in the first device.

7. A method as claimed in any preceding claim in which the encrypted version of a session key communicated to the second device also includes a signature value readable by the second device to verify the authenticity of the encrypted version of the session key.

8. A method as claimed in any preceding claim in which an algorithm and transport key used to encrypt and decrypt a session key correspond to a symmetric algorithm and associated symmetric key.

9. A method as claimed in any preceding claim in which an encryption algorithm used with a session key to encrypt and decrypt data communicated between the first and second device corresponds to a symmetric algorithm.

10. A method as claimed in any preceding claim, in which the first device is a decoder.

11. A method as claimed in any preceding claim, in which the second device is a portable security module.

12. A method as claimed in claim 11, in which the portable security module corresponds to one of a smart card and a conditional access module.

13. A method as claimed in any of claims 1 to 9, in which the first device corresponds to a conditional access module and the second device corresponds to a smart card.

14. A method as claimed in any of claims 10 to 13, in which data encrypted and decrypted with a session key corresponds to control word data.

15. A method as claimed in any of claims 10 to 13, in which data encrypted and decrypted with a session key corresponds to descrambled broadcast data.

16. A method as claimed in any of claims 1 to 9 in which the first and second device correspond to a first and second decoder respectively.

17. A method as claimed in any of claims 1 to 9 as applied to a home network system, the first and second devices corresponding to first and second consumer electronic devices adapted to transfer data via a communication link.

18. A first device adapted to be used in a method as claimed in any of claims 1 to 17, the first device including a memory in which at least one precalculated key pair is stored, said at least one precalculated key pair comprising a session key and an encrypted version of this session key.

19. A second device adapted to be used in a method as claimed in any of claims 1 to 18 and with a first device as claimed in claim 18, the second device comprising a memory in which is stored a key and algorithm that are needed to decrypt the encrypted session key value stored in the memory of the first device.

20. A first and second device as claimed in claims 18 and 19, in which the first device corresponds to a decoder and the second device to a portable security module.

21. A system for providing secure communication of data between first and second devices, said first device comprising a memory for storing at least one precalculated key pair comprising a session key and an encrypted version of the session key prepared using a transport key, and communication means for communicating the encrypted version of the session key to said second device, said second device comprising a memory for storing an equivalent transport key, decryption means for decrypting said encrypted version of the session key using said equivalent transport key, and means for encrypting data to be communicated to said first device using said session key.

22. A system as claimed in claim 21, wherein the memory of the first device is adapted to store a plurality of key pairs, the first device comprising means for selecting and processing at least one session key to generate a definitive session key said communication means being adapted to communicate the associated encrypted version of said at least one session key to the second device, said second device comprising means for processing said at least one session key to generate the definitive session key.

23. A system as claimed in claim 21 or 22, in which the encrypted version of a session key includes a signature value readable by the second device to verify the authenticity of the encrypted version of the session key.

24. A system as claimed in any of claims 21 to 23, in which an algorithm and transport key used to encrypt and decrypt a session key correspond to a symmetric algorithm and associated symmetric key.

25. A system as claimed in any of claims 21 to 24, in which an encryption algorithm used with a session key to encrypt and decrypt data communicated between the first and second device corresponds to a symmetric algorithm.

26. A system as claimed in any of claims 21 to 25, in which the first device is a decoder.

27. A system as claimed in any of claims 21 to 26, in which the second device is a portable security module.

28. A system as claimed in claim 27, in which the portable security module corresponds to one of a smart card and a conditional access module.

29. A system as claimed in any of claims 21 to 25, in which the first device corresponds to a conditional access module and the second device corresponds to a

smart card.

30. A system as claimed in any of claims 21 to 25 in which the first and second device correspond to a first and second decoder respectively.

31. A system as claimed in any of claims 21 to 25 as applied to a home network system, the first and second devices corresponding to first and second consumer electronic devices adapted to transfer data via a communication link.

32. A method of encryption of data communicated between a first and second device substantially as herein described.

33. A system for providing secure communication of data between first and second devices substantially as herein described.

FIG. 1

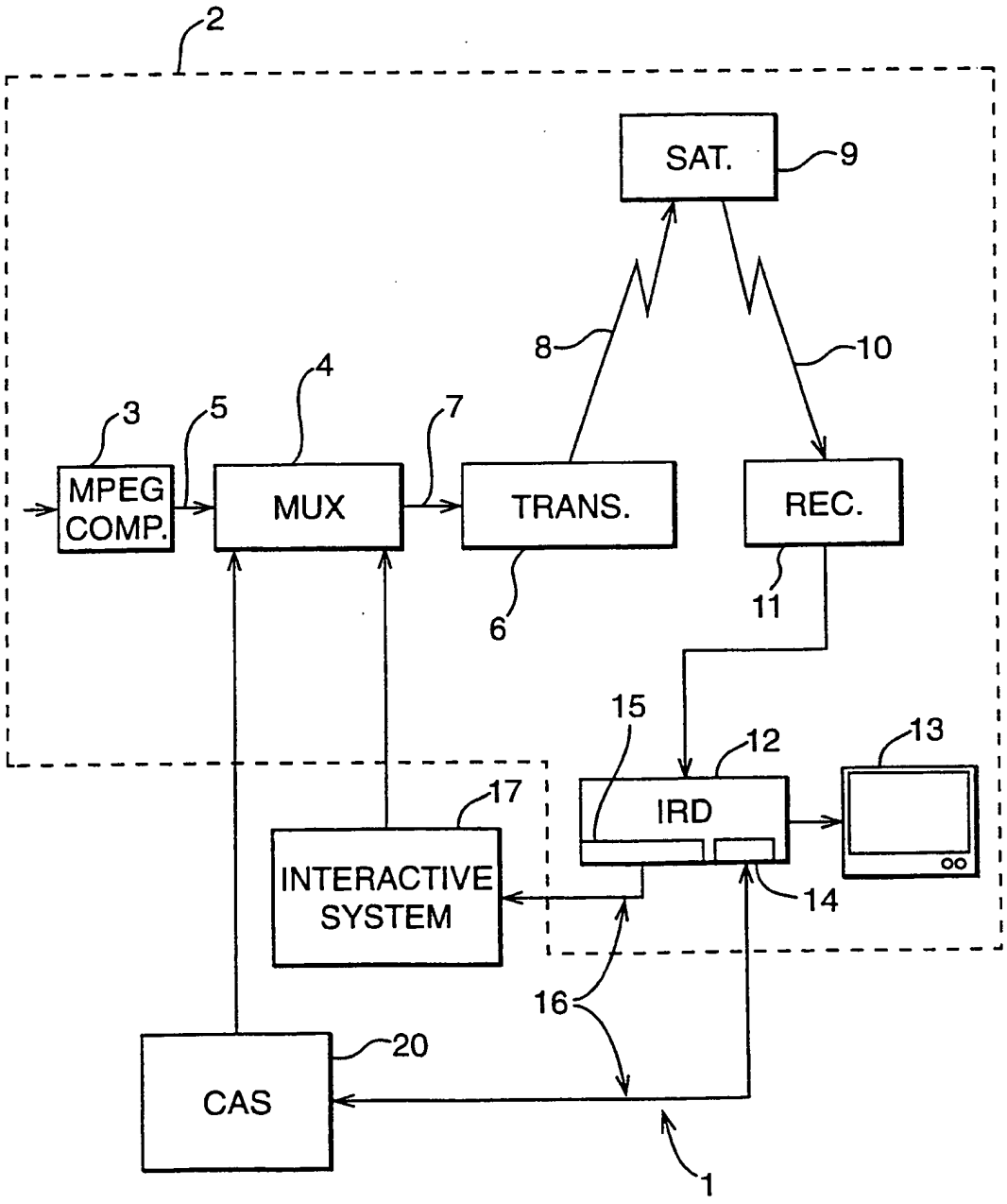

2
SAT.
9
8
10
4
3
5
7
MPEG COMP.
MUX
TRANS.
REC.
6
11
15
12
13
17
IRD
INTERACTIVE SYSTEM
14
16
20
CAS
1

FIG. 2

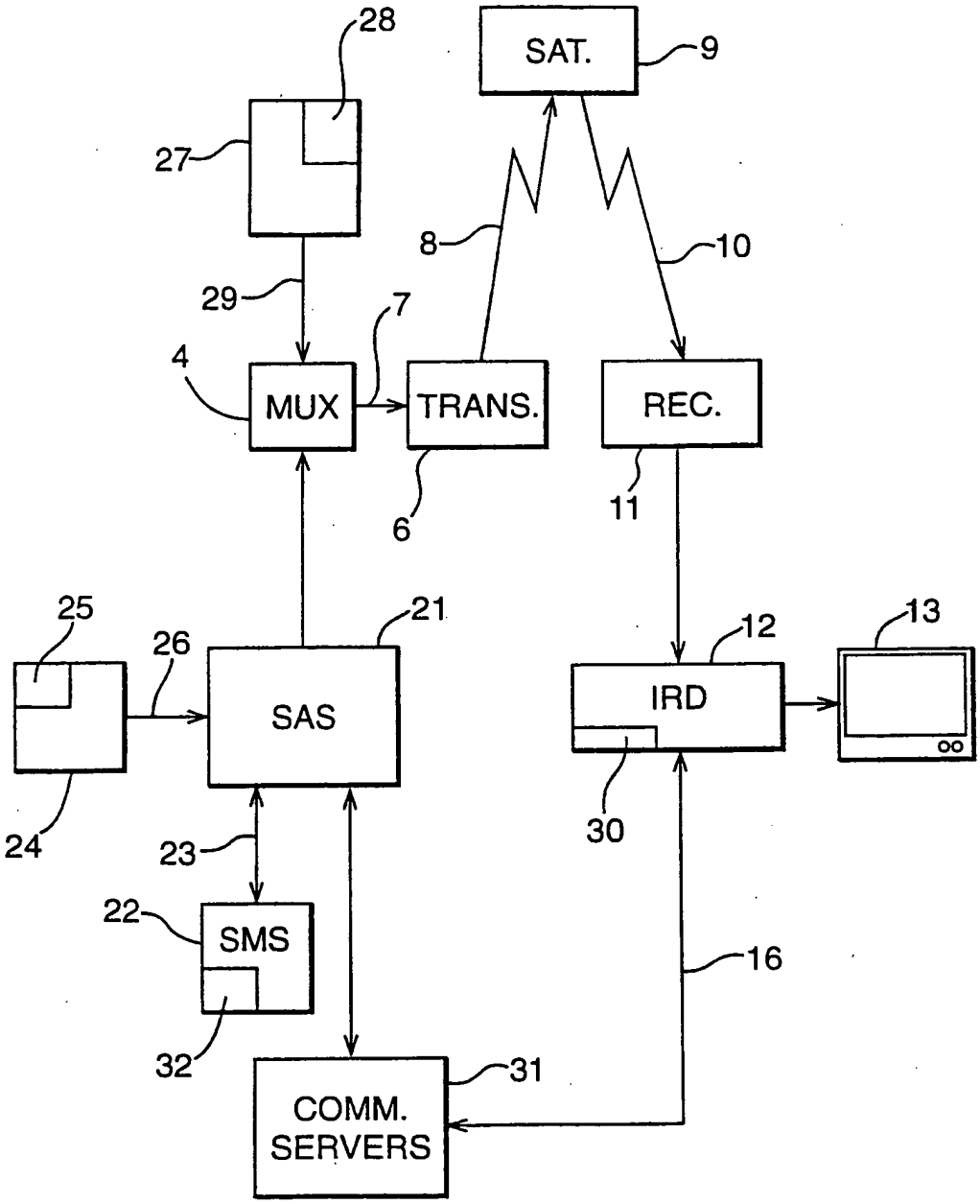
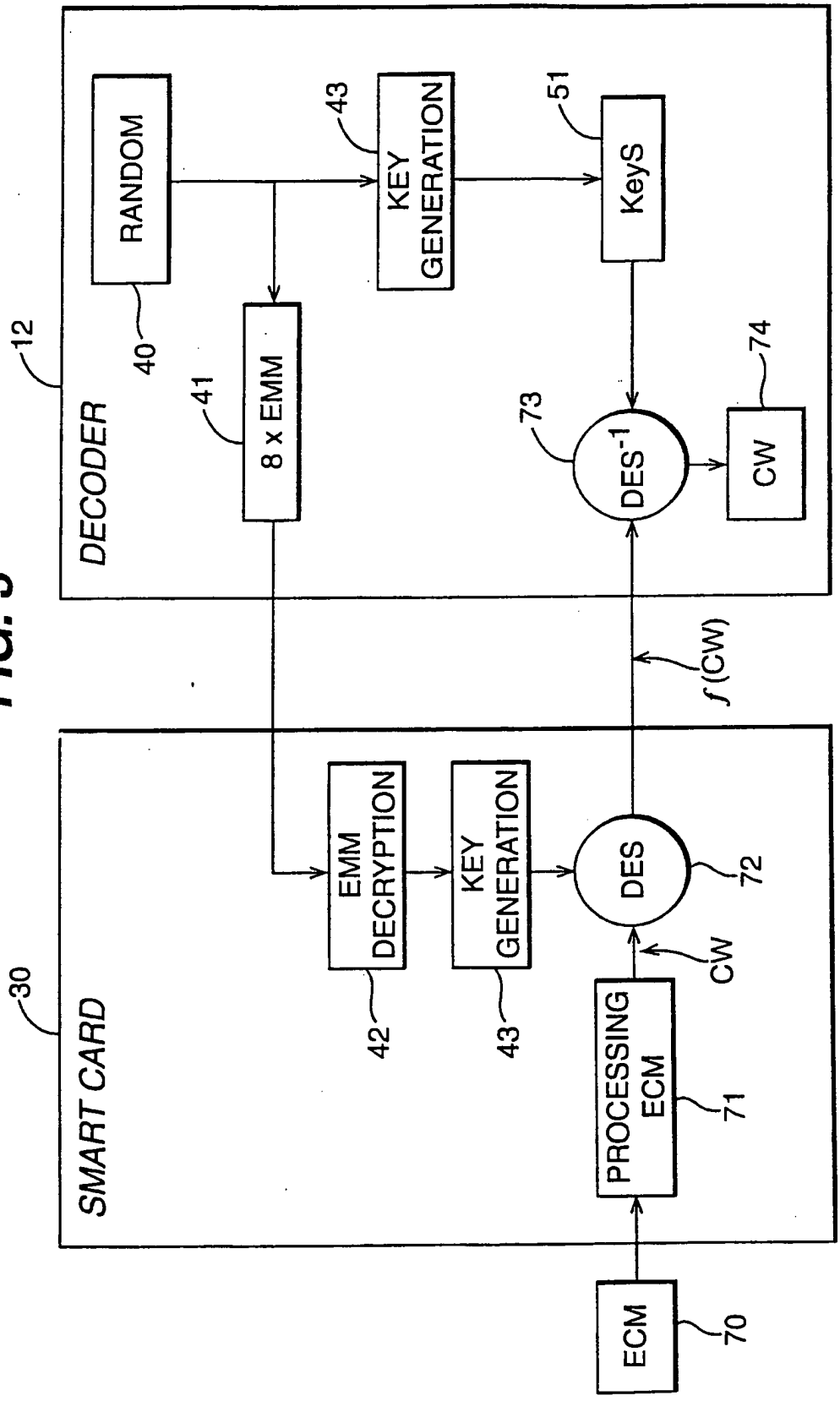

28
SAT.
9
27
8
10
29
7
4
MUX
TRANS.
REC.
6
11
25
26
21
12
13
SAS
IRD
24
23
30
22
SMS
16
32
31
COMM.
SERVERS

FIG. 3


DECODER
12
RANDOM
40
8 x EMM
41
KEY GENERATION
43
KeyS
51
DES-1
73
CW
74
f(CW)
SMART CARD
30
EMM DECRYPTION
42
KEY GENERATION
43
DES
72
CW
PROCESSING ECM
71
ECM
70

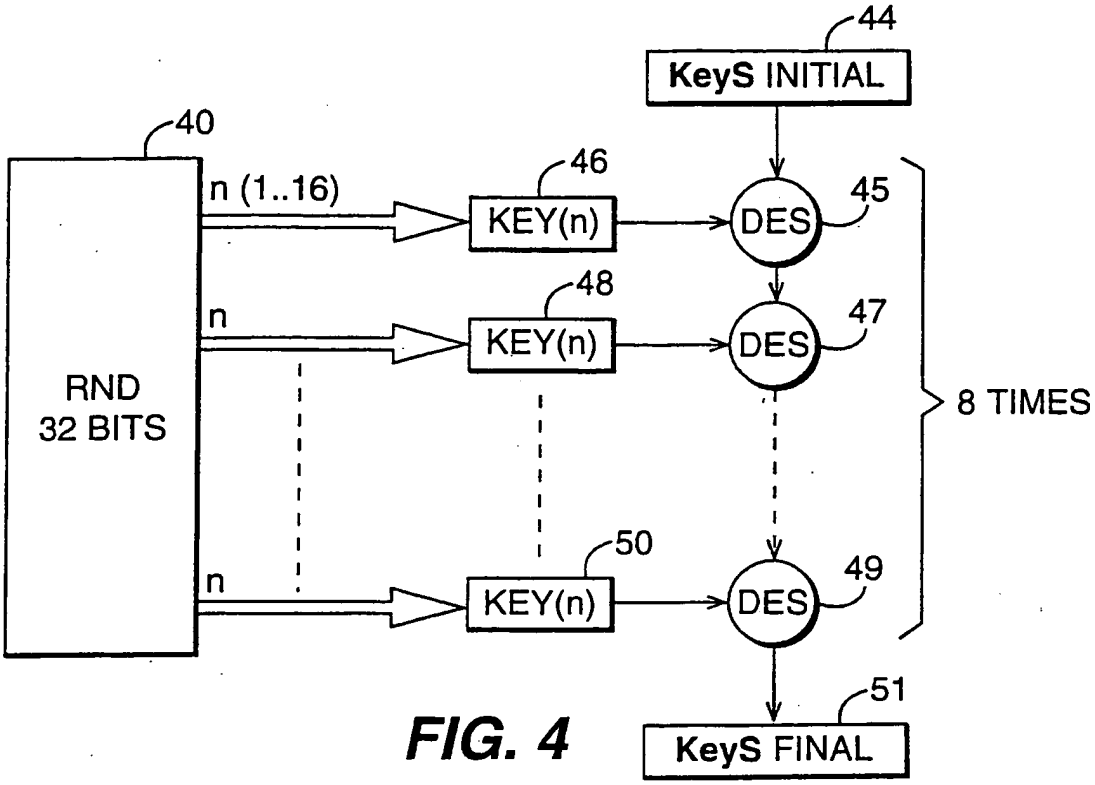

44
KeyS INITIAL
40
n (1..16)
46
KEY(n)
45
DES
48
n
KEY(n)
47
DES
RND
32 BITS
8 TIMES
n
50
KEY(n)
49
DES
51
KeyS FINAL

FIG. 4

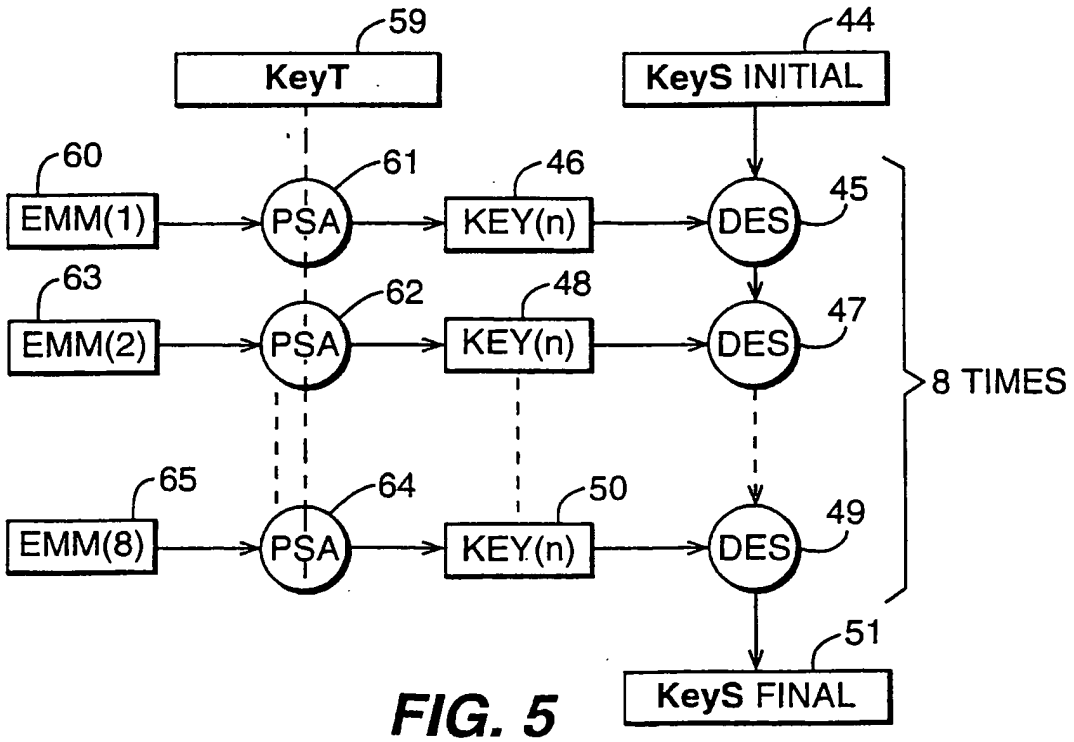

59
KeyT
44
KeyS INITIAL
60
61
46
EMM(1)
PSA
KEY(n)
45
DES
63
62
48
EMM(2)
PSA
KEY(n)
47
DES
8 TIMES
65
64
50
EMM(8)
PSA
KEY(n)
49
DES
51
KeyS FINAL

FIG. 5

INTERNATIONAL SEARCH REPORT

International Application No
PCT/IB 00/00163

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/16 H04N7/167

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 817 485 A (THOMSON MULTIMEDIA SA) 7 January 1998 (1998-01-07) page 3, column 3, line 54 -page 5, column 8, line 11 figures 1-5	1,2,4, 10-15, 17, 19-22, 26-29
X	EP 0 723 371 A (THOMSON MULTIMEDIA SA) 24 July 1996 (1996-07-24) page 3, column 3, line 57 -page 5, column 7, line 8 figures 1-4	1,2,4, 10-15, 19-22, 26-29

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
31 May 2000	07/06/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Van der Zaal, R

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/IB 00/00163

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EBU PROJECT GROUP B/CA: "FUNCTIONAL MODEL OF A CONDITIONAL ACCESS SYSTEM" EBU REVIEW- TECHNICAL, no. 266, 21 December 1995 (1995-12-21), pages 64-77, XP000559450 Grand Saconnex, CH page 64, left-hand column, line 1 -page 72, right-hand column, line 29 figures 1-8	1-33

1

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int ional Application No
PCT/IB 00/00163

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0817485 A	07-01-1998	FR 2750554 A	02-01-1998
		CN 1171015 A	21-01-1998
		JP 10164052 A	19-06-1998
		US 6035038 A	07-03-2000
EP 0723371 A	24-07-1996	FR 2729521 A	19-07-1996
		JP 8307850 A	22-11-1996

Search result: 1 of 1

(WO/2000/062260) METHOD AND SYSTEM FOR ORDERING, LOADING AND USING ACCESS TICKETS

Biblio. Data | Description | Claims | National Phase | Notices | Documents

Latest bibliographic data on file with the International Bureau

Publication Number: WO/2000/062260
Publication Date: 19.10.2000
Chapter 2 Demand Filed: 22.04.2000
International Application No.: PCT/CH1999/000142
International Filing Date: 07.04.1999

Int. Class.: ***G06Q 20/00*** (2006.01), ***G07B 15/00*** (2006.01), ***G07F 17/42*** (2006.01), ***G07F 7/00*** (2006.01), ***G07F 7/08*** (2006.01)

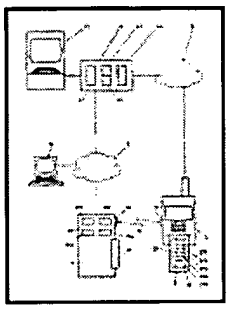
Applicants: **SWISSCOM MOBILE AG** [CH/CH]; Schwarztorstrasse 61 CH-3050 Bern (CH) *(All Except US)*.
RITTER, Rudolf [CH/CH]; Rossweidweg 8 CH-3052 Zollikofen (CH) *(US Only)*.
LAUPER, Eric [CH/CH]; Hochfeldstrasse 96 CH-3012 Bern (CH) *(US Only)*.

Inventors: **RITTER, Rudolf** [CH/CH]; Rossweidweg 8 CH-3052 Zollikofen (CH).
LAUPER, Eric [CH/CH]; Hochfeldstrasse 96 CH-3012 Bern (CH).

Agent: **BOVARD AG**; Optingenstrasse 16 CH-3000 Bern 25 (CH).

Title: METHOD AND SYSTEM FOR ORDERING, LOADING AND USING ACCESS TICKETS

Abstract: The invention relates to a method and a system for ordering, loading and using access tickets for the access to access-controlled service devices (3). Access tickets are ordered by a reservation centre (4) in said service device (3) by transmitting order information via an order channel. The order information comprises the telephone number of a mobile communications terminal (1). The ordered access tickets are transmitted to said terminal (1) via a mobile network (6) and are stored in a storage module (21) of the communications terminal (1). Data is exchanged between the storage module (21) and a reading device (31) of a service device (3) via a contactless interface (13). Decisions on the access permission for the user of said communications terminal (1) are made, e.g. in the reading device (31) or in the communications terminal (1), considering ticket information contained in said access ticket. Said information can be limited to a digitally signed ticket number or can contain data on the relevant service device. Access for the user to the service device (3) is given or denied according to the decision and by means of an access device (32) that is connected to the reading device.



Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZA, ZW.
African Regional Intellectual Property Org. (ARIPO) (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW)
Eurasian Patent Organization (EAPO) (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM)
European Patent Office (EPO) (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE)
African Intellectual Property Organization (OAPI) (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Publication Language: German (DE)

Filing Language: German (DE)

Capability-Based Computer Systems

1

Capability- and Object-Based System Concepts

Although the complexity of computer applications increases yearly, the underlying hardware architecture for applications has remained unchanged for decades. It is, therefore, not surprising that the demands of modern applications have exposed limitations in conventional architectures. For example, many conventional systems lack support in:

1. *Information sharing and communications.* An essential system function is the dynamic sharing and exchange of information, whether on a timesharing system or across a network. Fundamental to the sharing of storage is the addressing or naming of objects. Sharing is difficult on conventional systems because addressing is local to a single process. Sharing would be simplified if addresses could be transmitted between processes and used to access the shared data.
2. *Protection and security.* As information sharing becomes easier, users require access controls on their private data. It must also be possible to share information with, or run programs written by, other users without compromising confidential data. On conventional systems, all of a user's objects are accessible to any program which the user runs. Protection would be enhanced if a user could restrict access to only those objects a program requires for its execution.
3. *Reliable construction and maintenance of complex systems.* Conventional architectures support a single privileged mode of operation. This structure leads to monolithic design; any module needing protection must be part of the single operating system kernel. If, instead, any module could execute within a protected domain, systems could be built as a collection of independent modules extensible by any user.

Over the last several decades, computer industry and university scientists have been searching for alternative architectures that better support these essential functions. One alternative architectural structure is *capability-based* addressing. Capability-based systems support the *object-based* approach to computing.

This book explains the capability/object-based approach and its implications, and examines the features, advantages, and disadvantages of many existing designs. Each chapter presents details of one or more capability-based systems. Table 1-1 lists the systems described, where they were developed, and when they were designed or introduced.

System	Developer	Year	Attributes
Rice University Computer	Rice University	1959	segmented memory with "codeword" addressing
Burroughs B5000	Burroughs Corp.	1961	stack machine with descriptor addressing
Basic Language Machine	International Computers Ltd., U.K.	1964	high-level machine with codeword addressing
Dennis and Van Horn Supervisor	MIT	1966	conceptual design for capability supervisor
PDP-1 Time-sharing System	MIT	1967	capability supervisor
Multicomputer/ Magic Number Machine	University of Chicago Institute for Computer Research	1967	first capability hardware system design
CAL-TSS	U.C. Berkeley Computer Center	1968	capability operating system for CDC 6400
System 250	Plessey Corp., U.K.	1969	first industrial capability hardware and software system
CAP Computer	University of Cambridge, U.K.	1970	capability hardware with microcode support
Hydra	Carnegie-Mellon University	1971	object-based multi-processor O.S.
StarOS	Carnegie-Mellon University	1975	object-based multi-processor O.S.
System/38	IBM, Rochester, MN.	1978	first major commercial capability system, tagged capabilities
iAPX 432	Intel, Aloha, OR.	1981	highly-integrated object-based micro-processor system

Table 1-1: Major Descriptor and Capability Systems

Before surveying these systems at a detailed architectural level, it is useful to introduce the concepts of capabilities and object-based systems. This chapter defines the concept of capability, describes the use of capabilities in memory addressing and protection, introduces the object-based programming approach, and relates object-based systems to capability-based addressing.

Simplified examples of capability-based and conventional computer systems are presented throughout this chapter. These examples are meant to introduce the capability model by contrasting it with more traditional addressing mechanisms. In fact, many design choices are possible in both domains, and many conventional systems exhibit some of the properties of capability systems. No one of the following models is representative of all capability or conventional systems.

1.1 Capability-Based Systems

Capability-based systems differ significantly from conventional computer systems. Capabilities provide (1) a single mechanism to address both primary and secondary memory, and (2) a single mechanism to address both hardware and software resources. While solving many difficult problems in complex system design, capability systems introduce new challenges of their own.

Conceptually, a capability is a token, ticket, or key that gives the possessor permission to access an entity or object in a computer system. A capability is implemented as a data structure that contains two items of information: a *unique object identifier* and *access rights*, as shown in Figure 1-1.

The identifier *addresses* or *names* a single object in the computer system. An object, in this context, can be any logical or physical entity, such as a segment of memory, an array, a file, a

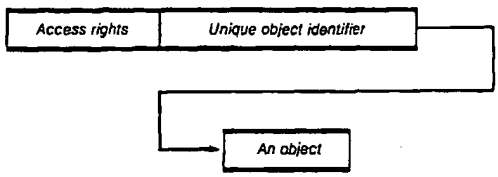


Figure 1-1: A Capability

line printer, or a message port. The access rights define the *operations* that can be performed on that object. For example, the access rights can permit read-only access to a memory segment or send-and-receive access to a message port.

Each user, program, or procedure in a capability system has access to a *list of capabilities*. These capabilities identify all of the objects which that user, program, or procedure is permitted to access. To specify an object, the user provides the *index* of a capability in the list. For example, to output a record to a file, the user might call the file system as follows:

```
PUT( file_capability , "this is a record" );
```

The capability specified in the call serves two purposes. First, it identifies the file to be written. Second, it indicates whether the operation to be performed (PUT in this case) is permitted.

A capability thus provides addressing and access rights to an object. Capabilities are the basis for object protection; a program cannot access an object unless its capability list contains a suitably privileged capability for the object. Therefore, the system must prohibit a program from directly modifying the bits in a capability. If a program could modify the bits in a capability, it could forge access to any object in the system by changing the identifier and access rights fields.

Capability system integrity is usually maintained by prohibiting direct program modification of the capability list. The capability list is modified only by the operating system or the hardware. However, programs can obtain new capabilities by executing operating system or hardware operations. For example, when a program calls an operating system routine to create a new file, the operating system stores a capability for that file in the program's capability list. A capability system also provides other capability operations. Examples include operations to:

1. Move capabilities to different locations in a capability list.
2. Delete a capability.
3. Restrict the rights in a capability, producing a less-privileged version.
4. Pass a capability as a parameter to a procedure.
5. Transmit a capability to another user in the system.

Thus, a program can execute direct control over the movement of capabilities and can share capabilities, and therefore, objects, with other programs and users.

It is possible for a user to have several capability lists. One list will generally be the master capability list containing capabilities for secondary lists, and so on. This structure is similar to a multi-level directory system, but, while directories address only files, capabilities address objects of many types.

1.1.1 Memory Addressing in Computer Systems

This section presents simplified models for both conventional and capability-based memory addressing systems. Although capabilities can control access to many object types, early capability-based systems concentrated on using capabilities for primary memory addressing. The first use of capabilities for memory protection was in the Chicago Magic Number Machine [Fabry 67, Yyngve 68], and an early description of capability-based memory protection appeared in Wilkes' book on timesharing systems [Wilkes 68]. Later, [Fabry 74] described the advantages of capabilities for generalized addressing and sharing.

For purposes of a simplified model, consider a conventional computer supporting a multiprogramming system in which each program executes within a single process. A program is divided into a collection of segments, where a segment is a contiguous section of memory that represents some logical entity, such as a procedure or array. A process defines a program's address space: that is, the memory segments it can access. The process also contains data structures that describe the user, and a directory that contains the names of a set of files. These files represent the user's long-term storage.

When a program is run, the operating system creates a process-local segment table that defines the memory segments available to the program. The segment table is a list of *descriptors* that contain physical information about each segment. Figure 1-2 shows example formats for a process virtual address and segment table descriptor. The operating system loads various segments needed by the program into primary memory, and loads the segment table descriptors with the physical address and length of each segment. A process can then access segments by reading from or writing to virtual addresses.

Each virtual address contains two fields: the segment number and the offset of a memory element within that specified segment. On each virtual address reference the hardware uses the segment number field as an index to locate an entry in the

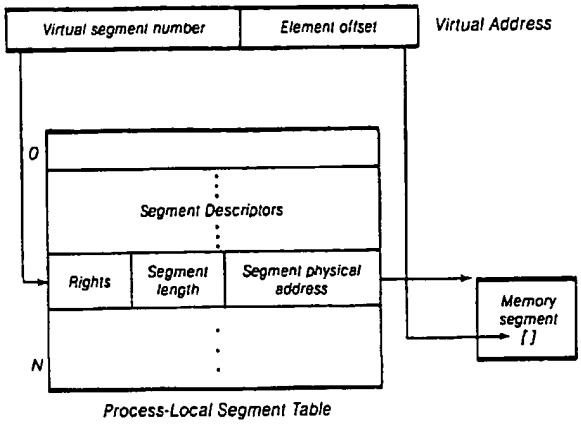


Figure 1-2: Conventional Segment Address Translation

process segment table. This descriptor contains the physical location of the segment. The length field in the descriptor is used to check that the offset in the virtual address is within the segment bounds. The rights field in the segment table entry indicates the type of access permitted to that segment (for example, read or write).

The model shown in Figure 1-2 has the following properties:

1. The system supports a segmented process virtual address space. A virtual address is local to the process and is translated through the process-local segment table.
2. A program can construct any virtual address and can attempt to read or write that address. On each reference, the hardware ensures that (a) the segment exists, (b) the offset is valid, and (c) the attempted operation is permitted. Otherwise, an error is signaled.
3. Loading of segment table entries is a privileged operation and can be accomplished only by the operating system. In general, a segment table is created at the time a program is loaded. The program then executes in a static addressing environment.
4. Sharing of segments between processes requires that the operating system arrange for both process-local segment tables to address the shared segments. If two processes wish to use the same virtual address to access a shared segment,

the segment descriptors must be in the same locations in both segment tables.

5. Any dynamic sharing of segments requires operating system intervention to load segment descriptors.

A *capability-based system* also supports the concept of a process that defines a program's execution environment. In the capability system, each process has a capability list that defines the segments it can access. Instead of the segment table descriptors available to the conventional system hardware, the capability addressing system consists of a set of *capability registers*. The *program* can execute hardware instructions to transfer capabilities between the capability list and the capability registers. The number of capability registers is generally small compared to the size of the capability list. Thus, at any time, the capability registers define a subset of the potentially accessible segments that can be physically addressed by the hardware. A simplified hardware model for this system is shown in Figure 1-3.

The model shown in Figure 1-3 has the following properties:

1. The system has a segmented virtual address space. A segment of memory can only be addressed by an instruction if a capability for that segment has been loaded into a capability register.

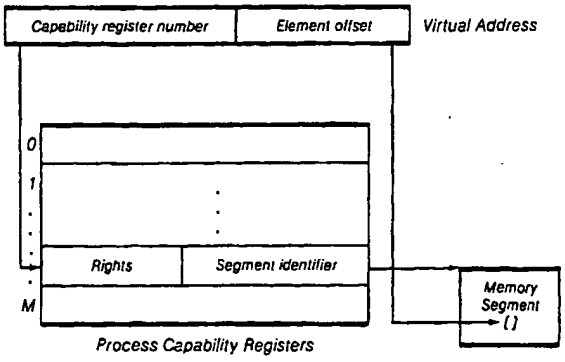


Figure 1-3: Capability Register Addressing

2. While loading of a segment descriptor in the conventional system is privileged, loading of a capability register is not. Instead of controlling the loading of the register, the capability system controls *the pattern of bits* that can be loaded. Only a valid capability can be loaded into a capability register.
3. The capability system provides a dynamically changing address space. The address space changes whenever the program changes one of the capability registers.
4. A virtual address identifies a process-local capability register. In this sense, a virtual address has similar properties to a virtual address in the conventional system. Sharing a virtual address does not in itself give access to the same segment.
5. A capability, however, is *not* process-local. Capabilities are *context independent*; that is, the segment addressed by a capability is independent of the process using that capability. A process can share a segment by copying or sending a capability from its capability list to the capability list of a cooperating process. Each of the processes can then access the segment.

One important difference between the conventional and capability approaches involves the ability of a program to affect system-wide or process-local objects. In the conventional system, a program executes within a virtual address space defined by a process. Every procedure called by that program has access to the process address space, including segments and files. Every procedure executes within an identical protection environment.

In the capability system, a procedure can only affect objects for which capability registers have been loaded. It is possible, therefore, for different procedures called by the same program to have access to different segments. Although all procedures may have the potential to load capability registers from the capability list, some procedures may choose to execute within a very small addressing sphere.

The ability to restrict the execution or addressing environment of a procedure has several benefits. First, if a procedure is allowed access only to those segments absolutely needed, the hardware can detect any erroneous references. For example, a reference past the end of an array might be caught before it destroys another variable. Second, if a procedure is found to be in error, it is easy to determine what segments might have been affected. If the segments that could have been modified were local to the procedure, recovery might be substantially easier.

Most capability systems go a step further by allowing each procedure to have a *private* capability list. A procedure can

thus protect its objects from accidental or malicious access by its callers, and a program can protect its objects from access by called procedures. Every procedure can have, in effect, its own address space. To permit a procedure access to a local object, a program can pass a capability for the object as a parameter when the procedure is called. Therefore, in a capability system, every procedure can be protected from every other procedure because each has a private capability list. When one procedure calls another, it knows that the called procedure can access only local objects for which capabilities are passed.

1.1.2 The Context of an Address

Each object in a capability system has a unique identifier. Conceptually, each object's identifier is unique for all time. That is, an identifier is assigned when an object is created and that identifier is never reused, even after the object is deleted. During the object's lifetime, its unique identifier is used within capabilities to specify the object. An attempt to use a capability with an identifier for a deleted object causes an error.

In practice, the object identifier field of a capability must be used by hardware to locate the object. From the hardware viewpoint, the identifier is an address—either the address of a segment or perhaps the address of a central descriptor that contains physical information about the segment. The need to handle addresses efficiently in hardware typically causes addresses to be small—16 or 32 bits, for example. For this reason, identifiers tend to have too few bits to be unique for all time. However, the choice of the number of bits in an identifier is an important system design decision that dictates the way in which capabilities can be used.

In conventional systems, an address is meaningful only within a single process. In a capability system, addresses (capabilities and their identifiers) are context-independent. That is, the interpretation of a capability is independent of the process using it. The unique identifier within a capability must have a system-wide interpretation. Unique identifiers must be large enough to address all of the segments likely to be in use by all executing processes at any time. This allows capabilities to be freely passed between processes and used to access shared data.

Addressing on most conventional systems is restricted in terms of time as well as context. An address is meaningful only within the lifetime of a single process. Therefore, addresses cannot be used to name objects whose lifetimes are greater than

the process creating the objects. If a process wishes to create a long-term storage object, such as a file, it must interface to the file system. Files typically require different naming, protection, and storage mechanisms than memory segments.

A significant advance made possible by capabilities is the naming and protection of both long-term and short-term objects with a single mechanism. If the identifier field is very large, it may be possible to implement identifiers unique for all time. Each object is addressed by capabilities containing its unique identifier, independent of whether it is stored in primary or secondary memory. The operating system or hardware can maintain data structures that indicate the location of each object. If a program attempts to access an object in secondary memory, the hardware or operating system can bring the object into primary memory so that the operation can be completed. From the program's point of view, however, there is a single-level address space. Capabilities, as well as data, can be saved for long periods of time and stored in secondary memory.

There are, therefore, several contexts in which an address can have meaning. For example, for:

1. Primary memory segments of a single process.
2. Primary memory segments of all existing processes.
3. All existing segments in both primary and secondary memory.

Most conventional systems support only type 1, while capabilities allow for any of the listed addressing types. More importantly, while conventional systems are concerned only with the protection of *data*, capability systems are concerned also with the protection of *addresses*. A process on a capability system cannot fabricate new addresses. As systems become more general in their addressing structure as in types 2 and 3, the protection of addresses becomes crucial to the integrity of the system.

1.1.3 Protection in Computer Systems

Lampson contrasts the capability approach with the traditional approach by showing the structure of protection information needed in a traditional operating system [Lampson 71]. Figure 1-4 depicts an access matrix showing the privileges that each system user is permitted with respect to each system object. For example, user Fred has read and write privileges to File1 and no privileges to File2, while user Sandy is allowed to read both files.

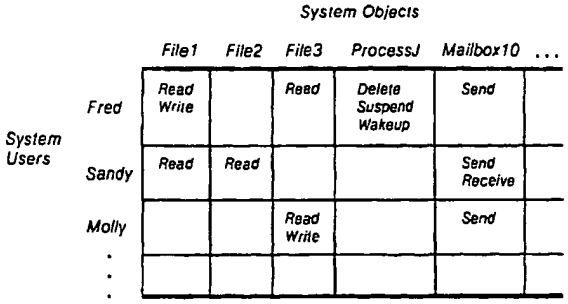


Figure 1-4: System Object Access Matrix

One conventional approach to the maintenance of protection information is *access control lists*, in which the operating system keeps an *access list* for each object in the system. Each object's list contains the names of users permitted access to the object and the privileges they may exercise. When a user attempts to access an object, the operating system checks the access list associated with that object to see if the operation is authorized. Each of the columns of Figure 1-4 represents an access control list.

The capability system offers an alternative structure in which the operating system arranges protection information by user instead of by object. A *capability list* is associated with each user in the system. Each capability contains the name of an object in the system and the user's permitted privileges for accessing the object. To access an object, the user specifies a capability in the local capability list. Each of the rows of Figure 1-4 represents a capability list. Figure 1-5 shows an access list

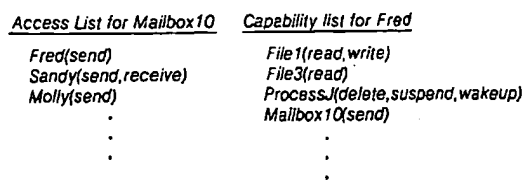


Figure 1-5: Access Control and Capability Lists

and a capability list derived from the protection matrix in Figure 1-4.

One important difference between the capability list and access list is the user's ability to *name* objects. In the access list approach, a user can attempt to name any object in the system as the target of an operation. The system then checks that object's access list. In the capability system, however, a user can only name those objects for which a capability is held: that is, to which some access is permitted.

In either case, the integrity of the system is only as good as the integrity of the data structures used to maintain the protection information. Both access control list and capability list mechanisms must be carefully controlled so that users cannot gain unauthorized access to an object.

Similar protection options exist outside the computer world. A useful analogy is the control of a safe deposit box. Suppose, for example, that Carla wishes to keep all of her valuables in a safe deposit box in the bank. On occasion, she would like one or more trustworthy friends to make deposits or withdrawals. There are basically two ways that the bank can control access to the box. First, the bank can maintain a list of people authorized to access the box. To make a transaction, Carla or any of her friends must prove their identity to the bank's satisfaction. The bank checks the (access control) list for Carla's safe deposit box and allows the transaction if the person is authorized. Or, instead of maintaining a list, the bank can issue Carla one or more keys to her safe deposit box. If Carla needs to have a friend access the box, she simply gives a key to the friend.

A number of observations can be made about these two alternative protection systems. The properties of the access list scheme are:

1. The bank must maintain a list for each safe deposit box.
2. The bank must ensure the validity of the list at all times (e.g., it cannot allow the night watchman to add a name).
3. The bank must be able to verify the identity of those asking to use a box.
4. To allow a new person to use the box, the owner must visit the bank, verify that he or she is the owner of the box, and have the new name added to the list.
5. A friend cannot extend his or her privilege to someone else.
6. If a friend becomes untrustworthy, the owner can visit the bank and have that person's name removed from the list.

The alternative scheme involving keys has the following properties:

1. The bank need not be involved in any transactions once the keys are given, except to allow a valid keyholder into the vault.
2. The physical lock and key system must be relatively secure; that is, it must be extremely difficult to forge a key or to pick the lock on a safe deposit box.
3. The owner of a box can simply pass a key to anyone who needs to access the box.
4. Once a key has been passed to a friend, it is difficult to keep them from giving the key to someone else.
5. Once a friend has made a transaction, the owner can ask for the key back, although it may not be possible to know whether or not the friend has made a copy.

The advantage of the key-based system is ease of use for both the bank and customer. However, if today's friends are likely to become tomorrow's enemies, the access list has the advantage of simple guaranteed access removal. Of course, the access control list and the key (or capability) systems are not mutually exclusive, and can be combined in either the computer or banking world to provide the advantages of both systems for increased protection.

1.2 The Object-Based Approach

Over the last few decades, several areas of computer science have converged on a single approach to system design. This approach, known as *object-based computing*, seeks to raise the level of abstraction in system design. The events that have encouraged object-based design include:

1. Advances in computer architecture, including capability systems and hardware support for operating systems concepts.
2. Advances in programming languages, as demonstrated in Simula [Dahl 66], Pascal [Jensen 75], Smalltalk [Ingalls 78], CLU [Liskov 77], and Ada [DOD 80].
3. Advances in programming methodology, including modularization and information hiding [Parnas 72] and monitors [Hoare 74].

This section introduces the object approach and discusses its relationship to capability-based computer systems.

What is object-based computing? Simply stated, the object approach is a method of structuring systems that supports *ab-*

straction. It is a philosophy of system design that decomposes a problem into (1) a set of *abstract object types*, or resources in the system, and (2) a set of *operations* that manipulate *instances* of each object type.

To make this idea more concrete, consider the following simplified example. Imagine that we are programming a traffic simulation for a city. First, define a set of objects that represent, abstractly, the fundamental entities that make up the traffic system. Some of the object *types* for the traffic simulation might be:

- passenger
- bus
- bus stop
- taxi
- car

Then, for each object type, define the operations that can be performed. Bus objects, for example, might support the operations:

- PUT_BUS_INTO_SERVICE(bus_number)
- MOVE_BUS(bus_number, bus_stop)
- LOAD_PASSENGERS(bus_number, passenger_list)
- UNLOAD_PASSENGERS(bus_number, passenger_list)
- GET_PASSENGER_COUNT(bus_number)
- GET_POSITION(bus_number)
- REMOVE_BUS_FROM_SERVICE(bus_number)

Each bus operation accepts a bus number as a parameter. At any time there may be many bus objects in the system, and we identify each bus by a unique number. Each of these bus objects is an *instance* of the *type* bus. The *type* of an object identifies it as a member of a class of objects that share some behavioral properties, such as the set of operations that can be performed on them.

What has been gained by defining the system in this way? First, there now exist a fundamental set of objects and operations for the simulation. We can now implement the procedures to perform the operations on each type of object. Since only a limited number of procedures operate on each object type, access to the internal data structures used to maintain the state of each type can be restricted. This isolation of the knowledge of those data structures should simplify any future

changes to one of the object abstractions because only a limited set of procedures is affected.

Second, and more importantly, we have raised the *level of abstraction* in the simulation program. That is, we can now program the simulation using buses, passengers, and bus stops as the fundamental objects, instead of bits, bytes, and words, which are normally provided by the underlying hardware. The buses and passengers are our data types just as bits and bytes are the data types supported in hardware. The simulation program will consist mainly of control structures plus procedure calls to perform operations on instances of our fundamental objects.

Of course, in this example, the procedures implementing the operations are programmed using lower-level objects, such as bytes, words, and so on. Or, they may be further decomposed into simpler abstract objects that are then implemented at a low level. Object-based systems provide a fundamental set of objects that can be used for computing. From this basis, the programmer constructs new higher-level object types using combinations of the fundamental objects. In this way the system is extended to provide new features by creating more sophisticated abstractions.

This methodology aims to increase productivity, improve reliability, and ease system modification. Through the use of well-defined and well-controlled object interfaces, systems designers hope to simplify the construction of complex computer systems.

1.2.1 Capabilities and Object-Based Systems

In the simulation example, each object is identified by a unique number. To move a bus from one stop to another, we call the MOVE_BUS operation with the unique number of the bus to move. For purposes of the simple simulation, a small set of integers suffices to identify the buses or other objects. No protection is needed because these objects are implemented and used by a single program.

The use of the object approach to build operating system facilities presents different requirements. For example, suppose we wish to build a calendar system to keep track of scheduled meetings, deadlines, reminders, and so on. The fundamental object of the calendar system, from the user's point of view, is a calendar object. Our calendar management system provides routines that create a new calendar, and modify,

query, or display an existing calendar. Many users in the system will, of course, want to use this facility.

Several familiar issues now arise: (1) how does a user name a calendar object, (2) how is that calendar protected from access by other users, and (3) how can calendars be shared under controlled circumstances? Only the owner of a calendar should be able to make changes, and the annotations in each calendar must be protected from other users, since they might contain confidential information. However, a user might permit selected other users to check if he or she is busy during a certain time, in order to automate the scheduling of meetings.

Capabilities provide a solution to these problems. When a user creates a new calendar, the calendar creation routine allocates a segment of memory for which it receives a capability. This segment is used to store data structures that will hold the calendar's state. The create routine uses this capability to initialize the data structures, and then returns it to the caller as proof of ownership of the calendar. In order to later modify or query the calendar, the user specifies the returned capability; the capability identifies the calendar and allows the modify or query procedure to gain access to the data structures. Only a user with a valid capability can access a calendar.

A weakness with this scenario is that the calendar system cannot prevent the calendar owner from using its capability to access the data structures directly. The calendar system would like to protect its data structures both to ensure consistency and to guarantee that future changes in data format are invisible outside of the subsystem. In addition, if a user passes a calendar capability to another user, the second user can then modify the data structures or read confidential information.

These problems exist because the calendar system returns a fully-privileged calendar capability to the user. Instead, what is needed is a capability that identifies a specific calendar and is proof of ownership, but does not allow direct access to the underlying data structures. In other words, the calendar system would like to return only *restricted* capabilities to its clients. However, the calendar system must retain the ability to later *amplify* the privileges in one of its restricted capabilities so that it can access the data structures for a calendar.

There are several ways of providing type managers with this special ability. (These mechanisms are examined in detail throughout the book.) However, given this power over capabilities for its objects, a type manager can ensure that its clients operate only through the well-defined object operation interface. A client can pass a capability parameter to the type man-

ager when requesting a service, but cannot otherwise use the capability to read or write the object it addresses. This facility is fundamental to any system that allows creation and protection of new system types.

1.3 Summary

The capability concept can be applied in hardware and software to many problems in computer system design. Capabilities provide a different way of thinking about addressing, protection, and sharing of objects. Some of the properties of capabilities illustrated in this chapter include their use in:

1. Addressing primary memory in a computer system.
2. Sharing objects.
3. Providing a uniform means of addressing short- and long-term storage.
4. Support for a dynamic addressing environment.
5. Support for data abstraction and information hiding.

These, of course, are advantages of capability-based systems. The most important advantage is support for object-based programming. Object-based programming methodology seeks to simplify the design, implementation, debugging, and maintenance of sophisticated applications. While capabilities solve a number of system problems, their use raises a whole new set of concerns. And, as is often the case in computer system design, the concept is much simpler than the implementation.

The remainder of this book is devoted to examining many different capability-based and object-based designs. The characteristics of each system are described with emphasis on addressing, protection, and object management. Each system represents a different set of tradeoffs and presents different advantages and disadvantages. When comparing the systems, consider the differences in goals, technologies, and resources available to the system developers.

The final chapter of this book considers issues in capability system design common to all of the systems described. A few of the questions to be considered follow. It may be useful to remember these questions when examining each system design.

1. What is the structure of an address?
2. How is a capability represented? How is a capability used to locate an object?

3. How are capabilities protected?
4. What is the lifetime of a capability?
5. What types of objects are supported by the hardware and software?
6. What is the lifetime of an object?
7. How can users extend the primitive set of objects provided by the base hardware and software?

1.4 For Further Reading

The concept of capability is formally defined in the 1966 paper by Dennis and Van Horn [Dennis 66]. Chapter 3 examines this paper in some detail. The paper by Fabry [Fabry 74] compares capability addressing and conventional segmented addressing of primary memory, while Redell [Redell 74a] describes issues in capability systems and the use of sealing mechanisms that support the addition of new object types to a system. These papers are a fundamental part of capability literature.

Capability systems have been discussed in various contexts. Two papers by Lampson [Lampson 69 and Lampson 71] describe the requirements for protection in operating systems and the capability protection model. The surveys by Linden [Linden 76] and Denning [Denning 76], which appeared in a special issue of *ACM Computing Surveys*, describe capability systems and their relationship to security and fault tolerance in operating systems.

The architecture books by Myers [Myers 82] and Iliffe [Iliffe 82] also discuss some of the systems described in this book. Myers' book contains details of Sward [Myers 80], a capability-based research system built at IBM that is omitted here. A capability system model, as well as discussion of some existing capability systems, appears in the book by Gehringer [Gehringer 82]. Jones [Jones 78a] provides a good introduction to the concepts of object-based programming.

The Burroughs B5000 computer. (Courtesy Burroughs Corporation.)

print version

A3

Wired News

Search:

Wired News Search

[an error occurred while processing this directive][an error occurred while processing this directive][an error occurred while processing this directive]

Issue 2.09 - Sep 1994

Superdistribution

By Brad Cox

It has become a maxim: intangible electronic goods - software applications, online magazine stories, clip art - are quite distinct from tangible goods like baskets, potatoes, and oil refinery machinery. Tangible goods are made of atoms; electronic goods are made of bits. While those who produce electronic goods must expend the same capital, labor, and knowledge as those producing tangible goods, their products can be copied in nanoseconds and transported at the speed of light.

The hard-to-copy nature of tangible goods made the traditional pay-per-copy mechanism a natural choice. But an info product's ease of duplication so thoroughly undercuts the traditional notion of pay-per-copy that the possibility of an abundant supply of pre-fabricated information-age goods is nearly nixed.

But imagine a significantly altered market mechanism for electronic goods. Instead of treating ease-of-replication as a liability to be prevented - via labor-intensive copy protection and legal or moral restrictions - this new model treats ease-of-replication as the asset upon which a new foundation for software engineering could be based. In Japan this new way is called superdistribution. Superdistribution lets information flow freely, without resistance. Eschewing the low-tech property-rights mechanisms already widespread (shrinkwrap software, license servers, dongles, demoware, shareware), superdistribution allows miners, refiners, fabricators, assemblers, distributors, and marketers to cooperate and compete as producers and consumers of electronic goods within a global information-age society.

Existing copyright law distinguishes between copyright (the right to copy or distribute) and useright (the right to "perform," or to use a copy once obtained). In the eyes of the law, when Joe Sixpack buys a record or CD at a store, he's actually purchasing a *bundle* of rights that includes ownership of a physical medium along with a limited useright that allows use of the music on that medium only for personal enjoyment. Large television and radio companies buy an entirely different bundle of rights. They often have the same media (whose only difference is a "not for resale" sticker on the cover) thrust upon them for free by publishing companies in expectation of substantial fees for the useright to play the music on the air. These fees are administered by ASCAP (American Society of Composers, Authors and Publishers) and BMI (Broadcast Musicians Institute), who monitor how often each record is broadcast to how large a listening audience.

Similarly, superdistribution treats each personal computer as a broadcasting station whose "audience"

consists of a single "listener." First pioneered in 1987 by Ryoichi Mori, head of the Japan Electronics Industry Development Association, superdistribution is based on the observation that electronic objects are fundamentally unable to monitor their own copying but trivially able to monitor their use. For example, making software - whether it's Microsoft's Word or Mike's string-compare subroutine - count how many times it has been invoked is easy, but making it count how many times it has been copied is much more difficult. So why not build an information-age market economy around this difference? If revenue collection were based on monitoring the use of software inside a computer, vendors could dispense with copy protection altogether. They could distribute electronic objects for free in expectation of a usage-based revenue stream. (This, of course, raises the same hairy privacy issues that we trade off when we choose to use credit cards instead of cash or talk by telephone rather than face to face. The real risk to privacy here does not arise when usage information is used only for billing, but from any possibility that it might be used for other purposes.)

Treating ease-of-replication as an *asset* rather than a *liability*, superdistribution actively encourages free distribution of information-age goods via any distribution mechanism imaginable. It invites users to download superdistribution software from networks, to give it away to their friends, or to send it as junk mail to people they've never met.

Why this generosity? Because the software is actually "meterware." It has strings attached, whose effect is to decouple revenue collection from the way the software was distributed. Superdistribution software contains embedded instructions that make it useless except on machines that are equipped for this new kind of revenue collection.

Superdistribution-equipped computers are otherwise quite ordinary. They run ordinary pay-by-copy software just fine, but they have additional capabilities that only superdistribution software uses. In Mori's prototype, these extra services are provided by a silicon chip that plugs into a Macintosh coprocessor slot. The hardware is surprisingly uncomplicated (its main complexities being tamper-proofing, not base functionality), and far less complicated than hardware that the computer industry has been routinely building for decades. Electronic objects intended for superdistribution invoke this hardware, which provides instructions. These instructions check that revenue-collection hardware is present, prior usage reports have been uploaded, and prior usage fees have been paid. They also keep track of how many times they have been invoked, storing the resulting usage information temporarily in a tamper-proof persistent RAM. Periodically (say monthly), this usage information is uploaded to an administrative organization for billing, using encryption technology to discourage tampering and to protect the secrecy of the metered information. (Think of your utility bill.)

Software users receive monthly bills for use of each top-level component - say, Microsoft Excel, *Myst*, or a Net-based rock video. When these bills are paid, payments are divvied up between the makers of the component and makers of subcomponents - in proportion to usage. For example, for the rock video, payment might go to MTV as well as to the artist. In other words, the end-user's payments are recursively distributed through the producer-consumer hierarchy. The distribution is governed by usage metering information collected from each end-user's machine, plus usage pricing data provided to the administrative organization by each component vendor. (The various rounds of payment resemble those made by Visa or MasterCard.)

Since communication is infrequent and involves only a small amount of metering information, the communication channel could be as simple as a modem that autodials a hardwired 800 number each month. Many other solutions are viable, such as flash cards or even floppy disks to be mailed back and forth each month.

Consider an author who wishes to distribute or sell a multimedia document that cannot be handled as a simple text file. Without superdistribution, the author's market is confined to those who have already purchased a program capable of displaying this document - a run-time version of Macromedia Director, for example. The same occurs at each lower level of the producer/consumer hierarchy. The market of a programmer who wishes to sell a reusable software component is restricted to those who have already purchased the components and tools upon which the software component relies.

With superdistribution, the market is no longer restricted to those who own Director, because it will be acquired by the customers' operating system as if it were a part of the document. The creator of the document accrues revenue from those who read it, as does the creator of Director.

The user's operating system acquires subcomponents of the document, such as Director and any subcomponents it relies on (QuickDraw, etc.), from the hard drive's cache, automatically loading it as needed from the network. The operating system can do this automatically and transparently because loading software involves no financial commitments when revenue is based on usage instead of acquisition of copies.

Superdistribution addresses the perennial, implicit questions of those who might potentially provide the smaller-granularity reusable software components upon which an advanced software engineering culture could be founded: Where do software components come from? Why should I bother to provide them? Why should I engage in such gritty activities as testing and documenting reusable software components sufficiently so that others might use them? What is in it for me?

Where software's ease-of-replication is a liability today (by disincentivizing those who would provide it), superdistribution turns this liability into an asset by allowing goods to be distributed for free. Where software vendors must now spend heavily to overcome software's invisibility, superdistribution would thrust software out into the world to serve as its own advertisement. Where the personal computer revolution isolates individuals inside a stand-alone personal computer, superdistribution establishes a cooperative/competitive community around an information-age market economy.

By separating revenue collection from acquisition of copies, hard drives and computers can disappear and become just part of the plumbing that conveys information-age goods between producers and consumers. Computers and telecommunications links become invisible, a transparent window through which individuals can communicate, cooperate, coordinate, and compete as members of an advanced socioeconomic community.

Brad Cox (bcox@gmu.edu) is founder of the Coalition for Electronic Markets and a faculty member in George Mason University's Program on Social and Organizational Learning.

Atomic Proxy Cryptography

Matt Blaze Martin Strauss
AT&T Labs – Research
Florham Park, NJ 07932
{mab,mstrauss}@research.att.com

DRAFT – 2 November 1997 – DO NOT RE-DISTRIBUTE*

Abstract

This paper introduces *atomic proxy cryptography*, in which an *atomic proxy function*, in conjunction with a public *proxy key*, converts ciphertext (messages in a public key encryption scheme or signatures in a digital signature scheme) for one key (k_1) into ciphertext for another (k_2). Proxy keys, once generated, may be made public and proxy functions applied in untrusted environments. Various kinds of proxy functions might exist; *symmetric* atomic proxy functions assume that the holder of k_2 unconditionally trusts the holder of k_1, while *asymmetric* proxy functions do not. It is not clear whether proxy functions exist for previous public-key cryptosystems. Several new public-key cryptosystems with symmetric proxy functions are described: an encryption scheme, which is at least as secure as Diffie-Hellman, an identification scheme, which is at least as secure as the discrete log, and a signature scheme derived from the identification scheme via a hash function.

1 Introduction

1.1 Overview

A basic goal of public-key encryption is to allow only the key or keys selected at the time of encryption to decrypt the ciphertext. To change the ciphertext to a different key requires re-encryption of the message with the new key, which implies access to the original cleartext and to a reliable copy of the new encryption key. Intuitively, this seems a fundamental, and quite desirable, property of good cryptography; it should not be possible for an untrusted party to change the key with which a message can be decrypted.

This paper, on the other hand, investigates the possibility of *atomic proxy functions* that convert ciphertext for one key into ciphertext for another without revealing secret decryption keys or cleartext messages. An atomic proxy function allows an untrusted party to convert ciphertext between keys without access to either the original message or to the secret component of the old key or the new key. In proxy cryptography, the holders of public-key pairs A and B create and publish a *proxy key* $\pi_{A\to B}$ such that $D(\Pi(E(m,e_A),\pi_{A\to B}),d_B) = m$, where $E(m,e)$ is the public encryption function of message m under encryption key e, $D(c,d)$ is the decryption function of ciphertext c under decryption key d, $\Pi(c,\pi)$ is the atomic proxy function that converts ciphertext

*Current draft available at ftp://ftp.research.att.com/dist/mab/proxy.ps

c according to proxy key π, and e_A, e_B, d_A, d_B are the public encryption and secret decryption component keys for key pairs A and B, respectively. The proxy key gives the owner of B the ability to decrypt "on behalf of" A; B can act as A's "proxy." In other words, the Π function effectively allows the "atomic" computation of $E(D(c, d_A), e_B)$ without revealing the intermediate result $D(c, d_A)$.

We consider atomic proxy schemes for encryption, identification and signatures. An encryption proxy key $\pi_{A\to B}$ allows B to decrypt messages encrypted for A and an identification or signature proxy key $\pi_{A\to B}$ allows A to identify herself as B or to sign for B (i.e., transforms A's signature into B's signature). Generating encryption proxy key $\pi_{A\to B}$ obviously requires knowledge of at least the secret component of A (otherwise the underlying system is not secure) and similarly generating identification or signature proxy key $\pi_{A\to B}$ requires B's secret, but the proxy key itself, once generated, can be published safely.

1.2 Categories of proxy schemes

Encryption proxy functions (and similarly but contravariantly, identification or signature proxy functions) can be categorized according to the degree of trust they imply between the two key holders. Clearly, A must (unconditionally) trust B, since the encryption proxy function by definition allows B to decrypt on behalf of A. *Symmetric* proxy functions also imply that B trusts A, e.g., because d_B can be feasibly calculated given the proxy key plus d_A. *Asymmetric* proxy functions do not imply this bilateral trust. (Note that this model implies that proxy cryptography probably makes sense only in the context of public-key cryptosystems. Any secret-key cryptosystem with an asymmetric proxy function could be converted into a public-key system by publishing one key along with a proxy key that converts ciphertext for that key into ciphertext for a second key (which is kept secret.))

We can also categorize the asymmetric proxy schemes that might exist according to the convenience in creating the proxy key. In an *active asymmetric* scheme, B has to cooperate to produce the proxy key $\pi_{A\to B}$ feasibly, although the proxy key (even together with A's secret key) might not compromise B's secret key. In a *passive asymmetric* scheme, on the other hand, A's secret key and B's public key suffice to construct the proxy key. Clearly, any passive asymmetric scheme can be used as an active asymmetric scheme, and any asymmetric scheme can be used as a symmetric scheme.

Finally, we can distinguish proxy schemes according to the "metadata" they reveal about the identity of the keys being transformed. *Transparent* proxy keys reveal the original two keys to a third party. *Translucent* proxy keys allow a third party to verify a guess as to which two keys are involved (given their public keys). *Opaque* proxy keys reveal nothing, even to an adversary who correctly guesses the original keys (but who does not know the private keys involved).

1.3 Proxy schemes in theory and practice

The proxy relationship is necessarily transitive. If there are public proxy keys $\pi_{A\to B}$ and $\pi_{B\to C}$, then anyone can compute a proxy function for $A \to C$. Symmetric proxy schemes further establish equivalence classes of keys where the secret component of any key can be used to decrypt messages for any other key in the same class. Note that creating a single symmetric proxy key between a key in one class and a key in another effectively joins the two classes into one.

The notion of proxy cryptography is a rather natural generalization of public-key cryptography and has some pleasing theoretical properties. The proxy schemes we consider below have the additional property that anyone can use the proxy key $\pi_{A\to B}$ to transform the public key of A to the public key of B. For such proxy schemes, as we will see in the various examples below, certain aspects of the security of publishing a proxy key actually follow from the fact that anyone, trusted or not, can use a proxy key to transform ciphertext and keys.

For example, suppose random messages m and m' are encrypted with random secret keys a and b as $E(m,a)$, $E(m',b)$. Suppose that knowing the proxy key $\pi_{A\to B}$ enables Eve, who knows neither a nor b, to recover m or m'. Then, ignoring B altogether and starting with just two (presumably secure) ciphertexts $E(m,a)$ and $E(m',a)$, Eve can pick a random proxy key $r = \pi_{A\to Q}$ for some Q, transform $E(m',a)$ to $E(m',q)$ (where q is the unknown secret key of Q), transform A's public key into Q's public key, and proceed with the hypothesized cryptanalysis. We conclude that if it is safe for A to publish k messages then it is safe for A and B to publish a total of k messages *and* to publish a proxy key, provided only that Eve can successfully *apply* the proxy key to transform ciphertext and public keys.

Because proxy keys are tied to specific key pairs, it is not necessary in many applications to certify or otherwise take special care in distributing them (except to prevent denial-of-service). In particular, it is generally sufficient to rely on the certification and trust established in A (for encryption) or B (for signatures) when using proxy key $\pi_{A\to B}$, since a valid proxy key can by definition only be generated with the cooperation of the owner. Furthermore, the proxy function can be safely applied at any convenient time or place, by the message's sender or receiver, or at any intermediate (and possibly untrusted) point in the network.

Proxy functions potentially also have practical utility for key management in real systems. For example, some pieces of secure hardware (*e.g.*, smartcards) limit the number of secret keys that can be stored in secure memory, while some applications might require the ability to decrypt messages for more keys than the hardware can accommodate. With proxy cryptography, once a new key is created and a corresponding proxy key generated, the secret component of the old (or new) key can be destroyed, with the (public and externally-applied) proxy key maintaining the ability to decrypt for both. In effect, proxy functions allow us to increase the number of public keys without also increasing the number of secret bits or the amount of secret computation. Because proxy functions can be computed anywhere, messaging systems, such as electronic mail, can proxy "forward" messages encrypted with one key to a recipient who holds a different key. Proxy functions make it possible to associate a single key with a network or physical address but still decrypt messages forwarded (and proxied) from other addresses. Finally, proxy functions effectively allow changing or adding a key without obtaining new certificates or altering the distribution channel for the previous public key; this could be useful when it is difficult to distribute or certify new keys (e.g., old keys were published in widely-distributed advertisements or embedded in published software, or the certification authority charges high fees for new certificates).

1.4 Security of proxy schemes and ad hoc substitutes

If Alice wants Bob to be able to read her mail, instead of issuing a proxy key she might just give Bob her secret key (perhaps, obviating the need to involve Bob, by encrypting it in Bob's public key and publishing it). This would be inferior to using a proxy scheme for several reasons. First, as discussed above, Bob's computing environment may be limited and therefore incapable

of automatically processing encrypted secret keys; any new software to decrypt and manage such keys would have to run within the environment trusted by Bob. Proxy processing, on the other hand, can take place entirely outside of Alice's and Bob's trusted environments and without their active involvement. Furthermore, encrypting one's secret key with another's public key is not in general secure. The cryptosystem we present below, a variant[1] of ElGamal, is thought to be secure in part because the cryptanalysis problem is random-self-reducibile—which allows one to assert mathematically that recovering m from the public information $(e_a, E(m, e_a), e_b)$ is hard on average if it is hard at worst. The task of recovering m from $(e_a, E(m, e_a), E(d_a, e_b), e_b)$, however, may be considerably easier since $E(d_a, e_b)$, in the context of e_a and e_b, may leak information about d_a—specifically, the new cryptanalysis problem is probably not random-self-reducible and due to the problem's obscurity it is not clear what, if any, mathematical guarantees of security can be given. By contrast, the proxy scheme we give below is just as strong as the underlying ElGamal-like cryptosystem.[2]

1.5 Related work

A natural question to ask is whether there exist atomic proxy functions (and feasible schemes to generate proxy keys) for any public key cryptosystems.

Previous work on delegating the power to decrypt has focused on developing efficient transformations that allow the original recipient to forward *specific ciphertexts* to another recipient. Mambo and Okamoto[MO97] develop this formulation and give efficient transforms (more efficient than decryption and re-encryption) for ElGamal and RSA. Mambo, Usuda and Okamoto[MUO96] apply a similar notion to signature schemes.

While such schemes have value from the standpoint of efficiency, they are not, however, "atomic proxy cryptosystems" by our definition because the transforming function must be kept secret and applied online by the original keyholder on a message-by-message basis (the schemes are not atomic). The security semantics of these systems are essentially the same as a decryption operation followed by a re-encryption operation for the new recipient. Our formulation of proxy cryptography is distinguished from the previous literature by the ability of the keyholder to publish the proxy function and have it applied by untrusted parties without further involvement by the original keyholder.

2 Proxy encryption

Although the problem of proxy cryptography seems like a natural extension of public-key cryptography, existing cryptosystems do not lend themselves to obvious proxy functions. RSA[RSA78] with a common modulus is an obvious candidate, but that scheme is known to be insecure[Sim83][DeL84]. Similarly, there do not appear to be obvious proxy functions for many of the previous discrete-log-

[1] David Wagner notes that our proxy scheme can be extended to work with standard ElGamal[ElG85] encryption.

[2] Note that Bob of this example may be a government mandating that Alice provide him with access to her key. It has been argued that such a scheme makes the system as a whole less trustworthy due to the extra engineering effort involved; we argue here that in the case of random-self-reducible cryptosystems such as ElGamal variants, requiring Alice to encrypt her secret key using the government's public key may also weaken the underlying cryptosystem in the precise mathematical sense of spoiling the random-self-reducibility.

based cryptosystems. This is not to say, of course, that proxy functions for existing systems do not exist, only that we have not discoverd them.

We now describe a new secure discrete-log-based public-key cryptosystem that does have a simple proxy function. The scheme is similar in structure to ElGamal encryption[ElG85], but with the parameters used differently and the inverse of the secret used to recover the message. (This approach has merit beyond proxy encryption; [Hug94] proposed a Diffie-Hellman-like key agreement protocol based on the inverse of the secret, which allows a message's sender to determine the key prior to identifying its recipient).

2.1 Cryptosystem $\mathcal{X}$ (encryption)

Let p be a prime of the form $2q+1$ for a prime q and let g be a generator in Z_p^*; p and g are global parameters shared by all users. A's secret key a, $0 < a < p-1$, is selected at random and must be in Z_{2q}^*, *i.e.*, relatively prime to $p-1$. (A also calculates the inverse $a^{-1} \bmod 2q$). A publishes the public key $g^a \bmod p$. Message encryption requires a unique randomly-selected secret parameter $k \in Z_{2q}^*$. To encrypt m with A's key, the sender computes and sends two ciphertext values (c_1, c_2):

$$\begin{aligned} c_1 &= mg^k \bmod p \\ c_2 &= (g^a)^k \bmod p \end{aligned}$$

Decryption reverses the process; since

$$c_2^{(a^{-1})} = g^k \pmod p$$

it is easy for A (who knows a^{-1}) to calculate g^k and recover m:

$$m = c_1((c_2^{(a^{-1})})^{-1}) \bmod p$$

The speed of the scheme is comparable to standard ElGamal encryption, although initial key generation requires the additional calculation and storage of a^{-1}.

2.2 Symmetric proxy function for $\mathcal{X}$

Observe that the c_1 ciphertext component produced by Cryptosystem $\mathcal{X}$ is independent of the recipient's public key. Recipient A's key is embedded only in the c_2 exponent; it is sufficient for a proxy function to convert ciphertext for A into ciphertext for B to remove A's key a from c_2 and replace it with B's key b. Part of what a proxy function must do, then, is similar to the first step of the decryption function, raising c_2 to a^{-1} to remove a. The proxy function must also contribute a factor of b to the exponent. Clearly, simply rasing c_2 to a^{-1} and then to b would accomplish this, but obviously such a scheme would not qualify as a secure proxy function; anyone who examines the proxy key learns the secret keys for both A and B.

This problem is avoided, of course, by combining the two steps into one. Hence, the proxy key $\pi_{A\to B}$ is $a^{-1}b$ and the proxy function is simply $c_2^{\pi_{A\to B}}$. Note that this is a symmetric proxy function; A and B must trust one another bilaterally. B can learn A's secret (by multiplying the proxy key by b^{-1}), and A can similarly discover B's key. This proxy function is also translucent; the proxy key does not directly reveal A or B, but anyone can verify a guess by encrypting a message with A's public key, applying the proxy function, and comparing the result with the encryption of the same message (with the same k) with B's public key. Observe that applying the proxy function is more efficient than decryption and re-encryption, in that only one exponentiation is required.

2.3 Security of $\mathcal{X}$

First, we show that $\mathcal{X}$ is secure—that cleartext and secret keys cannot be recovered from ciphertext and public keys. Beyond that, we also show that publishing the proxy key compromises neither messages nor secret keys. Since recovering a secret key enables an adversary to recover a message and since cryptanalysis is easier with more information (i.e., a proxy key), it is sufficient to show that no cleartext is recoverable from ciphertext, public keys, and proxy keys. Specifically, we will show that the problem of recovering m from

$$(g^a, g^b, g^c, \ldots, mg^k, g^{ak}, a^{-1}b, a^{-1}c, \ldots).$$

is at least as hard as Diffie-Hellman.

Theorem 1 *Suppose there exists a randomized algorithm f that with probability $\epsilon > 1/|p|^{O(1)}$ succeeds in recovering m from the public information*

$$(g^a, g^b, \ldots, mg^k, g^{ak}, b/a, \ldots)$$

where the probability is taken over f's random choices as well as over m and the parameters a, b, and k. Then, for each $\eta = 2^{-|p|^{O(1)}}$, there exists a randomized polynomial-time algorithm for Diffie-Hellman that succeeds with probability $1 - \eta$.

Proof. For simplicity we assume there are only two public keys and one proxy key; the general case is similar. Suppose we had an algorithm F that always succeeds in recovering m from $(g^a, g^b, mg^k, g^{ak}, b/a)$. Then note that on input g^x, g^y, we have $F_1(g^x, g^y) = F(g^y, g^y, 1, g^x, 1)^{-1} = g^{x/y}$, and since $F_1(g, g^y) = g^{1/y}$ we'd have

$$F_2(g^x, g^y) = F_1(g^x, F_1(g, g^y)) = F_1(g^x, g^{1/y}) = g^{xy}.$$

In fact, f is only guaranteed to recover m with probability ϵ so we need to use the random-self-reducibility [Fei93] of the discrete log to achieve our objective.

On input g^x, g^y for $x, y \in Z_{2q}^*$ let f_1 pick $r, s, t, u \in Z_{2q}^*$ at random and query

$$f(g^{ry}, g^{sy}, g^t, g^{ux}, s/r).$$

By hypothesis, with probability ϵ we get $g^{t-(ux)/(ry)}$ from which f_1 can recover $g^{x/y}$. Since $f_1(g, g^y) = g^{1/y}$ we can define f_2 by

$$f_2(g^x, g^y) = f_1(g^x, f_1(g, g^y));$$

this is equal to

$$f_1(g^x, g^{1/y}) = g^{xy}$$

with probability at least ϵ^2.

Our next step is to construct an algorithm that runs correctly with high probability on most inputs in Z_{2q}^*. For this, we define the algorithm $f_3(g^x, g^y)$, $x, y \in Z_{2q}^*$, as follows. Pick r, s, t, u at random with $r, s, u \in Z_{2q}^*$ and $0 \le t < 2q$ even. Compute $f_2(g^{rx}, g^{sy}) = g^{rsxy+c}$ and $f_2(g^{(t+x)/u}, g^{uy}) = g^{ty+xy+c'}$ for some c, c' that depend on the respective input to f_2. Check whether $(g^{rsxy+c})^{1/rs} =$

$(g^{ty+xy+c'})/g^{ty}$ and is of the form g^z for $z \in Z_{2q}^*$ and if so output the common value, otherwise abort.

We need to show that the probability that f_3 outputs the correct answer is substantial and the probability that it outputs an incorrect answer is negligible. Note that the inputs to f_2 are random, so, by hypothesis, at least ϵ^4 of the time $c = c' = 0$ and therefore f_3 will answer correctly. For f_3 to answer incorrectly, we must have $c, c' \neq 0$ and $c/rs = c'$. Note also that in this case c and c' must be even so that $xy + c' = xy + c/(rs)$ are in Z_{2q}^*. Even conditioned on the four inputs to two calls to f_2, the six random variables r, s, t, u, x, y have two degrees of freedom left, and it is easy to see that r and s and therefore rs remain uniformly distributed. Thus, c/rs is uniformly distributed among even numbers modulo $2q$ and only equals c' with probability $1/q$. Thus if we repeat f_3 a total of $O(-\log \eta)/\epsilon^4$ times we will have a probability $1 - \eta$ of a correct answer and only a tiny chance of getting any incorrect answer.

Next, we show how to construct an algorithm $f_4(g^x, g^y)$ that succeeds, with high probability, on all inputs g^x, g^y such that $x, y \in Z_{2q}^*$. Pick r, s at random from Z_{2q}^* and compute $f_3(g^{rx}, g^{sy})^{1/rs}$. The input to f_3 is uniformly distributed, so by hypothesis $f_3(g^{rx}, g^{sy}) = g^{rsxy}$ with high probability and we recover g^{xy}.

Before considering general g^x, g^y we recall some facts about arithmetic modulo $2q$. The integers modulo $2q$ consist of 0, q, $(q-1)$ multiples of 2 (other than 0), and $(q-1)$ invertible elements (the odd numbers other than q). Given an input g^x where g is a primitive element modulo $p = 2q + 1$, one regards x modulo $2q$. We can learn whether x is invertible from g^x: If $x = 0$ then $g^x = 1$, if $x = q$ then $g^x = -1$, if x is odd then $(g^x)^q = g^q = -1$ and if x is even then $(g^x)^q = g^0 = 1$. (Raising g^x to the power q is polynomial-time, but expensive. However, we do not need to do this when using the cryptosystem.)

Finally, consider general input g^x, g^y. The cases $x = 0, x = q$ or $y = 0, y = q$ are easy to detect and handle, so we assume that we are not in one of these cases. We can determine s and t in $\{0, 1\}$ so that $x + sq, y + tq \in Z_{2q}^*$. We have $g^{xy} = g^{(x+sq)(y+tq)}/g^{xtq+ysq+stq^2} = \pm g^{(x+sq)(y+tq)} = \pm f_4(g^{x+sq}, g^{y+tq})$ (and we can determine the sign). □

Similarly one can show that recovering a from $(g^a, g^b, mg^k, g^{ak}, b/a)$ is as hard as the discrete log, so publishing the proxy key does not compromise a—not even to the level of Diffie-Hellman.

3 Proxy identification

In this section we describe a discrete-log-based identification scheme. With p, g, a as before, Alice wishes to convince Charlotte that she controls a; Charlotte will verify using public key g^a. As before, the proxy key $\pi_{A \to B}$ will be a/b—it will be safe to publish a/b and Alice and Charlotte can easily use a/b to transform the protocol so Charlotte is convinced that Alice controls b.

Note that in the case of a secure identification proxy key that transforms identification by A into identification by B, it is B whose secret is required to construct the proxy key because identification as B should not be possible without B's cooperation.

3.1 Cryptosystem $\mathcal{Y}$ (identification)

Let p and g be a prime and a generator in Z_p^*, respectively. Alice picks random $a \in Z_{2q}^*$ to be her secret key and publishes g^a as her public key. Each round of the identification protocol is as

follows:

- Alice picks a random $k \in Z_{2q}^*$ and sends Charlotte $s_1 = g^k$.
- Charlotte picks a random bit and sends it to Alice.
- Depending on the bit received, Alice sends Charlotte either $s_2 = k$ or $s_2' = k/a$.
- Depending on the bit, Charlotte checks that $(g^a)^{s_2'} = s_1$ or that $g^{s_2} = g^k$.

This round is repeated as desired. As with existing protocols, there may be ways to perform several rounds in parallel for efficiency [FFS88].

3.2 Symmetric proxy function for $\mathcal{Y}$

A symmetric proxy key is simply a/b. Proxy identification is useful as follows: Suppose Charlotte wants to run the protocol with g^b instead of g^a. Either Alice or Charlotte or any intermediary can use the proxy key to convert Alice's responses k/a to k/b. Also, either party can transform its share of the key pair (a, g^a) to b or g^b before any protocol takes place. Thus Alice and Bob can authenticate for each other but otherwise the protocol is secure. This proxy scheme is translucent.

3.3 Security of $\mathcal{Y}$

Theorem 2 *Protocol $\mathcal{Y}$, with or without proxy keys published, is a zero-knowledge protocol that convinces the verifier that the prover knows the secret key.*

Proof. Without proxy keys published, this protocol is similar to others in the literature (see, e.g., [FFS88]). Note that if a prover could produce both k/a and k then the prover could produce a from g^a (perhaps only with significant probability).

Now suppose that a proxy key a/b is published for random public keys g^a and g^b, and suppose that D can then impersonate A. Since D could already generate a random proxy key r and matching public key g^{ar}, it follows that D could impersonate A even without knowing a/b and g^b. Thus publishing proxy keys does not weaken the system. □

4 Proxy signature

The concept of proxy cryptography also extends to digital signature schemes. A signature proxy function transforms a message signature so that it will verify with a public key other than that of the original signer. In other words, a signature proxy function $\Pi(s, \pi_{A\to B})$ with proxy key $\pi_{A\to B}$ transforms signature s signed by the secret component of key A such that $V(m, \Pi(S(m, A), \pi_{A\to B}), B) =$ VALID, where $S(m, k)$ is the signature function for message m by key k and $V(m, s, k)$ is the verify function for message m with signature s by key k.

Again, existing digital signature schemes such as RSA[RSA78], DSA[NIS91], ElGamal[ElG85], etc. do not have obvious proxy functions (which, again, is not to say that such functions do not exist).

As in the case of proxy identification, in order to construct a proxy key that transforms A's signature into B's signature, B's secret must be required to construct the proxy key because signing for B should not be possible without B's cooperation.

Now we will see how to use the proxy identification scheme to construct a proxy signature scheme. We suppose there exists a hash function h whose exact security requirements will be discussed below. The parameters p, g, a, b are as before.

4.1 Cryptosystem $\mathcal{Z}$ (signature)

To sign a message m, Alice picks $k_1, k_2, \ldots k_\ell$ at random and computes $g^{k_1}, \ldots g^{k_\ell}$. Next Alice computes $h(g^{k_1}, \ldots g^{k_\ell})$ and extracts ℓ pseudorandom bits $\beta_1, \ldots, \beta_\ell$. For each i, depending on the i'th pseudorandom bit, Alice (who knows a) computes $s_{2,i} = (k_i - m\beta_i)/a$; that is, $s_{2,i} = (k_i - m)/a$ or $s_{2,i} = k_i/a$. The signature consists of two components:

$$\begin{aligned} s_1 &= (g^{k_1}, \ldots, g^{k_\ell}) \\ s_2 &= ((k_1 - m\beta_1)/a, \ldots, (k_\ell - m\beta_\ell)/a) \end{aligned}$$

To verify the signature, first the β_i's are recovered using the hash function. The signature is then verified one "round" at a time, where the i'th round is $(g^{k_i}, (k_i - m\beta_i)/a)$. To verify $(g^k, (k - m\beta)/a)$ using public key g^a, the recipient Charlotte raises (g^a) to the power $(k - m\beta)/a$ and checks that it matches $g^k / g^{m\beta}$.

4.2 Symmetric proxy function for $\mathcal{Z}$

A symmetric proxy key $\pi_{A \to B}$ for this signature scheme is a/b. The proxy function Π leaves s_1 alone and maps each component $s_{2,i}$ to $s_{2,i}\pi_{A \to B}$. The proxy scheme is translucent.

4.3 Security of $\mathcal{Z}$

This scheme relies on the existence of a "hash" function h. Specifically,

Assumption 1 *We assume there exists a function h such that:*

- *On random input (g^a, m), it is difficult to generate $\{r_i\}$ and $\{\beta_i\}$ such that*

$$h(g^{ar_1 + m\beta_1}, \ldots, g^{ar_\ell + m\beta_\ell}) = \langle \beta_1 \ldots, \beta_\ell \rangle.$$

- *More generally, it is difficult to generate such $\{r_i\}$ and $\{\beta_i\}$ on input g^a, m, and samples of signatures on random messages signed with a.*

It is not our intention to conjecture about the existence of such functions h. In particular, we do not know the relationship between Assumption 1 and assumptions about collision freedom or hardness to invert.[3] We note that this generic transformation of a protocol to a signature scheme has appeared in the literature [FS86].

[3] Assumption 1 *does* imply that, on random input g^a, it is hard to find $\langle r_i \rangle$ making all the β_i's zero, i.e., such that $h(g^{ar_1}, \ldots, g^{ar_\ell}) = 0$.

We now analyze Assumption 1. Note that in order to produce a legitimate signature on m that verifies with g^a, a signer needs to produce $\langle g^{k_i} \rangle$ and $\langle (k_i - m\beta_i)/a \rangle$. Thus, putting $\langle \beta_i \rangle = h(\langle g^{k_i} \rangle)$ and then $\langle r_i \rangle = \langle (k_i - m\beta_i)/a \rangle$, it is straightforward to see that the signer could actually produce r_i's and β_i's of the stated type in the course of producing the signature.

While we do not address the security of h, we can state that issuing proxy keys does not weaken the system.

Theorem 3 *Suppose h satisfies Assumption 1. Then, for most b, it is also hard to produce $\{r_i\}$ and $\{\beta_i\}$ given additional input $a/b, g^b$, and samples of messages signed with b.*

Proof. As above, a signer not having access to b's messages and proxy keys could simulate this by choosing a random proxy key r, generating $g^b = g^{ar}$, and convert some messages signed with a into messages signed with b. □

5 Conclusions

Intuitively, atomic proxy cryptography is a fairly natural extension of the basic notion of public-key cryptography. It surely seems plausible, given that there exist cryptosystems that can grant the ability to encrypt without granting the ability to decrypt, that there might also exist cryptosystems that can grant the ability to *re-encrypt* without granting the ability to decrypt. However, it is not at all obvious whether there exist atomic proxy schemes in general.

Indeed, while this paper has demonstrated that there do exist efficient and secure public-key encryption and signature schemes with symmetric atomic proxy functions, this observation probably raises more new questions than it answers. In particular, do proxy functions exist for public-key cryptosystems based on problems other than discrete-log? (One possibility is that, for some cryptosystems, proxy functions do exist but it is infeasible to find a proxy key.) More importantly, we have yet to discover a secure *asymmetric* proxy function of any kind; asymmetric proxy functions are probably much more useful in practice, since there are likely many situations where trust is only unidirectional. Are there cryptosystems for which asymmetric proxy functions exist?

6 Acknowledgements

The authors thank Steve Bellovin, Jack Lacy, Dave Maher, Andrew Odlyzko and David Wagner for helpful discussions and comments on earlier drafts.

References

[DeL84] J. M. DeLaurentis. A further weakness in the common modulus protocol for the RSA cryptoalgorithm. *Cryptologia*, 8:235–239, 1984.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. IT-31(4):469–472, July 1985.

[Fei93] Joan Feigenbaum. Locally random reductions in interactive complexity theory. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 13:73–98, 1993.

[FFS88] U. Feige, A. Fiat, and A. Shamir. Zero knowledge proofs of identity. *Journal of Cryptology*, 1(2):77–94, 1988.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *Crypto 86*, number 263 in LNCS, pages 186–194, Santa Barbara, CA, USA, August 1986.

[Hug94] Eric Hughes. An encrypted key transmission protocol. *CRYPTO '94* Rump Session presentation, August 1994.

[MO97] M. Mambo and E. Okamoto. Proxy cryptosystems: delegation of the power to decrypt ciphertexts. *IEICE Trans. Fundamentals*, E80-A(1), 1997.

[MUO96] M. Mambo, K. Usuda, and E. Okamoto. Proxy signatures: delegation of the power to sign messages. *IEICE Trans. Fundamentals*, E79-A(9), 1996.

[NIS91] NIST. A proposed federal information processing standard for digital signature standard (DSS). Draft Tech. Rep. FIPS PUB XXX, August 1991.

[RSA78] Ron L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, February 1978.

[Sim83] G. J. Simmons. A "weak" privacy protocol using the RSA crypto algorithm. *Cryptologia*, 7:180–182, 1983.

Atomic Proxy Cryptography

Matt Blaze

AT&T Shannon Laboratory

This talk introduces *atomic proxy cryptography*, in which an atomic proxy function, in conjunction with a public proxy key, converts ciphertext (messages in a public key encryption scheme or signatures in a digital signature scheme) for one key (k_1) into ciphertext for another (k_2). Proxy keys, once generated, may be made public and proxy functions applied in untrusted environments. Various kinds of proxy functions might exist; {\em symmetric} atomic proxy functions assume that the holder of k_2 unconditionally trusts the holder of k_1, while {\em asymmetric} proxy functions do not. It is not clear whether proxy functions exist for previous public-key cryptosystems. Several new public-key cryptosystems with symmetric proxy functions are described: an encryption scheme, which is at least as secure as Diffie-Hellman, an identification scheme, which is at least as secure as the discrete log, and a signature scheme derived from the identification scheme via a hash function.

Full paper available.

This is joint work with Martin Strauss.

Matt Blaze, Atomic Proxy Cryptography

Gates 498, 10/20/98, 4:15 PM

Matt Blaze's Technical Papers

Last updated 6 August 2006

Many of my technical papers are available here. Newer papers are usually in Adobe PDF format; like it or not, PDF is the de facto standard format for scientific papers these days. Most of the older papers are in PostScript format; you'll need a PostScript printer or viewer (such as GhostView) to read them. Most of these files have also been converted to Adobe PDF format (using ps2pdf) and can be viewed or printed with a PDF viewer such as Acrobat, acroread4, or xpdf. If you have a choice, you'll probably find the PostScript version looks and works better than the PDF version does (ps2pdf doesn't do particularly well with some of the fonts). A few papers are available as plain ASCII text or LaTeX source.

Wiretapping, Surveillance and Countermeasures

The Trustworthy Network Eavesdropping and Countermeasures (TNEC) project studies the reliability of communications interception systems and technologies. A better understanding of the limitations of eavesdropping techniques could lead to more trustworthy law enforcement wiretap evidence (or at least more appropriate treatment of electronic evidence), networks with properties that inherently frustrate (or facilitate) interception, and new techniques for achieving communications security.

One of our first efforts is a comprehensive analysis of the wiretapping technologies used by law enforcement (for both voice and data). We have found serious exploitable weaknesses in fielded interception systems. For details, including audio demos of novel eavesdropping countermeasures, see the wiretapping web page here.

- M. Sherr. E. Cronin, S. Clark and M. Blaze. http://www.crypto.com/papers/wiretapping/ web page.

- M. Sherr, E. Cronin, S. Clark and M. Blaze. "Signaling Vulnerabilities in Wiretapping Systems." *IEEE Security and Privacy.* November/December 2005. [PDF].

Similar vulnerabilities exist in digital Internet eavedropping systems as well:

- E. Cronin, M. Sherr, and M. Blaze. "The Eavedsdropper's Dilemma." Technical Report MS-CIS-05-24. University of Pennsylvania. 2005. [PDF].

Another focus of the TNEC project examines local host-based surveillance. The *JitterBug* demonstrates a novel eavesdropping threat against typed keyboard input. Commercially-available hardware keyboard "sniffers" can easily capture and store an unsuspecting user's keystrokes. Because a subverted keyboard has no direct network connection, sniffer attacks are generally assumed to require either support software on the host or periodic in-person access by the attacker to retrieve the data. We show that this need not be the case. A new technique based on "JitterBugs" can exflitrate captured data entirely through subtle perturbations in the precise times at which typed keystrokes are passed to the host. Whenever a user runs an interactive network application (such as SSH), an attacker can derive previously captured keystrokes entirely by observing the timing of network packets, even from across the

Internet or via encrypted wireless traffic. The JitterBug demonstrates that input devices must be scrutinized as part of any trusted computing base and, more generally, that simple "supply chain attacks" can represent a practical and serious threat to data confidentiality. (Gaurav Shah and Andres Molina won the Best Student Paper award at USENIX Security 2006 for this work.)

- G. Shah, A. Molina, and M. Blaze. "Keyboards and Covert Channels." *Proc. 15th USENIX Security Symposium.* Vancouver, BC. August 2006. [PDF].

- Gaurav Shah's JitterBug page: `http://www.cis.upenn.edu/~gauravsh/jitterbug.html`. JitterBug prototype code and PCB template.

Physical and "Human-Scale" Security

Cryptologic techniques can be applied outside of computers and networks, Perhaps surprisingly, the abstractions used in analyzing secure computing and communications systems turn out also to be useful for understanding mechnical locks and their keyspaces. Indeed, modeling master keyed locks as online authentication oracles leads directly to efficient solutions for what might naively seem like exponential problems for the attacker. In fact, it seems like almost a textbook example, as if master keying practices for locks were designed specifically to illustrate this class of weakness. We sometimes assume that hardware-based security is inherently superior to that based in software, but even the humble mechanical lock can be just as insecure as complex computing systems, and can fail in similar ways.

A widely circulated paper of mine describes attacks against master keyed mechanical locks. For an overview of the attack, which was described in the January 23rd 2003 *New York Times*, click here. For a brief commentary on the reaction to this paper, see my essay, "Keep it secret, stupid!" (click here), which was originally posted to comp.risks.

(Warning: there are embedded photos in this paper; they make the PS and PDF files very large. The GZIPed PostScript version is 5.7MB long (uncompresses to 14MB), and the PDF version is 4MB long.)

- M. Blaze. "Cryptology and Physical Security: Rights Amplification in Master-Keyed Mechanical Locks." March 2003. *IEEE Security and Privacy.* March/April 2003. [GZIPed PostScript], [PDF].

- My *Notes on Picking Pin Tumbler Locks*, intended primarily for use by students in my security seminar, can be found here [HTML].

While the security metrics and mechanical safeguards used in safes and vaults may not relay on the latest technology, they are often quite ingenious. They may have much to teach computer security. Some of what I understand about the subject is in the survey paper below (warning -- heavily illustrated 2.5MB .pdf file). And for a brief commentary on the reaction to *this* paper, see my essay, "the second sincerest form of flattery" (click here), which was originally posted to `interesting-people`.

- M. Blaze. "Safecracking for the Computer Scientist." *U. Penn CIS Department Technical Report*. 7 December 2004 (revised 20 December 2004). [PDF].

This position paper, presented at the Cambridge Security Protocols Workshop 2004, introduces and advocates the "Human Scale Security Project," which supports the above work.

- M. Blaze. "Toward a broader view of security protocols." *12th Cambridge International Workshop on Security Protocols*. Cambridge, UK. April 2004. [PDF].

Trust Management

These papers introduce the "trust management" approach to specifying and enforcing security policy.

- The Trust Management Web Page, updated regularly.
- M. Blaze, J. Ioannidis, A. Keromytis. "Offline Micropayments without Trusted Hardware." *Financial Cryptography 2001*. Grand Cayman, February 2001. [PostScript], [PDF].
- M. Blaze, J. Ioannidis, A. Keromytis. "Trust Management for IPSEC." *NDSS 2001*. San Diego, February 2001. [PDF].
- M. Blaze, J. Feigenbaum, J. Ioannidis, A. Keromytis. The KeyNote Trust Management System, Version 2. *RFC-2704*. IETF, September 1999. [ASCII Text].
- M. Blaze, J. Ioannidis, A. Keromytis. "Compliance Checking and IPSEC Policy Management." *Internet Draft*. `draft-blaze-ipsp-trustmgt-00.txt`. IETF, March 2000. [ASCII Text].
- M. Blaze, J. Ioannidis, A. Keromytis. "DSA and RSA Key and Signature Encoding for the KeyNote Trust Management System." *RFC-2792*. IETF, March 2000. [ASCII Text]
- M. Blaze, J. Ioannidis, and A. Keromytis. " Trust Management and Network-Layer Security Protocols." *1999 Cambridge Protocols Workshop*. Cambridge, April 1999. [PostScript], [PDF], [LaTeX Source].
- M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. "The Role of Trust Management in Distributed Systems Security." Chapter in *Secure Internet Programming: Security Issues for Mobile and Distributed Objects*, (Vitek and Jensen, eds.) Springer-Verlag, 1999. [PostScript], [PDF].
- M. Blaze, J. Feigenbaum, M. Strauss. "Compliance-Checking in the PolicyMaker Trust-Management System." *Proc. 2nd Conference on Financial Cryptography*. Anguilla 1998. LNCS 1465, pp 251-265, Springer-Verlag, 1998. [PostScript], [PDF].
- M. Blaze, J. Feigenbaum and J. Lacy. "Decentralized Trust Management." *IEEE Symposium on Security and Privacy*, Oakland, CA. May 1996. [PostScript], [PDF].

Angelos Keromytis's KeyNote Trust Management toolkit and open-source reference

implementation is available here as a GZIPed TAR archive. The toolkit runs under most Unix-like (BSD, linux, etc.) platforms, with limited support for Win32 platforms.

Also see Angelos Keromytis' KeyNote web page for the latest details on the KeyNote implementation.

Remotely-Keyed Encryption

These papers introduce and formalize the notion of "remotely-keyed" encryption, in which a low-bandwidth, but trusted device (such as a smart card) assists a high-bandwidth, but untrusted host with bulk encryption.

- M. Blaze, J. Feigenbaum, and M. Naor. "A Formal Treatment of Remotely Keyed Encryption (Extended Abstract)". Eurocrypt '98, Helsinki. LNCS 1403 pp. 251-265. [PostScript], [PDF].

- M. Blaze. "High-Bandwidth Encryption with Low-Bandwidth Smartcards." January 18, 1996. *Cambridge Workshop on Fast Software Encryption*, February 1996. [PostScript], [PDF].

Key Escrow

These papers describe and evaluate various key escrow proposals, from a technical (as opposed to political) perspective.

- *The Risks of Key Recovery, Key Escrow, and Trusted Third-Party Encryption* (second edition). June 1998. [HTML], [PDF].

- *The Risks of Key Recovery, Key Escrow, and Trusted Third-Party Encryption* (first edition). May 1997. (OBSOLETE: superseded by second edition, above). [ASCII Text], [PDF], [PostScript].

- M. Blaze. "Oblivious Key Escrow." *First Cambridge Workshop on Information Hiding* May 1996. Springer 1997. [PostScript], [PDF], [LaTeX source].

- Memo from NSA regarding key length report, with comments from M. Blaze and W. Diffie. July 18, 1996. [ASCII Text].

- M. Blaze, W. Diffie, R. Rivest, B. Schneier, T. Shimomura, E. Thompson and M. Wiener. "Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security". Report of ad hoc panel of cryptographers and computer scientists. January 1996. [ASCII Text], [PDF], [PostScript].

- M. Blaze, J. Feigenbaum and F.T. Leighton. "Master-Key Cryptosytems." Abstract presented at *Crypto '95 (rump session)*, Santa Barbara, CA, August 1995. [PostScript], [PDF]

- M. Blaze. "Protocol Failure in the Escrowed Encryption Standard." *Proceedings of Second ACM Conference on Computer and Communications Security*, Fairfax, VA, November 1994. [PostScript], [PDF].

Network-Layer Security

These papers describe the design and implementation network-layer and related security protocols, including JFK, a secure key exchange protocol, and swIPe, a predecessor to the IPSEC standard. (At this point, swIPe is of primarily historical interest, although the USENIX paper should be of some value to IPSEC implementors. JFK is a useful key exchange protocol that should be especially valuable for IPSEC and network security key management).

- W. Aiello, S. M. Bellovin, M. Blaze, R. Canetti, J. Ioannidis, A. D. Keromytis, and O. Reingold. "Efficient, DoS-Resistant, Secure Key Exchange for Internet Protocols." In Proc. ACM Computer and Communications Security (CCS) Conference. November 2002, Washington, DC. (pp 48-58). [PDF].

- J. Ioannidis and M. Blaze. "The swIPe IP Security Protocol." *Internet Draft*. December 1993. [ASCII Text].

- J. Ioannidis and M. Blaze. "Architecture and Implementation of Network Layer Security Under UNIX." *Proceedings of the Fourth USENIX Security Workshop*, October 1993. [PostScript], [PDF].

Cryptographic Applications

- R. Levein, L. McCarthy, M. Blaze. "Transparent Internet E-mail Security (DRAFT)". August 9, 1996. [PostScript], [PDF].

- M. Blaze and S.M. Bellovin. "Session-Layer Encryption." *Proceedings of the USENIX Security Workshop*, June 1995. [PostScript].

- M. Blaze. "Key Management in an Encrypting File System." *USENIX Summer 1994 Technical Conference*, Boston, MA, June 1994. [PostScript], [PDF].

- M. Blaze. "A Cryptographic File System for Unix." *Proceedings of the First ACM Conference on Computer and Communications Security*, Fairfax, VA, November 1993. [PostScript], [PDF].

  The latest CFS code can be found here.

Ciphers and Algorithms

- S. M. Bellovin, M. Blaze. "Cryptographic Modes of Operation for the Internet." NIST Workshop on AES Modes. Santa Barbara, CA. August 2001. [PDF].

- M. Blaze, M. Strauss. "Atomic Proxy Cryptography." Full version of our *EuroCrypt '98* paper. May 1997. [PostScript], [PDF].

- M. Blaze. "Efficient Symmetric-Key Ciphers Based on an NP-Complete Subproblem (DRAFT)". October 2, 1996. [PostScript], [PDF]

- M. Blaze and B. Schneier. "The MacGuffin Block Cipher Algorithm." *Leuven Workshop on Cryptographic Algorithms*, Leuven, Belgium, December 1994.

[PostScript], [PDF].

Cryptography Policy, Export Regulations, and Politics

- M. Blaze. Declaration in Felten, et al v. RIAA. 13 August 2001. [ASCII Text].
- S. Bellovin, M. Blaze, D. Farber, P. Neumann, E. Spafford. "Comments on the Carnivore System Technical Review." Formal comments to the US Department of Justice. 3 December 2000. [HTML].
- M. Blaze & S. M. Bellovin. "Tapping, Tapping on my Network Door." INSIDE RISKS 124. *CACM*, October 2000. [HTML].
- M. Blaze. "Cryptography Policy and the Information Economy." Draft. 17 December 1996. [PostScript], [PDF], [ASCII Text].
- My prepared testimony before the Senate Commerce Committee subcommittee on Science, Technology, and Space. June 26, 1996 [ASCII Text].
- M. Blaze. "My Life as an International Arms Courier." January, 1995. Adapated from posting to *comp.risks* [ASCII Text]

Peer-to-Peer Networking

My dissertation work, over ten years ago, anticipated and analyzed what we would now call "Peer-to-Peer" file distribution.

- M. Blaze. Caching in Large-Scale Distributed File Systems. PhD thesis. Princeton University Department of Computer Science. November 1992. [PostScript].

Other People's Papers

From time to time, I make available papers from other researchers that I didn't write myself but that are of wide interest and don't otherwise have a home. Here's what's available now:

- S. Fluhrer, I. Mantin and A. Shamir. Weaknesses in the Key Scheduling Algorithm of RC4. Preliminary Draft, July 25, 2001. [PostScript].
- A. Biryukov and A. Shamir. Real Time Cryptanalysis of the Alleged A5/1 on a PC. Preliminary Draft, December 9, 1999. [PostScript].

Click here to return to the crypto.com home page.

Bruce Schneier

Crypto Bibliography

Citations by First Author - B

A. Back, U. Möller, and A. Stiglic, Traffic Analysis Attacks and Trade-Offs in Anonymity Providing Systems, Proceedings of the 4th Information Hiding Workshop (IHW2001), Springer-Verlag, LNCS v. 2137, pp. 243-254. [.pdf]

S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, A Message Authentication Code based on Latin Squares, Australian Conference on Information Security and Privacy (ACISP '97), Springer-Verlag, LNCS 1270, pp. 194-203, 1997. [.ps.Z]

S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, On Password-Based Authenticated Key Exchange using Collisionful Hash Functions. In Australian Conference on Information Security and Privacy (ACISP '96), Springer-Verlag, LNCS 1172, pp. 299-310, 1996. [.ps.Z]

S. Bakhtiari, R. Safavi-Naini, and J. Piprzyk, On Selectable Collisionful Hash Functions, Australian Conference on Information Security and Privacy (ACISP '96), Springer-Verlag, LNCS 1172, pages 287-298, 1996. [.ps.Z]

T. Baldin, G. Bleumer, and R. Kanne, CryptoManager - Eine intuitive Programmierschnittstelle für kryptographische Systeme; Sicherheitsschnittstellen - Konzepte, Anwendungen und Einsatzbeispiele, Proc. Workshop Security Application Programming Interfaces 94, Deutscher Universitäts Verlag, München 1994, 79-94. [.ps.gz]

T. Baldin and G. Bleumer, CryptoManager++ -- An object oriented software library for cryptographic mechanisms; 12th IFIP International Conference on Information Security (IFIP/Sec '96), Chapman & Hall, London 1996, 489-491. [.ps.gz]

D. Balfanz and L. Gong, Experience with Secure Multi-Processing in Java, Proceedings of the 18th IEEE International Conference on Distributed Computing Systems (ICDCS), Amsterdam, Netherlands, May 1998. [.ps.gz]

J. Bar-Ilan and D. Beaver, Non-Cryptographic Fault-Tolerant Computing in a Constant Expected Number of Rounds of Interaction (extended abstract); Proceedings of **PODC**, ACM, 1989, 201-209. [.pdf]

R. Bar-Yehuda, B. Chor, E. Kushilevitz, and A. Orlitsky, Privacy, Additional Information, and Communication, IEEE IT 39(6), 1993, pp. 1930-1943. [.ps.Z]

N. Baric and B. Pfitzmann, Collision-Free Accumulators and Fail-Stop Signature Schemes Without Trees; Eurocrypt '97, LNCS 1233, Springer-Verlag, Berlin 1997, 480-494. [.ps.gz]

E. Basturk, M. Bellare, C. S. Chow, and R. Guerin, Secure transport protocols for high-speed networks, IBM Research Report 19981, March, 1994.

O. Baudron, H. Gilbert, L. Granboulan, H. Handschuh, A. Joux, P. Nguyen, F. Noilhan, D. Pointcheval, T. Pornin, G. Poupard, J. Stern, and S. Vaudenay, Report on the AES Candidates, Proceedings of the Second AES Candidate Conference, Rome, Italy, 1999. [.pdf]

B. Baum-Waidner, B. Pfitzmann, and M. Waidner, Unconditional Byzantine Agreement with Good Majority; STACS'91, LNCS 480, Springer-Verlag, Heidelberg 1991, 285-295. [.ps.gz]

D. Bayer, S. Haber, and W. Stornetta, Improving the Efficiency and Reliability of Digital Time-Stamping, Sequences II: Methods in Communication, Security, and Computer Science, eds. R. Capocelli, A. DeSantis, and U. Vaccaro, Springer-Verlag, 1993, pp. 329-334. [.pdf]

P. Beauchemin, G. Brassard, C. Crépeau, C. Goutier, and C. Pomerance, Two observations on probabilistic primality testing; In Advances in Cryptology: Proceedings of Crypto '86, volume 263 of Lecture Notes in Computer Science, pages 443-450. Springer-Verlag, 1987. [.ps.gz]

P. Beauchemin, G. Brassard, C. Crépeau, C. Goutier, and C. Pomerance, The generation of random

numbers that are probably prime, Journal of Cryptology, 1(1):53-64, 1988. [.ps]

D. Beaver, S. Micali, and P. Rogaway, The Round Complexity of Secure Protocols (extended abstract); Proceedings of the 22nd **STOC**, ACM, 1990, 503-513. [.ps] [.ps.gz]

D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, Security with Low Communication Overhead (extended abstract), Advances in Cryptology - Crypto '90 Proceedings, Springer-Verlag, 1991, 62-76. [.pdf]

D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, Locally Random Reductions: Improvements and Applications, Journal of Cryptology, 10 (1997), pp. 17-36. [.pdf] [.ps]

D. Beaver, Commodity-Based Cryptography (extended abstract); Proceedings of the 29th **STOC**, ACM, 1997, 446-455. [.pdf]

D. Beaver and S. Haber, Cryptographic Protocols Provably Secure Against Dynamic Adversaries (extended abstract); Advances in Cryptology - Eurocrypt '92, Springer-Verlag, 1993, 307-323. [.pdf]

D. Beaver, J. Feigenbaum, and V. Shoup, Hiding Instances in Zero-Knowledge Proof Systems (extended abstract), in Advances in Cryptology - Crypto '90, Lecture Notes in Computer Science, vol. 537, Springer, Berlin, 1991, pp. 326-338. [.pdf]

D. Beaver, S. Micali, and P. Rogaway, The round complexity of secure protocols; Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing, (STOC 90), 1990, 503-513. [.ps] [.ps.gz]

D. Beaver, Foundations of Secure Interactive Computing (extended abstract); Advances in Cryptology - Crypto '91 Proceedings, Springer-Verlag, 1992, 377-391. [.pdf]

D. Beaver and S. Goldwasser, Multiparty Computation with Faulty Majority, Advances in Cryptology: Crypto '89, ed. Gilles Brassard. [.pdf]

D. Beaver and N. So, Global, Unpredictable Bit Generation Without Broadcast (extended abstract); Advances in Cryptology - Eurocrypt '93, Springer-Verlag, 1994, 424-434. [.pdf]

D. Beaver, Efficient Multiparty Protocols Using Circuit Randomization (extended abstract); Advances in Cryptology - Crypto '91 Proceedings, Springer-Verlag, 1992, 420-432. [.pdf]

D. Beaver, How to Break a "Secure" Oblivious Transfer Protocol (extended abstract); Advances in Cryptology - Eurocrypt '92, Springer-Verlag, 1993, 285-296. [.pdf]

D. Beaver, J. Feigenbaum, R. Ostrovsky, and V. Shoup, Instance-Hiding Proof Systems; submitted for journal publication. Available as DIMACS Technical Report 93-65, Rutgers University, Piscataway, 1993. [.ps.Z]

R. Beigel and J. Feigenbaum, On Being Incoherent Without Being Very Hard, Computational Complexity, 2 (1992), pp. 1-17.

A. Beimel, Y. Isahi, T. Malkin, and E. Kushilevitz, One-way functions are essential for single-server private information retrieval, Proc. of the 31st Annu. ACM Symp. on the Theory of Computing (STOC), pp. 89-98, 1999. [.ps]

A. Beimel and B. Chor, Secret Sharing with Public Reconstruction, IEEE Trans. on Info. Theory, 44 (5):1887-1896, 1998. Extended abstract in Crypto '95. [.ps]

A. Beimel and M. Franklin, Reliable communication over partially authenticated networks, Theoretical Computer Science, (220)1:185--210, 1999. Preliminary version in WDAG '97, volume 1320 of LNCS, pages 245-259, Springer, 1997. [.ps]

A. Beimel, Secure Schemes for Secret Sharing and Key Distribution, Ph.D. Thesis, Dept. of Computer Science, Technion, 1996. [.ps]

A. Beimel, T. Malkin, and S. Micali, The All-or-Nothing Nature of Two-Party Secure Computation, CRYPTO '99., vol. 1666 of LNCS, pages 80 - 97, 1999. [.ps]

A. Beimel and B. Chor, Universally ideal secret sharing schemes. IEEE Trans. on Info. Theory, 40 (3):786-794, 1994. Extended abstract in Crypto '92. [.ps]

This textbook is the first for many years which covers all the principal topics of telecommunications. It provides a comprehensive coverage for those students seeking a sound foundation in the subject. In addition, its rigorous treatment of both theory and applications makes the book suitable for students intending to follow more specialist courses.

The authors have provided a sound analytical base, and developed a number of theoretical models. However, the limitations and assumptions of these models are always stressed, and emphasis is placed on relating the theory to practical problems.

The book is generously illustrated with more than 300 diagrams; each chapter contains problems, with solutions, and references to more specialized texts.

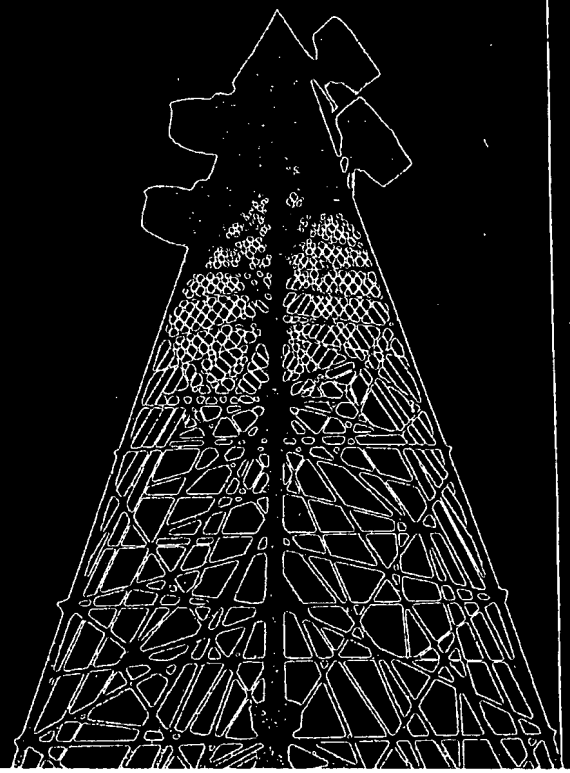
John Dunlop is a senior lecturer at the University of Strathclyde. His research interests include speech signal processing, underwater communications, digital processing of radar signals, local area communications networks and distributed processor systems.

Geoffrey Smith is a reader at the University of Strathclyde and is responsible for teletraffic engineering research. He is also involved in research in packet-switched computer communication networks and performance evaluation of several local area network mechanisms when carrying both voice and data traffic.

Van Nostrand Reinhold (UK)

TELECOMMUNICATIONS ENGINEERING

J. Dunlop & D.G. Smith



First published in 1984 by
Van Nostrand Reinhold (UK) Co. Ltd
Molly Millars Lane, Wokingham, Berkshire, England

Reprinted 1984, 1986, 1987

Typeset in 9/11 pt Times by Thomson Press (India) Ltd, New Delhi

Printed and bound in Hong Kong

Library of Congress Cataloging in Publication Data

Dunlop, J.
Telecommunications engineering.

Includes bibliographies and index.
1. Telecommunications. I. Smith, D.G. II. Title.
TK 5101.D845 1984 621.38 84-3531
ISBN 0-442-30586-9 (pbk.)

Table 10.5 Zone numbers

1	North America	6	Australasia
2	Africa	7	USSR
3	Europe	8	Eastern Asia
4	Europe	9	Far East and Middle East
5	South America	0	Spare

(USSR). Throughout each of these zones there is a linked numbering scheme that means, for example, that no subscriber in Canada has the same national number as a subscriber in the USA. Consequently, to connect to anyone in zone 1 the digit 1 is followed by the national number. A similar situation exists in the USSR. Europe is at the other extreme; there are many countries with large national networks that have nine digits in their national numbers. For these, a two-digit country code is required, and that can only be achieved by having two zone numbers allocated to Europe.

The division of the world into the zones shown in Table 10.5 is intended to be satisfactory until early in the next century, but clearly, as some large countries develop their telephone networks, some adjustment will be necessary at some future time.

The national and international numbering schemes we have discussed above are the simplest. However, in several parts of the world there are small exceptions, particularly in regard to local calls. In the scheme where the national number is used for all calls within a country, it can lead to irritation on the part of the subscriber and long set-up times for the exchange equipment. Consequently, in many countries local calls use a shorter code. For calls within the same area, the area code is omitted, and for small single-exchange areas no exchange code is used for own-exchange calls. Coupled with this last arrangement will be a very short code for calls to adjacent exchanges; these arrangements are particularly well suited to rural areas. The disadvantage of short codes is that they change with the location of the calling subscriber, and therefore a short code directory must be available in each exchange area.

10.33 Routeing Calls

The early type of switching equipment, called step-by-step or Strowger, operated by using the dialled pulses to move the selectors to the position corresponding to the digit dialled. In many ways this was an excellent system, but one major disadvantage was that it allowed no flexibility in the way calls were routed – the route was pre-determined by the dialled digits. Although some systems were modified to overcome this problem it was not until common-control equipment became widely used that the path a call took between calling and called subscribers could be chosen to allow the most efficient use of the available capacity in the system. The function of the common control in the routing process was to store the dialled digits in a register and then translate them into routing digits which would indicate to the switching

system the path to take through the network. This register – translator combination is essential to automatic trunk and international dialling schemes; it allows the telephone administration to manage the system efficiently by changing routes as circumstances alter without having to change subscribers' numbers. This therefore separates the subscriber from the system. The subscriber dials the national number from any location and the register–translator automatically selects an appropriate route.

10.34 Digital Systems

Several factors have acted to push telephony from analogue to digital working. To make such a major change, telephone administrations and equipment manufacturers have been persuaded that it is in digital operation that the future lies, and the decade from the mid-seventies has been characterized in all equipment producing countries by huge investments of manpower and plant in a race to manufacture an efficient digital telecommunications system. There are more manufacturers involved than at any other time and great financial commitments have been made in the hope that a market will be available for the many products being developed.

Traditionally the two basic elements of a telephone system were transmission and

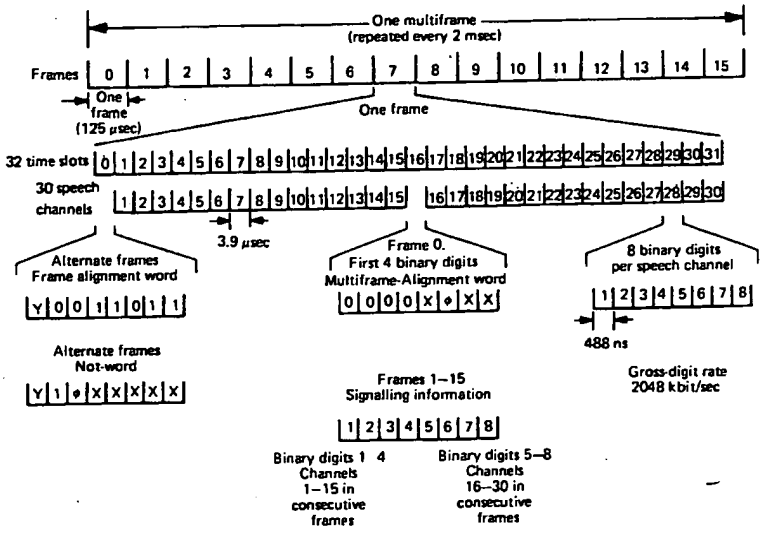


Fig. 10.37 32-frame PCM multiplex frame arrangement: *x*, digits not allocated to any particular function and set to state *one*; *y*, reserved for international use (normally set to state *one*); 0, digits normally *zero* but changed to *one* when loss of frame alignment occurs and/or system-fail alarm occurs (TS0 only) or when loss of multiframe alignment occurs (TS16 only).

switching. However, with digital operation the whole system is considered as an entity.

Following the development of PCM several countries installed PCM links between analogue exchanges. These worked in a basic 24- or 32-channel format with an encoder and decoder. We can see how the 32-channel system is formed with reference to Fig. 10.37. Strictly speaking, it should be referred to as a 30-channel, 32-time-slot system, because two of the time slots contain signalling and synchronizing information, not speech samples. The sampling rate of each speech channel determines the length of each time slot. For telephony the sampling rate used is 8 kHz, which means that the time between adjacent samples of the same channel is 125 μsec. One frame, consisting of 32 time-slots lasts for this time. A time slot is therefore approximately 3.9 μsec long. For reasons that we shall see in a moment, 16 frames are put together to form a multi-frame which has a time span of 2 msec. This is the basic unit of the PCM system applied to telephony.

Within a time slot there is the encoded information about the speech sample for that channel. In most systems it is coded into 256 ($=2^8$) levels and there are therefore 8 binary digits within each time slot; each digit must be less than 0.48 μsec long. The technique of PCM is given in detail in Section 3.5.

The channel digits bearing speech information must be sent to the right destination and monitored for release and clear-down. Signals must therefore be associated with each channel and in PCM telephony that is done by allocating a signalling word to each channel once per multi-frame. Sufficient information for signalling can be contained in a 4-digit word so that an 8 digit word can contain two signals. Hence a signalling time slot can contain enough information for two signals.

Figure 10.37 indicates that time slot sixteen (TS16) is used to carry the signalling. In the second frame it carries the signals related to speech channels 0 and 15, in the third frame those for channels 1 and 16 and so on until frame 16 when TS16 has signals for speech channels 14 and 29. The next frame is the first of the following multi-frame and the sequence is repeated.

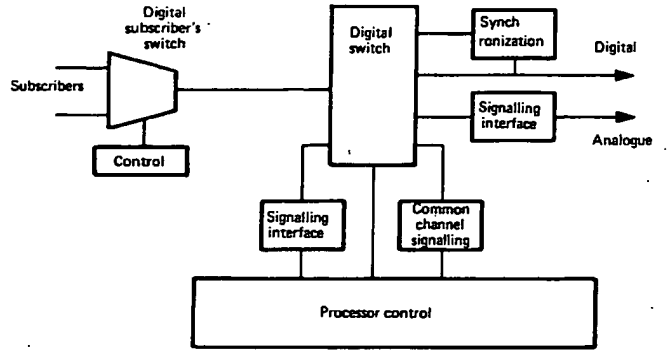


Fig. 10.38 Digital telephone exchange.

switching. However, with digital operation the whole system is considered as an entity.

Following the development of PCM several countries installed PCM links between analogue exchanges. These worked in a basic 24- or 32-channel format with an encoder and decoder. We can see how the 32-channel system is formed with reference to Fig. 10.37. Strictly speaking, it should be referred to as a 30-channel, 32-time-slot system, because two of the time slots contain signalling and synchronizing information, not speech samples. The sampling rate of each speech channel determines the length of each time slot. For telephony the sampling rate used is 8 kHz, which means that the time between adjacent samples of the same channel is 125 μsec. One frame, consisting of 32 time-slots lasts for this time. A time slot is therefore approximately 3.9 μsec long. For reasons that we shall see in a moment, 16 frames are put together to form a multi-frame which has a time span of 2 msec. This is the basic unit of the PCM system applied to telephony.

Within a time slot there is the encoded information about the speech sample for that channel. In most systems it is coded into 256 ($=2^8$) levels and there are therefore 8 binary digits within each time slot; each digit must be less than 0.48 μsec long. The technique of PCM is given in detail in Section 3.5.

The channel digits bearing speech information must be sent to the right destination and monitored for release and clear-down. Signals must therefore be associated with each channel and in PCM telephony that is done by allocating a signalling word to each channel once per multi-frame. Sufficient information for signalling can be contained in a 4-digit word so that an 8 digit word can contain two signals. Hence a signalling time slot can contain enough information for two signals.

Figure 10.37 indicates that time slot sixteen (TS16) is used to carry the signalling. In the second frame it carries the signals related to speech channels 0 and 15, in the third frame those for channels 1 and 16 and so on until frame 16 when TS16 has signals for speech channels 14 and 29. The next frame is the first of the following multi-frame and the sequence is repeated.

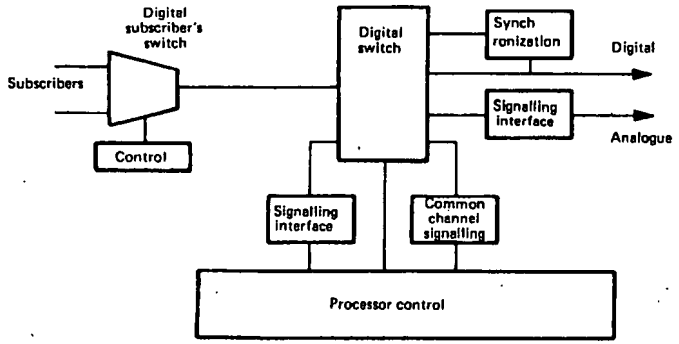


Fig. 10.38 Digital telephone exchange.

TS0 in each frame and TS16 in the first frame, carry synchronization and alignment words to ensure that the transmission and reception of the system is maintained in step.

Modern digital electronics, with its fast logic and complex integrated circuitry, has provided the techniques for producing digital switching and control systems at a cost comparable with analogue, coupled with an ability to provide better facilities, cheaper maintenance and more flexibility. The increasingly widespread use of data in various forms, and the need to send such information over large distances, has also encouraged the development of digital telecommunications and it has led to the intention to integrate together speech and data links.

Digital exchanges consist of the basic components showing in Fig. 10.38. Most of the control of the system is handled by microprocessor devices, singly or in clusters, which are driven by software. Here we are not able to discuss the huge new field of telecommunications software engineering, but it provides the most important challenge in modern system design. The software must be efficient, reliable, secure, understandable and well documented. In theory it affords a degree of flexibility in the operation of the system which is much higher than is possible in hard-wire control. However, such is the complexity of large telephone networks that software development presents the most difficult problems to designers, for it must last for tens of years and although made up of very long programs, it must be able to cope with dramatic changes in hardware technology.

In analogue exchanges the usual figure of merit used is the grade of service, or probability of blocking, but in digital switches the blocking is virtually zero. In these systems the major problems concern delay in the processing of calls caused by the processor units becoming overloaded. The analysis of such problems is very difficult and if simple queueing theory models do not apply resort must be made to computer simulation. One of the difficult tasks of the software engineer is to produce a satisfactory compromise between short efficient programs and those that are longer, more complex and more reliable and secure.

Referring again to Fig. 10.38, look first at the digital switch. It can have many structural forms, depending on the size of the system and the technology used, but as an example we will consider the common time–space–time (TST) configuration. TST is a shorthand for the time-switch, space-switch, time-switch arrangement shown in Fig. 10.39(a).

The time switch is split into several units, each having M PCM links of L channels, as Fig. 10.39(b). Consequently, if the time switch is non-blocking it will have an outlet highway of $N = ML$ time slots. The space switch is square with R inlet highways and R outlet highways. The purpose of the TST unit is to allow a particular call, which occupies a specific channel into one of the time switches, to be connected to a particular outlet channel. Basically the switching is between highways on either side of the space switch. Each highway has N time slots and in order for a particular call to be connected say from $H1$ to $H3$ it must find a time slot which is free in both highways. This slot may not be the same as the required incoming and outgoing slots for the call, and so some time delay, provided by the time switches, is necessary. The method used to produce the delay again depends on

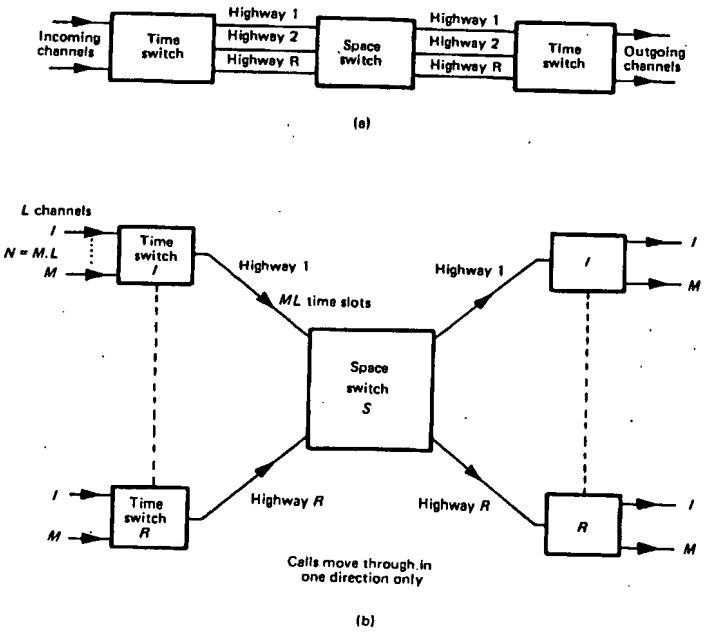


Fig. 10.39 Digital switch: (a) TST block diagram; (b) network representation.

the technology employed, but the delay may be from 1 to 31 time slots depending on the relative positions of the time slots in and out of the unit, and the chosen free time slot in the space switch.

To understand the behaviour of the TST switch in terms of the link systems considered earlier it is important to appreciate that for each time slot the interconnections in the space switch will be different; at each time slot there will be a different set of calls in progress and the connections between the highways will only last for one time slot period then new connections will be established. This can be represented by having N space switches (Fig. 10.40), one for each time slot.

Whether or not blocking occurs in the TST unit depends entirely on the dimensions of the space switch, and since in modern systems switches are comparatively inexpensive they are usually large enough to make blocking negligible. For total non-blocking there must be at least as many outlets as inlets on the time switches, and the space switch highways must have $2N - 1$ time slots, where N is the number of time slots in a link to a time switch.

Digital switches are uni-directional, and that implies that two paths are required to connect two channels X and Y, one for conversation from X to Y and the other

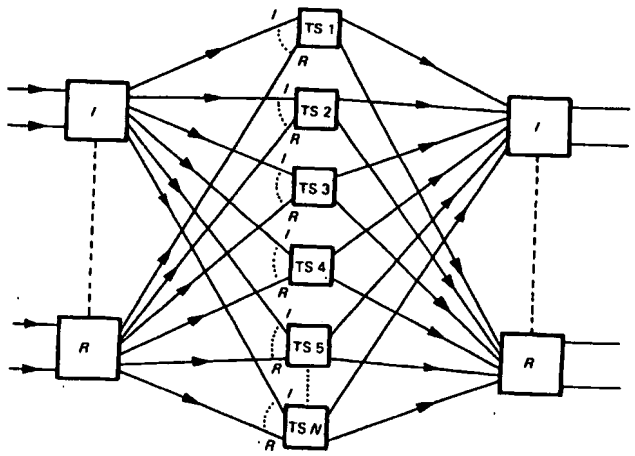


Fig. 10.40 Analogue equivalent of TST switch.

for conversation from Y to X. To reduce the control process, the X to Y slot is chosen according to whatever rules are used by the designer, and the Y to X inter-connection is allocated a fixed number of time-slots from it, e.g. one, or half a frame. By this method, if the X to Y connection is available, the Y to X must also be free.

The digital switch just described is situated in a main exchange, forming part of the trunk network. Interconnections between exchanges are made, for the information paths, via PCM links. However, signalling is carried over a common channel, using signalling system CCITT No. 7 as described in Section 10.3. The inter-exchange signals will not only be concerned with setting up calls between exchanges, but with accounting, administration fault diagnosis and maintenance. As described earlier, the CCITT No. 7 system transmits signals as messages that can have variable length. Each message is preceded by labels that identify its origin and destination exchange, the type of message (call handling, fault, etc.) and includes error-detection and acknowledgement bits. If an error is detected in a message, that and all subsequent messages are retransmitted to ensure that the sequence received at the far end is in the correct order.

The error rate has an important bearing on the capacity of the signalling channel; re-transmission of incorrectly received messages obviously takes up time that could be used for new signals and consequently slows down the overall process. The capacity is specified in terms of number of messages per busy hour given that the delay from end to end is not greater than some predetermined value.

On the subscriber side of the digital switch there will be a local unit of some description. For large areas a local digital exchange would be used with mf signalling from subscriber to exchange where conversion to a PCM format would

take place before concentration through a digital switch. Alternatively for very small units no exchange facility would be available, but a simple digital concentrator would be used to take in the analogue channels, convert them to PCM and multiplex them onto a single highway to the nearest local exchange. Calls between subscribers on the same concentrator would then have to pass through the local exchange. Signalling in these PCM links would be on TS16 and the control, software and firmware at the main exchange would convert it to common channel if a trunk call was required.

The introduction of digital systems is rarely a starting point for the telephone system. Usually a system exists and the digital equipment has to be grafted on to it. Whatever method is used, interworking between the old and new systems is required, and one area of difficulty is the interfacing of various analogue signalling systems with the new equipment designed to operate on TS16 or common channel. This interfacing can be a severe problem if there are many existing signalling schemes in a particular network, and the development of satisfactory units can add considerably to system costs.

10.35 Conclusion

In some respects, the whole concept of telecommunications is changing, and with it the services and functions provided by telecommunications operating companies. The very rapid growth of information technology, with its demand for high-speed, large-capacity, data links, with video output, and the introduction of new facilities on telephone exchanges, are both causing manufacturers to rethink the whole philosophy of how communication systems should best be provided. In the next few years the main point at issue will be to decide whether future systems should be fully integrated, or not. If so, should they be based on a digital telephone system, with its centralized switching units, or on one of the many data networks, with distributed control? Conversely, economy and efficiency may dictate an extension of the present hybrid scheme in which both methods co-exist, with perhaps an element of inter-working between them. It is too early in the development of distributed systems to predict what changes will take place in the next decade, but that major redesign will occur is beyond doubt.

References

1. Brockmeyer, B., Halstron, H.L. and Jensen, A., *The Life and Works of A.K. Erlang*, Copenhagen Telephone Company, 1943.
2. Jacobaeus, C., "A study on congestion in link systems", *Ericsson Technics*, No. 48, 1950.
3. Cooper, R.B., *Introduction to Queueing Theory*, Edward Arnold, London, 1981, Chapter 5.
4. Fry, T.C., *Probability and its Engineering Uses*, Van Nostrand Reinhold, Wokingham, 1965.

User's Guide

Microsoft® Word

The World's Most Popular Word Processor
Version 6.0

Microsoft Corporation

Document No. WB51157-1093
Printed in Ireland :09

Protecting Documents from Changes

Word provides several ways to restrict changes to documents. You can assign a password to prevent other users from opening a document or to keep others from saving changes to the document. You can also request or require that other users on a network open a document as read-only.

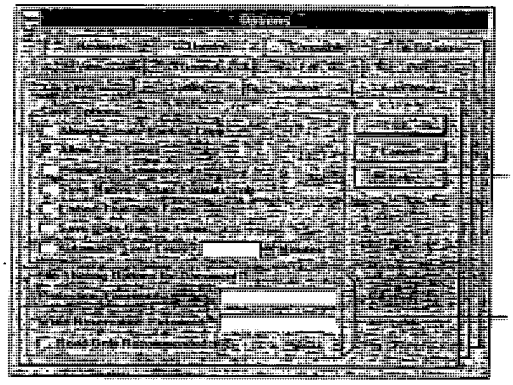
You can also assign a password so that other users can annotate a document and mark revisions. You or someone else who knows the password must open the document normally and review the changes before they become permanent. For more information, see Chapter 25, "Annotating, Revising, and Routing Documents."

If you use form fields to create a form, you can assign a password so that other users can fill in those parts of the form but cannot change anything else in the document. For more information, see Chapter 14, "Forms."

Warning If you assign a password for any of these types of protection, it's a good idea to write it down. Without the password, you cannot open the document.

Setting Passwords and Selecting Save Options

To assign a password to a document and set options that control whether changes can be saved, choose the Options button in the Save As dialog box or choose Options from the Tools menu, and then select the Save tab.



Protection Password To prevent other users from opening a document, type a password in the Protection Password box. Only users who know the password can open the document. Passwords are case-sensitive.

Write Reservation Password To prevent other users from saving changes to a document, type a password in the Write Reservation Password box, and then choose the OK button. Word will prompt you to type the password again to confirm it. Word then requires you to type the password to open the document normally. If you do not know the password, you can still open the document as read-only by choosing the Read Only button in the Password dialog box that appears when you open the document.

Read-Only Recommended To recommend, but not require, that other users open a document as read-only, select the Read-Only Recommended check box. When another user opens a document that's protected by this option, Word indicates that the document should be opened as read-only unless changes need to be saved. The user can then open the document normally or as a read-only document.

▶ **To protect a document with a password**

1. Open the document you want to protect with a password.
2. From the File menu, choose Save As.

   If you have not yet named the document, type a name in the File Name box.
3. Choose the Options button.
4. In the Protection Password box or the Write Reservation Password box, type a password, and then choose the OK button.

   A password can contain up to 15 characters and can include letters, numbers, symbols, and spaces. As you type the password, Word displays an asterisk (*) for each character you type. Note that passwords are case-sensitive.
5. When Word prompts you to confirm the password, retype it and then choose the OK button.
6. To save the document, choose the OK button.

   Make sure that you write down the document password, exactly as you typed it. You will need to type it the next time you open the document.

Tip If you want to allow other users to add only comments to a document, you can protect it by using the Protect Document command on the Tools menu. Other users can then open the document, but they can only make comments by using annotations.

---

cument, type a
v the password can

g changes to a
box, and then
ord again to
ı the document
he document as
alog box that

other users open a
eck box. When
Vord indicates that
:d to be saved. The
:ument.

File Name box.

sword box, type a

letters, numbers,
ys an asterisk (*)
ısitive.

ınd then choose

:tly as you typed
ent.

document, you
ools menu. Other
ıents by using

▶ **To change or delete a password**

1. Open the document whose password you want to change or delete.
2. From the File menu, choose Save As.
3. Choose the Options button.
4. In the Protection Password box or the Write Reservation Password box, select the row of asterisks that represents the existing password, and then do one of the following:
   - To change the password, type the new password.
   - To delete the password, press DELETE.
5. Choose the OK button.

   If you changed the password, Word asks you to retype the new password.
6. To save the document with the new password, choose the OK button.

Other Ways of Protecting Documents

Word offers other methods of protecting your documents.

For information about	See
Opening documents as read-only	"To open an existing document," earlier in this chapter
Preventing any changes to documents except for filling in form fields	Chapter 14, "Forms"
Preventing any changes to documents except for annotations and marked revisions	Chapter 25, "Annotating, Revising, and Routing Documents"

Note Protecting a form or locking a document for annotations or revisions does not keep another user from saving that document with a password or from setting other save options. If you want to protect a document from all types of changes, save it with a password by using one of the methods described in this chapter.

Some operating systems and networks also provide ways to protect documents. To find out if your system has these features, check with your network administrator or see the documentation for your operating system or network.

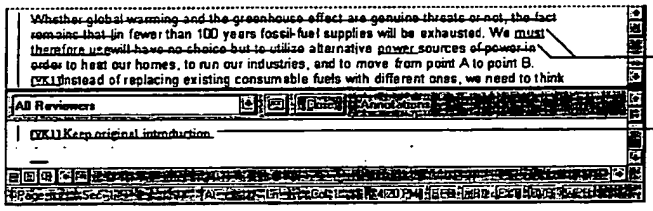
CHAPTER 25

Annotating, Revising, and Routing Documents

Word provides three features that make it easy to revise documents and incorporate comments from reviewers: annotations, revision marks, and routing.

To comment on a document rather than change it, use *annotations*—numbered comments added in a separate annotation pane. To make changes directly in the document, use *revision marks*. Revision marks show where text or graphics have been added, deleted, or moved. In addition, revision marks allow you to track changes by reviewer, date, and time.

You can use Word with Microsoft Mail or other compatible mail packages to send an online copy of a document to recipients who can comment on or add to the document. You can send multiple copies of the document to all reviewers at the same time, or you can route a single copy sequentially through a list of reviewers.

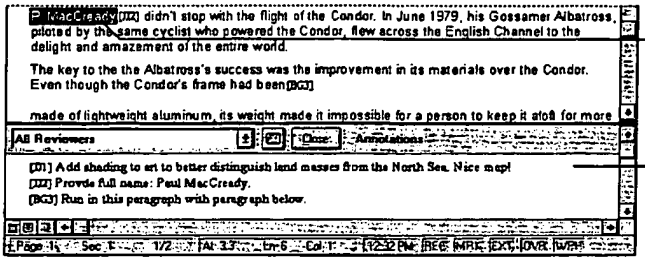


In This Chapter

QUICK START

Inserting Annotations

To comment on a document but not change the content, use annotations. To insert an annotation, select the text you want to comment on. From the Insert menu, choose Annotation. Word opens a separate annotation pane where you type your comments. To insert additional annotations, follow the same procedure, or select the text and press ALT+CTRL+A (Windows) or COMMAND+OPTION+A (Macintosh). When you finish, choose the Close button in the annotation pane.

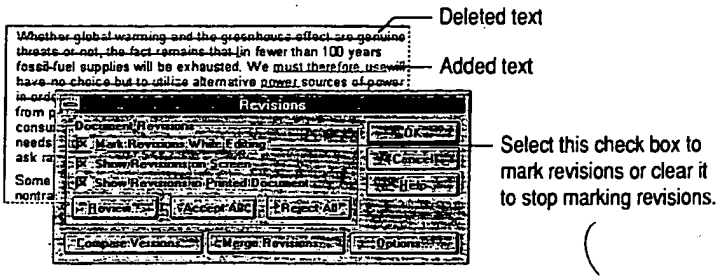


Viewing Annotations

To view annotations, choose Annotations from the View menu. The annotation pane displays comments from all reviewers. To view annotations from a single reviewer, select that person's name from the box at the top of the annotation pane. To view the range of text a particular annotation refers to, position the insertion point within the comment in the annotation pane. Word highlights the text the reviewer selected. When you finish viewing annotations, choose the Close button.

Marking Revisions

Use revision marking to track the changes that you or other reviewers make to a document. To turn on revision marking, choose Revisions from the Tools menu or double-click "MRK" in the status bar. Select the Mark Revisions While Editing check box, and then choose the OK button. From that point on, Word tracks all revisions. By default, the revision marks are displayed on the screen and in the printed document.



tations. To insert
Insert menu,
e you type your
edure, or select
I+A (Macintosh).

elected text that
ou want to
omment on

nnotation pane
here you type
omments

The annotation
s from a single
e annotation pane.
on the insertion
ts the text the
the Close button.

ewers make to a
the Tools menu or
s While Editing
Word tracks all
reen and in the

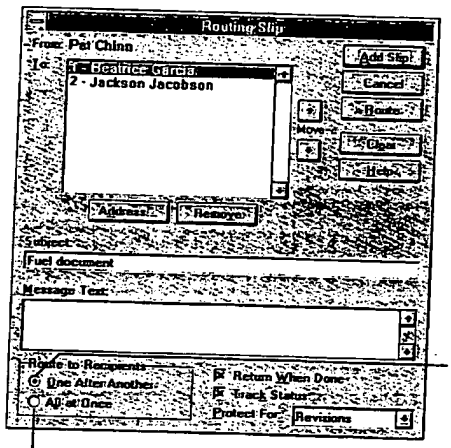
ox to
lear it
isions.

Hiding Revision Marks

If you want Word to track changes without displaying the revision marks on the screen, choose Revisions from the Tools menu. Then select the Mark Revisions While Editing check box, and clear the Show Revisions On Screen check box. To hide revision marks when printing, clear the Show Revisions In Printed Document check box, and then choose the OK button.

Routing a Document Online

You can use Word with Microsoft Mail or another compatible mail package to send a document to others. Choose Add Routing Slip from the File menu, and then choose the Address button. Select the names that you want to route the document to, choose the Add button, and then choose the OK button. Under Route To Recipients, select the distribution method. To send the document, choose the Route button. Recipients return their annotated or revised copies by using the Send command on the File menu.



To route the document to reviewers one after another, click here.

To route the document to all reviewers at once, click here.

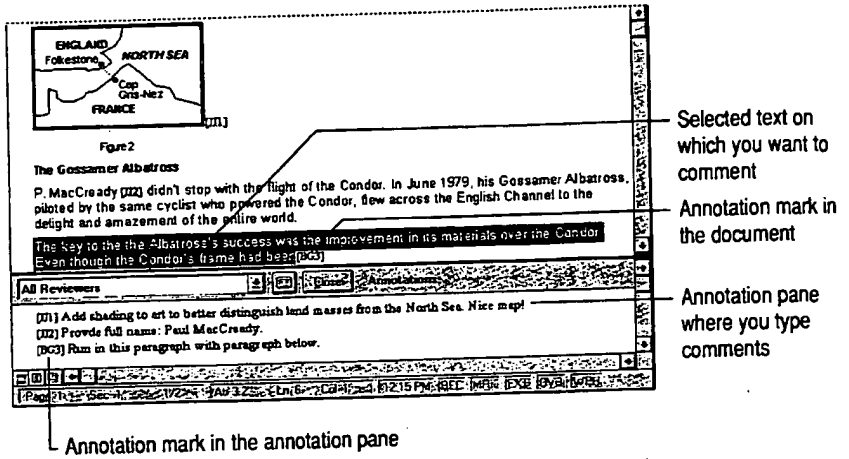
See the following pages for detailed information.

Using Annotations

When you want reviewers to comment on a document rather than make changes directly to it, have them use *annotations*. Annotations are numbered comments added in a separate annotation pane. Reviewers can include text as well as graphics in the annotation pane. With the appropriate hardware, voice and pen annotations can also be included.


Show/Hide ¶ button

Each annotation has an *annotation mark* that includes the reviewer's initials and a sequential number. Word obtains the reviewer's initials from the User Info tab in the Options dialog box (Tools menu). In the annotation pane, the annotation mark appears as normal text. In the document window, the annotation mark appears as hidden text. (You can view annotation marks in the document when the annotation pane is closed by clicking the Show/Hide ¶ button on the Standard toolbar.)



For information about routing online documents to multiple reviewers, see "Routing a Document Online," later in this chapter.

Inserting Annotations

Before you insert an annotation, it's a good idea to select the text or item that you want to comment on. That way, Word can later highlight the text or item to which the annotation refers. When you type an annotation, you can use the toolbars, the ruler, and commands on the Format menu to apply formatting to text in the annotation pane.

▶ To insert an annotation

1. Select the text or item you want to comment on, or position the insertion point at the end of the text or item you want to comment on.
2. From the Insert menu, choose Annotation.

   Word inserts an annotation mark (initials and a number formatted as hidden text) in the document and opens the annotation pane.

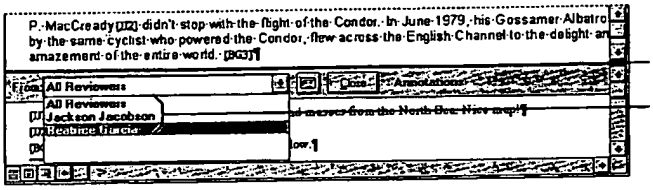
   Word uses the reviewer's initials from the User Info tab in the Options dialog box (Tools menu).
3. Type the annotation text in the annotation pane, and then do one of the following:
   - To close the annotation pane and return to the document, choose the Close button.
   - To keep the annotation pane open to add additional annotations, click in the document window, and repeat steps 1 through 3.

Using the keyboard You can also quickly insert annotations by using the keyboard. Press ALT+CTRL+A in Windows and COMMAND+OPTION+A on the Macintosh.

Tip Annotation marks are automatically displayed when the annotation pane is open. If the annotation pane is closed, you can display annotation marks and nonprinting characters in the document by clicking the Show/Hide ¶ button on the Standard toolbar. Click the button again to hide annotation marks.

Viewing and Locating Annotations

When the annotation pane is open, it shows the annotations that correspond to the part of the document displayed in the document window. As you scroll through the document, the annotation pane scrolls as well. You can change the size of the annotation pane by dragging the split box up or down on the vertical scroll bar.



▶ **To view annotations**

1. Do one of the following:
   - From the View menu, choose Annotations.
   - Double-click an annotation mark in the document window. If you don't see annotation marks, click the Show/Hide ¶ button on the Standard toolbar.
2. To view the annotations of a single reviewer, select the name of the reviewer from the Reviewers box at the top of the annotation pane. The default is All Reviewers, which displays the annotations of all reviewers.
3. When you finish viewing annotations, do one of the following:
   - To close the annotation pane and return to the document, choose the Close button.
   - To keep the annotation pane open and return to the document, click in the document window.

Note If a reviewer selected text before inserting an annotation, the selected text is highlighted when you position the insertion point within the corresponding annotation in the annotation pane. Text that is highlighted is not selected. To quickly select the highlighted text so that you can modify it, press ALT+F11 (Windows) or OPTION+F11 (Macintosh).

To locate a specific annotation, choose Go To from the Edit menu. Under Go To What, select Annotation. If you know which reviewer made the annotation, select the reviewer's name from the list. Choose the Next button until you find the annotation you want.

Incorporating and Deleting Annotations

To incorporate the text of an annotation into a document, select the annotation text or item, and then choose Copy from the Edit menu. Position the insertion point in the document where you want the text or item to appear, and then choose Paste from the Edit menu.

To delete an annotation, select the annotation mark in the document window and then press BACKSPACE or DELETE. Word automatically renumbers annotation marks each time you add, delete, or copy an annotation.

Printing Annotations

You can print annotations either separately from the rest of the document or with the document. If you print annotations only, Word prints the page number of the annotation mark, the reviewer's initials, the annotation number, and then the annotation text. If you print annotations with the document, Word prints this same information at the end of the document. Word prints hidden text in the document so that you can see the location of the annotation marks.

▶ **To print annotations only**

1. From the File menu, choose Print.
2. In the Print What box, select Annotations, and then choose the OK button.

▶ **To print a document with annotations**

1. From the File menu, choose Print.
2. Choose the Options button.
3. Under Include With Document, select the Annotations check box, and then choose the OK button.
4. In the Print dialog box, select any other printing options you want, and then choose the OK button.

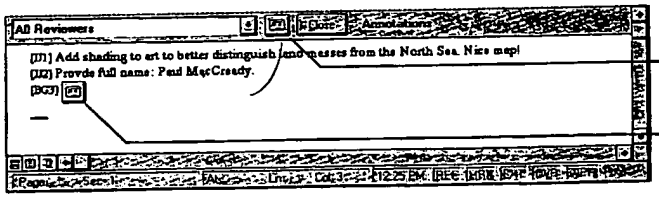
Inserting and Listening to Voice Annotations

If your computer runs Windows-based applications, you must have a sound board installed to listen to voice annotations. To record annotations, you must also have a microphone.

If you have a Macintosh computer, you can listen to voice annotations without having special sound equipment. To record annotations on a Macintosh, you must have a microphone.

You can distinguish a voice annotation from other annotations by the sound symbol—an audio cassette tape—which Word inserts in the annotation pane and in the document at the position of the insertion point.

Tip To insert a combination of text and voice annotation, insert the text annotation first. Then position the insertion point to the right of the annotation mark in the document and insert the voice annotation.



Note You can customize the marks Word uses to show document differences. For more information, see "Customizing Revision Marks," earlier in this chapter.

▶ **To compare two versions of a document**

1. Open the edited version of the document.
2. From the Tools menu, choose Revisions.
3. Choose the Compare Versions button.
4. In the Original File Name box, type or select the name of the original document, and then choose the OK button.

   Word displays the edited document marking inserted, deleted, and revised text with revision marks. The options for displaying revision marks are set on the Revisions tab in the Options dialog box (Tools menu).
5. To accept or reject the revisions, choose Revisions from the Tools menu. For more information, see "Incorporating Revisions," earlier in this chapter.

Protecting a Document for Annotations and Revisions

For more information on ways to protect a document, see Chapter 21, "Opening, Saving, and Protecting Documents."

To allow reviewers to comment on but not make changes to a document, you can protect it for annotations. To allow reviewers to change a document and keep a record of all changes, you can protect it for revisions.

For maximum protection, you should also use a password when you protect a document for annotations or revisions. Otherwise, anyone can remove protection from the document by choosing Unprotect Document from the Tools menu.

▶ **To protect a document for annotations or revision marks**

1. Open the document you want to protect.
2. From the Tools menu, choose Protect Document.
3. Do one of the following:
   - To allow reviewers to insert annotations but not change the contents of the document, select the Annotations option button.
   - To track revisions, select the Revisions option button. The reviewers cannot turn off revision marking, and revisions cannot be accepted or rejected.
4. To ensure that a document is protected against untracked changes, type a password. This prohibits anyone who does not know the password from unprotecting the document.
5. Choose the OK button.

▶ **To review and incorporate revisions**

1. From the Tools menu, choose Revisions.
2. Choose the Review button.
3. Do one of the following:
   - To move to a revision mark, click the appropriate Find button to search forward or backward in the document.
   - Click a revision mark in the document.

   For each selected revision mark, Word displays the reviewer's name, and the date and time the revision was entered.
4. Do one or more of the following:

To	Choose this button
Accept a revision	Accept
Reject a revision	Reject
Leave the revision mark unchanged and move to the next or previous revision	Find ➔ or ➔ Find
Undo the last acceptance or rejection of a revision	Undo Last

Note To automatically move to the next revision mark while reviewing the revisions, select the Find Next After Accept/Reject check box.

▶ **To accept or reject all revision marks**

1. From the Tools menu, choose Revisions.
2. Do one of the following:
   - To accept all revisions, choose the Accept All button.
   - To reject all revisions, choose the Reject All button.

   Word displays a message asking you to confirm that you want to accept or reject all revisions.
3. Choose the OK button.

Comparing Versions of a Document

To compare two versions of a document, you use the Compare Versions button in the Revisions dialog box. During the compare process, Word inserts revision marks that you can review and incorporate as described earlier in "Incorporating Revisions."

Make sure that the two documents you are comparing have different filenames, or—if they have the same name—that they are in different directories.

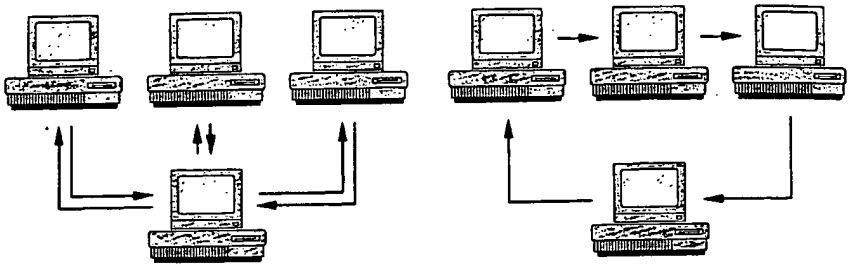
▶ **To unprotect a document for annotations or revision marks**

- From the Tools menu, choose Unprotect Document.

  If the author has protected the document with a password, you must know the password to unprotect the document.

Routing a Document Online

You can use Word and Microsoft Mail or a compatible mail program to route documents online. For example, you might want others to review an important memo before sending it out, or you might want several people to complete an online questionnaire or form.

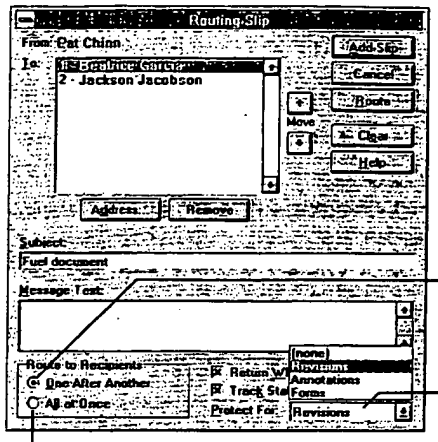


You can route a document to all reviewers at once ...

or you can route it to one reviewer after another.

You can route online copies in two ways. You can send a separate copy to all reviewers at the same time, or you can send a single copy that goes to each person on the list in turn, allowing each reviewer to see the comments of all previous reviewers.

Reviewers return their annotated or revised copies to you or the next person on the distribution list by choosing the Send command from the File menu. When all the copies have been returned, you can merge the annotations and revisions into the original document to simplify review of the comments. For more information on merging comments, see "Merging Annotations and Revisions," later in this chapter.



▶ To route a document to others

1. Open the document you want to route.
2. From the File menu, choose Add Routing Slip.
3. Choose the Address button. Select the names of the people to whom you want to route the document, choose the Add button, and then choose the OK button.

   If you want to route the document to one recipient after another, use the Move up and down arrows to put the names in the correct routing order.
4. In the Subject and Message Text boxes, type the subject and any message or instructions you want to send with the document. Each recipient will receive the same subject and message.

   Word automatically appends instructions to your message telling recipients to choose the Send command when they are finished.
5. Under Route To Recipients, do one of the following:
   - To route one copy of a document to one recipient after another, select the One After Another option button.
   - To route multiple copies of a document to all recipients at the same time, select the All At Once option button.

6. Select any other options you want, and then choose the Route button.

   If you want to continue to edit the document before you route it, choose the Add Slip button, and continue to edit the document. When you are ready to send the document, choose Send from the File menu. Word displays a message asking you to confirm that you want to route the document.

   The document is sent to the distribution list as an attached Word file. The recipients can add annotations or revisions to the document and then return the copy to you by choosing the Send command on the File menu.

   If the document is being routed to one recipient after another, the Send command automatically routes it to the next person on the list before it returns to you. You will receive all the recipients' comments in one document after it has been routed to the last person on the list.

   If you send the document to all recipients at the same time, you will receive multiple copies of the document. You can then merge all changes into one document. For more information, see the following section.

Merging Annotations and Revisions

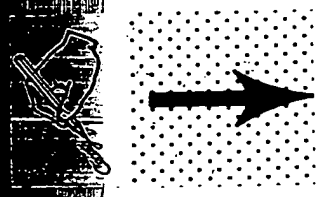
If you have given individual copies of a document to multiple reviewers, you can combine their annotations and revisions into the original document. When you merge annotations and revisions, any annotations and revisions already in the original document are preserved as additional comments are merged. Word assigns a different color to each reviewer. If there are more than eight reviewers, Word uses the colors again, so some reviewers may share the same color.

Note Annotations and revisions cannot be merged back to the original document unless they are marked. To ensure that revisions to a document are marked, you should protect the document for revisions or annotations before making the revisions. For information on protecting documents, see "Protecting Documents for Annotations and Revisions," earlier in this chapter.

▶ **To merge revision marks and annotations**

1. Open the document that has revisions you want to merge into the original document.
2. From the Tools menu, choose Revisions.
3. Choose the Merge Revisions button.

Opening a Document Created in Another Application



To convert documents to and from different file formats, you must install the appropriate *converters*. To import and export graphics contained in documents, you must install *graphics filters*. If you performed a complete installation when you installed Word, converters and graphics filters were also installed. If not, you can run the Microsoft Word Setup program again. For instructions, double-click the Help button on the Standard toolbar, and then type **setup**

Converters change the file format of a document, and they use graphics filters to import and export graphics that are within, or linked to, a document. For information about converting graphics files in such formats as TIFF and PICT, and for a list of supplied graphics filters, see Chapter 16, "Importing and Creating Graphics."

If you want to import data from a database into a Word document, you have three options. To convert an entire database file from database applications that Word can convert, such as Microsoft Excel, use the procedure under "Opening a Document," later in this chapter. To use information from a database to print form letters, mailing labels, and other types of mail merge documents, see Chapter 29, "Mail Merge: Step by Step," and Chapter 30, "Mail Merge: Advanced Techniques." To insert information from a database into a Word document as a table, see Chapter 28, "Exchanging Information with Other Applications."

To convert several documents at once, use the batch conversion macro BatchConversion in the CONVERT.DOT (Windows) or Conversion Macros (Macintosh) template. For information about using this macro, see "Converting Several Documents at Once," later in this chapter.

Supplied Converters

Word provides converters for the applications in the following list and for several plain-text file formats. For information about specific converters, and for instructions on how to obtain supplemental converters and graphics filters that were not shipped with Word, double-click the Help button on the Standard toolbar, and then type **readme**. Press ENTER twice, and then click File Conversion.

Converters Supplied with Word for Windows

Microsoft Word for Windows

Microsoft Word 4.*x* and 5.*x* for the Macintosh

Microsoft Word 3.0–6.0 for MS-DOS

WordPerfect 5.*x* for MS-DOS and Windows

Microsoft Write for Windows
Microsoft Excel BIFF 2.*x*, 3.0, 4.0*, and 5.0
Lotus 1-2-3 2.*x* and 3.*x**
RFT-DCA

* The converter can open, but not save, documents in this file format.

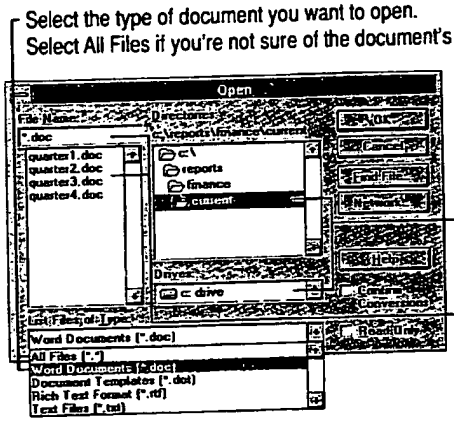
Converters Supplied with Word for the Macintosh

Microsoft Word for Windows
Microsoft Word 3.*x* for the Macintosh*
Microsoft Word 4.*x* and 5.*x* for the Macintosh
Microsoft Word 3.0–6.0 for MS-DOS
Microsoft Works 2.0 for the Macintosh
WordPerfect for MS-DOS and Windows 5.*x*
MacWrite®
MacWrite II 1.1
Microsoft Excel BIFF 3.0, 4.0*, and 5.0
RFT-DCA

* The converter can open, but not save, documents in this file format.

Opening a Document

To convert most types of documents, simply open them. Word automatically opens a copy of the document and converts the copy to Word format.



Most of the time, Word recognizes the file format, converts the document, and then displays it in a Word window. If Word cannot recognize the format, it displays the Convert File dialog box. Select the appropriate file format, and then choose the OK button. Word converts and opens the document.

If a document isn't converted correctly, close it without saving changes and then try converting again using a different converter. The original document remains unchanged until you save it in Word format or in some other format. You can also change the way Word converts some items. For more information, see "Customizing Conversions and Improving Compatibility," later in this chapter.

▶ To open a file created in another application


Open button

1. On the Standard toolbar, click the Open button.
2. In the List Files Of Type box, select the type of file you want to open. If you do not know the type of document or file format, select All Files.
3. In the File Name box, type or select the document you want to open.

   If the document you want to open does not appear in the File Name box, select a different drive, directory, or folder.
4. Choose the OK button to convert a copy of the document to Word format.

If the converter you need was not installed when you installed Word, Word displays the Convert File dialog box and asks you to choose a converter. To install additional converters, you need to run the Microsoft Word Setup program again. For online instructions, double-click the Help button on the Standard toolbar, and then type **setup**

Using the List Documents Of Type Box (Windows)

If you don't see the document you want to open in the File Name list, try selecting one of the options in the List Files Of Type box; these options are described in the following table. If you don't know the extension of the document you're looking for, or if it doesn't have an extension, select All Files.

Select	To display
All Files (*.*)	All documents in the selected directory.
Word Documents (.doc)	Word documents and other documents with the .DOC filename extension.
Document Templates (.dot)	Word templates with the .DOT filename extension.
Rich Text Format (.rtf)	Documents with the .RTF filename extension. For more information, see "Opening or Saving a Plain-Text File," later in this chapter.
Text Files (.txt)	Documents with the .TXT filename extension. Includes documents saved as ASCII text or MS-DOS text, including Text Only, Text With Line Breaks, and Text With Layout. For more information, see "Opening or Saving a Plain-Text File," later in this chapter.

locument isn't converted correctly, close it without saving changes and then
onverting again using a different converter. The original document remains
anged until you save it in Word format or in some other format. You can also
ge the way Word converts some items. For more information, see
tomizing Conversions and Improving Compatibility," later in this chapter.

pen a file created in another application

)n the Standard toolbar, click the Open button.

n the List Files Of Type box, select the type of file you want to open. If you
lo not know the type of document or file format, select All Files.

n the File Name box, type or select the document you want to open.

f the document you want to open does not appear in the File Name box, select
different drive, directory, or folder.

Choose the OK button to convert a copy of the document to Word format.

ie converter you need was not installed when you installed Word, Word
lays the Convert File dialog box and asks you to choose a converter. To
all additional converters, you need to run the Microsoft Word Setup program
n. For online instructions, double-click the Help button on the Standard
bar, and then type **setup**

ng the List Documents Of Type Box (Windows)

ou don't see the document you want to open in the File Name list, try selecting
of the options in the List Files Of Type box; these options are described in the
owing table. If you don't know the extension of the document you're looking
or if it doesn't have an extension, select All Files.

ct	To display
Files (*.*)	All documents in the selected directory.
d Documents (.doc)	Word documents and other documents with the .DOC filename extension.
ument Templates (.dot)	Word templates with the .DOT filename extension.
Text Format (.rtf)	Documents with the .RTF filename extension. For more information, see "Opening or Saving a Plain-Text File" later in this chapter.
t Files (.txt)	Documents with the .TXT filename extension. Includes documents saved as ASCII text or MS-DOS text, including Text Only, Text With Line Breaks, and Text With Layout. For more information, see "Opening or Saving a Plain-Text File," later in this chapter.

In Windows, Word lists documents according to their three-character filename extension. For example, Word documents usually end with .DOC, and plain-text files end with .TXT. When you select a file type from the List Files Of Type box, Word inserts the appropriate filename extension in the File Name box. You can also type extensions directly in the box to see a list of documents from a specific application. For example, type ***.xls** to see a list of Microsoft Excel documents, or type ***.wps** to see a list of word-processing documents from Microsoft Works. To see a list of both types at once type ***.xls; *.wps**. After typing the extensions, press ENTER to display the list.

Note The file format does not necessarily correspond to the filename extension. For example, a WordPerfect document may have a .DOC, .TXT, or .WPS extension, or no extension at all. When you open a file in another format, Word first looks at the contents of the file to determine the file format. If Word doesn't recognize the file format, it tries to use the converters that correspond to the filename extension. If Word is still unable to recognize the file format, it asks you to choose a converter and suggests Text Only.

Using the List Documents Of Type Box (Macintosh)

If you don't see the document you want to open in the File Name box, make sure that you have selected the correct file type for the document in the List Files Of Type box. If you still don't see the document name listed, select either Readable Files (to display files Word can read with the installed converters) or All Files (to display every document in the selected folder, regardless of file type).

Saving a Converted Document

Once you have converted a document from another file format to a Word document by opening it in Word, the converted document resides only in the computer's memory; the original document remains in its original format on disk. When you save the document, Word asks you whether you want to save it in Word format or in the document's original format. If you want to retain copies in both formats, use the Save As command on the File menu and give the document a different name (or filename extension, which is usually used to denote the file format). The document will be saved with the new name in Word format.

For more information, see "Saving and Closing Converted Documents," later in this chapter.

ır on my screen instead of the text effects, equations, or charts that I

.ve selected the Picture Placeholders option. From the Tools menu, ons, and then select the View tab. Make sure that the Picture s check box is cleared.

ıle-click a text effect created in WordArt or an equation I created with itor, the menus and toolbars don't change. Instead, I see the WordArt or itor window.

ıt display the WordArt or Equation Editor toolbars and menus if you :d the Picture Placeholders option. From the Tools menu, choose d then select the View tab. Make sure that the Picture Placeholders s cleared. Also make sure that the zoom setting in the Zoom Control Standard toolbar is 100 percent.

ıe size of a WordArt text effect, equation, or chart. When I double-click it, :turns to its original size.

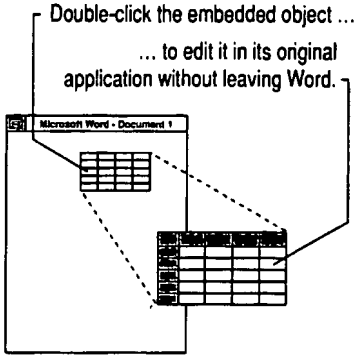
ıde for the object may contain an `\s` switch that tells Word to preserve original size. To remove the switch, choose Options from the Tools :t the View tab, and then select the Field Codes check box. If the field like this: `{EMBED WordArt \s \*mergeformat}`, delete the `\s` ıu to the Options dialog box and clear the Field Codes check box.

CHAPTER 28

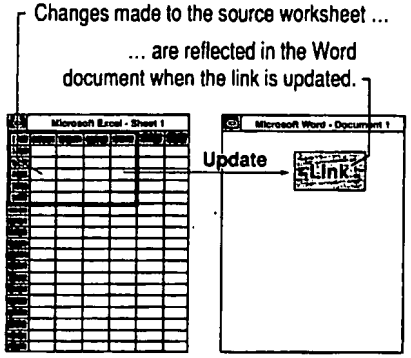
Exchanging Information with Other Applications

For online instructions, double-click the Help button on the Standard toolbar. Then type **embedding** or **linking** or **publishing**

With the linking and embedding features available in Word, you can include information, or *objects*, created in other applications. The main difference between linking and embedding is where the data is stored. Embedded objects become part of the Word document itself. Linked objects, on the other hand, are stored in the source file; the Word document stores only the location of the source file but displays a representation of the linked data.



Worksheet embedded in the Word document itself



Worksheet in a separate file linked to the Word document

If you are using Word for the Macintosh with System 7 or later, you can also use Publish and Subscribe to link a Word document to other files.

In This Chapter

QUICK START

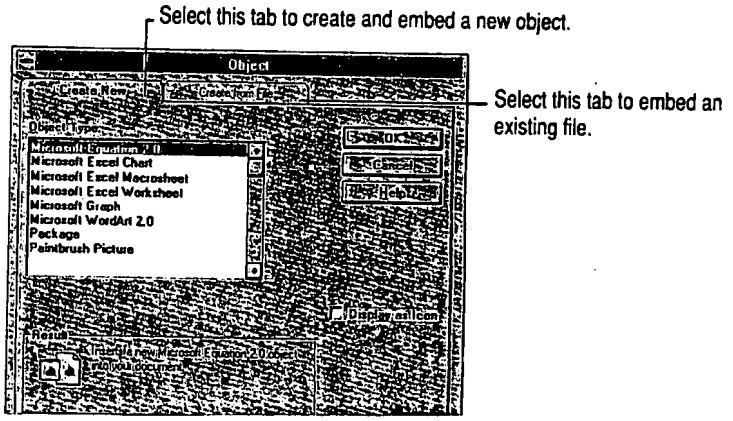
Should I embed an object or link it?

Use these guidelines to decide between embedding and linking information.

When you want to	Use this method	Comment
Include information that becomes part of the Word document and is always available, even if the original source file or the Word document is moved.	Embed the objects from another application in a Word document.	To edit the objects, all relevant applications must be installed on the computer you are using.
Include data maintained in a separate file; the Word document reflects any changes made to that file.	Create a link in the Word document to the source file.	
Include a very large file, such as a video clip or sound clip	Create a link in the Word document to the source file.	Word can store just the link; this keeps the size of the Word document manageable.
Include a file that may not always be available, such as a file stored on a network server	Embed the file from another application in a Word document.	

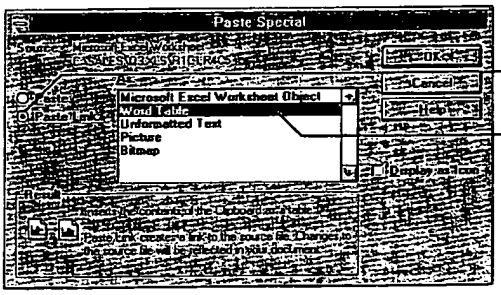
Embedding an Object

Position the insertion point where you want to embed the object. From the Insert menu, choose Object. Then, to create and embed a new object, select the Create New tab; select the type of object you want, and then choose the OK button. To embed an entire existing document, select the Create From File tab; select the filename, and then choose the OK button.



Creating a Link

Start an application that supports OLE or DDE, and then open the source file. Select the information to which you want to create a link, and then choose Copy from the source application's Edit menu. In the Word document, position the insertion point where you want the linked object, and then choose Paste Special from the Edit menu. Select the Paste Link option button, select the type of linked object in the As box, and then choose the OK button.



The Paste Special dialog box

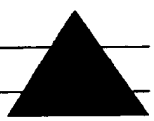
Editing an Embedded Object

Double-click the object to edit it. Some applications open a separate window for editing. Others temporarily replace some of the Word menus and toolbars with those of the source application. If the application opens a separate window, you return to Word by choosing Exit from the application's File menu. If the application replaces the Word menus and toolbars, you return to Word by clicking outside the embedded object.

Editing a Linked Object

To edit the linked object itself, select the object and then choose the object's name from the Edit menu. For example, if you select a link to a Microsoft Excel worksheet, the command on the Edit menu is Microsoft Excel Worksheet object. When you choose the command and then choose Edit from the submenu that appears, the source application opens for editing. To update, reconnect, or break a link, choose Links from the Edit menu, make the changes, and then choose the OK button.

See the following pages for detailed information.



Embedding Objects

For an example of embedding an object, see "Examples of Embedding and Linking," later in this chapter.

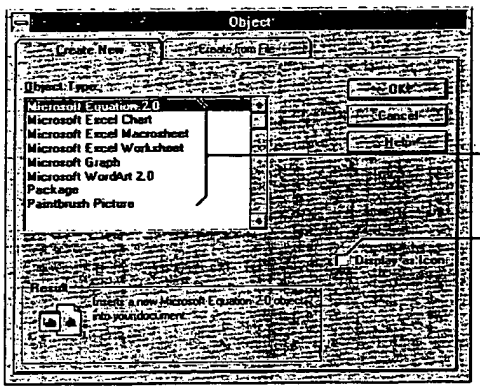
Embedding means inserting information, such as a chart, an equation, or spreadsheet data, in a Word document. Once embedded, the information—called an *object*—becomes part of the Word document. When you double-click an embedded object in Word, you open the application in which the object was created; the object is displayed and ready for editing. When you return to Word, any changes you've made to the object are reflected in the Word document.

Without leaving Word, you can embed an existing file or create and embed a new object. An embedded object increases the file size of a Word document, because the object is stored in the Word document. For example, if you embed a Microsoft Excel worksheet in a Word document, the file size of the Word document increases by approximately the file size of the worksheet. The file size increases even if you display the object as an icon in the Word document, because Word still stores the information about the file.

Creating an Embedded Object

For more information, see "Embedded Objects and Links Are Represented by Fields," later in this chapter.

Suppose you want to create a chart in a Word document. Position the insertion point where you want the chart, and then choose Object from the Insert menu. Word displays a list of the types of objects you can create and embed. Select Microsoft Graph in the Object Type box, and then choose the OK button. Word opens Microsoft Graph, in which you can create a chart. When you quit Microsoft Graph, the chart is displayed in your Word document. When you want to edit the chart, double-click it to start Microsoft Graph; you can then make changes to the chart.



Select the type of object you want to embed.

Select this check box to display the embedded object as an icon in the Word document.

You use the same process to create any other type of object, such as an equation, a Microsoft Excel worksheet, or even a Word object. The only difference is in the type of object you select in the Object dialog box. Whatever type you select, Word opens the appropriate application for creating the object.

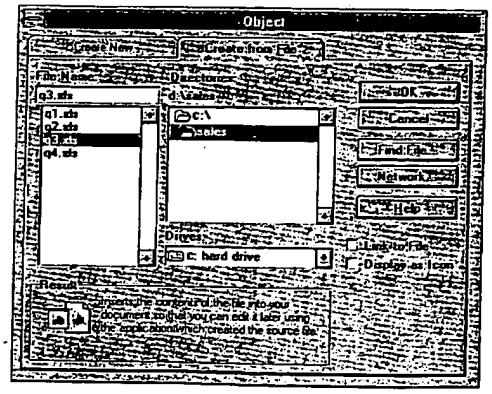
Note The applications you use to create embedded objects must have been properly installed by using their original installation programs; otherwise, they may not be listed in the Object dialog box. Supplemental applications supplied with Word (WordArt, Equation Editor, and Graph) must be installed by using the Microsoft Word Setup program.

In addition to creating embedded objects as you work, you can embed existing files or parts of files. If you embed an existing file, Word stores an independent copy of the file in the Word document. The original file remains unchanged, even if you change the embedded file. Changes to the original file don't affect the Word document.

Microsoft Excel icon

You can display an embedded object as an icon by selecting the Display As Icon check box in the Object dialog box when you insert the object. You might want to use this option when the Word document will be read on-screen and the embedded object contains supplementary information. For example, if the Word document is a summary of financial data from several Microsoft Excel worksheets, you might want to embed one of the worksheets as an icon next to the paragraph in the Word document analyzing that portion of the data. If readers want to see the data, they can double-click the icon to open the worksheet. If you print the document, the icon is printed at its current position in the Word document. Some objects created from a file—for example, ASCII text files—are always displayed as icons. You can change the icon by choosing the Change Icon button, which appears when you select the Display As Icon check box.

Note If the source application supports drag-and-drop editing, you can embed an object by selecting information in the source application and then dragging it into the Word document.



The Object dialog box (Create From File tab)

▶ To embed an object

1. Position the insertion point where you want to embed the object.
2. From the Insert menu, choose Object.
3. Do one of the following:
   - To embed a new object, select the Create New tab. In the Object Type box, select the type that describes the application in which you want to create the object, and then choose the OK button.

     The contents of the list depend on which applications installed on your computer support linking and embedding.
   - To embed an existing file, select the Create From File tab. In the File Name box, type or select the name of the file you want to embed, and then choose the OK button.

     If you do not see the file that you want to embed, select a different drive, directory, or folder, or choose the Find File button to search for the file you want.
4. To display the object in the Word document as an icon instead of as the object itself, select the Display As Icon check box.
5. To return to Word, do one of the following:
   - If the object was created in another application that is in a separate window, choose Exit from the File menu in that application. If a message appears asking if you want to update the document, choose the Yes button.
   - If the application temporarily replaces some of the Word menus and toolbars, click anywhere outside the embedded object.

Note An embedded object increases the file size of a Word document because the object is stored in the Word document. If you want to reduce the file size, see "Converting an Embedded Object to a Graphic," later in this chapter.

▶ To embed a selection from an existing file

1. In Word, position the insertion point where you want to embed the selection.
2. Switch to the source application, and then open the file from which you want to select information to embed in the Word document.
3. Select the information you want to embed in the Word document.
4. From the Edit menu in the application the selection was created in, choose Copy.

5. Switch to Word, and then choose Paste Special from the Edit menu.
6. Select the Paste option button.
7. In the As box, select the first item with the word "Object" in its name.

   To display the embedded information as an icon in the Word document, select the Display As Icon check box.
8. Choose the OK button.

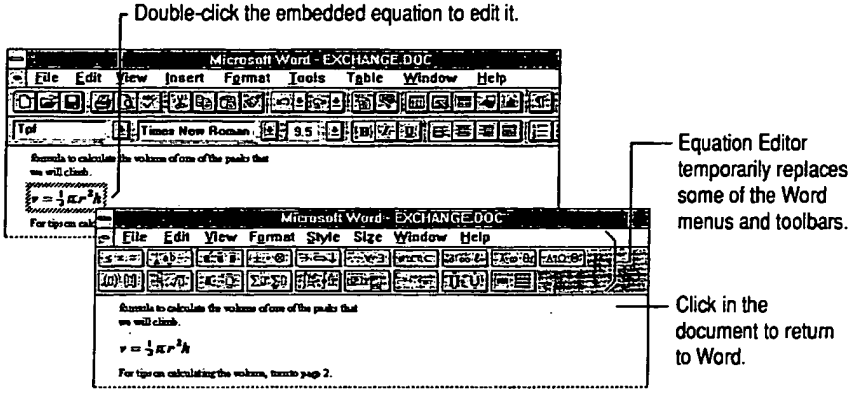
When you embed an object, Word adds an {EMBED...} field to the document. If you see this field instead of the object you embedded, select the field and press SHIFT+F9, or choose Options from the Tools menu, select the View tab, and then clear the Field Codes check box.

For information about field codes, see Chapter 32, "Inserting Information with Fields."

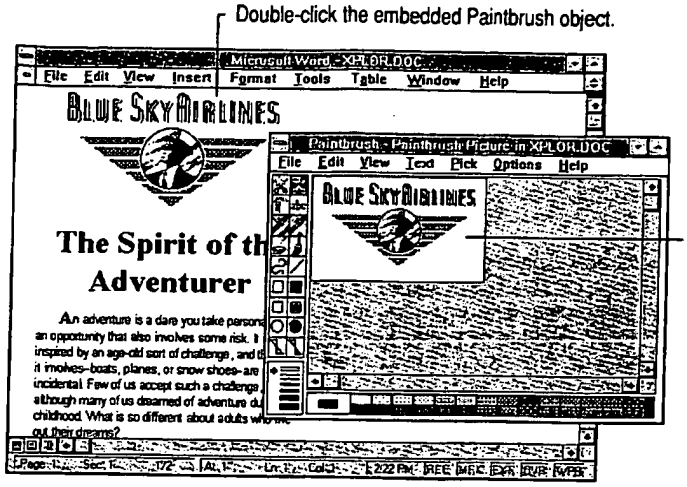
Editing an Embedded Object

In most cases, double-clicking an embedded object opens the application in which it was created. Note, however, that double-clicking some graphics brings up the Drawing toolbar rather than the original application.

For certain applications, some of the Word menus and toolbars are temporarily replaced by those of the original application. For other applications, a separate window opens, which contains the object in its original application.



Some applications, such as Equation Editor, temporarily replace the Word menus and toolbars for editing.



Some applications open a separate window for editing.

▶ **To edit an embedded object directly**

1. Double-click the embedded object.
2. Edit the object.
3. Do one of the following:
   - If you are editing the object in a separate application window, choose either Exit or Quit from the File menu to return to Word.
   - If you are editing the object in an application that temporarily replaces the Word menus and toolbars, click anywhere outside the embedded object to return to Word.

Tip You can use a shortcut menu to carry out common commands for editing an embedded object. Click the embedded object using the right mouse button (Windows) or hold down CTRL and click the embedded object (Macintosh), and then choose a command from the shortcut menu.

▶ **To edit an embedded object by using the Object command**

1. Select the object you want to edit.
2. From the bottom of the Edit menu, choose the name of the object you want to edit, and then choose Edit.

   If you see both an Edit command and an Open command, choose Open to edit the object in its own application window, or choose Edit to edit the object in the Word window (the Word toolbars and menus may change).

Note Some embedded objects, such as video and sound clips, play when you double-click them, instead of opening an application for editing. To edit one of these objects, select it, choose the name of the object from the Edit menu, and then choose Edit.

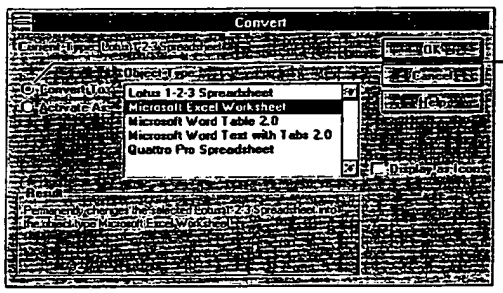
Converting an Embedded Object to a Different Format

Suppose you receive a Word document that has a Microsoft Excel worksheet embedded in it. Microsoft Excel is not installed on your computer, but you do have another spreadsheet application. When you double-click the Microsoft Excel worksheet to edit it, Word displays an error message saying that the Server could not be found. Select the worksheet, choose the Object command from the Edit menu, and then choose Convert. In the Convert dialog box, you can specify which application you'll use in the future to edit the object. The applications displayed in the list are the applications installed on your computer that are capable of converting the object to their respective file formats.

If you select the Convert To option, Word converts the embedded object to the format you choose in the Object Type box. The object remains in this format unless you specifically convert it to another format.

Now suppose you have to return the Word document to the person who created it. You know that this person uses Microsoft Excel; therefore, you don't want to permanently convert the embedded object to a different file format. However, you do want to be able to edit the embedded object on your computer. By selecting the Activate As option in the Convert dialog box, you can specify that all embedded Microsoft Excel objects are to be temporarily converted to the file format you specify. When you edit the object in question, you use the new application, but changes are saved in Microsoft Excel format. When you double-click the object, it is temporarily converted to the new format.

If you select the Activate As option, Word converts all embedded objects of the selected type to the format you specify in the Object Type box. You can edit the objects in the application you specify, but Word saves changes to the objects in their original file format.



Select this option button to convert an embedded object to a different file format.

▶ **To convert an embedded object to a different file format**

1. Select the embedded object whose source application you want to change.
2. From the Edit menu, choose the name of the object you want to convert, and then choose Convert.
3. Do one of the following:
   - To permanently convert the embedded object to the file format you specify in the Object Type box, select the Convert To option button.
   - To activate all embedded objects of the selected type in the file format you specify in the Object Type box, select the Activate As option button.
4. In the Object Type box, select the application whose file format you want to convert the embedded object to, and then choose the OK button.

Note If you upgrade to a more recent version of an application after creating an embedded object with that application and then want to edit the object, you must first convert the object to the current version of the application.

Some applications can convert files created in older versions of the application to the newer version when you open the file. When you run the application's setup program, you may be able to specify whether files are converted when you open them.

Converting an Embedded Object to a Graphic

All embedded objects are represented in Word documents as graphics. You can think of one of these graphics as a facade with information behind it. For example, a Microsoft Excel worksheet embedded in a Word document looks like an ordinary worksheet, but it is actually a graphic, or picture, of the original worksheet. Behind the graphic is all of the information necessary to open and edit the file in Microsoft Excel. When you convert the Microsoft Excel object to a graphic, you remove the information behind the facade—that is, the information that enables Word to open the object in Microsoft Excel for editing.

Converting an embedded object to a graphic reduces the file size of the Word document. If you double-click the object after converting it to a graphic, Word opens a separate window and displays the Drawing toolbar so that you can edit the graphic just as you would edit any other graphic.

▶ **To convert an embedded object to a graphic**

1. Select the object you want to convert.
2. From the Edit menu, choose the name of the object you want to convert, and then choose Convert.

3. In the Object Type box, select Picture, and then choose the OK button.

Using the keyboard You can convert objects to graphics quickly by using shortcut keys. Select the embedded object, and then press CTRL+SHIFT+F9 (Windows) or COMMAND+SHIFT+F9 (Macintosh). This method does not create an {Embed} field.

Linking to Another File

Let's say you are preparing a monthly report for the accounting department that must include up-to-date sales data. The sales department maintains this constantly changing data in a Microsoft Excel worksheet. You can link your Word document to the sales worksheet (or a portion of it) and specify that your document be automatically updated if the worksheet changes. Each time the sales department changes the worksheet, the changes are reflected in your Word document.

When you link to another file, Word stores the link in the form of a field code that indicates the source of the object. In addition, Word usually stores a visual representation of the linked information.

You can create links between two Word documents or between a Word document and a file created in another application. Once you have established a link, you can update it with a single keystroke, or you can specify that the data be updated in your Word document as soon as it changes in the source file.

For information about fields, see Chapter 32, "Inserting Information with Fields."

Embedded Objects and Links are Represented by Fields

Linked and embedded objects in Word are represented by fields. If you are working with field codes displayed, you don't see the linked or embedded object itself; you see the code that Word uses to designate the object. For example, the code for a link to a Microsoft Excel worksheet might look like the following in a Word for Windows document:

```
{LINK ExcelWorksheet "C:\\EXCEL\\SALES.XLS" "R1C1:R9C5" \a \p}
```

or like the following in a Word for the Macintosh document:

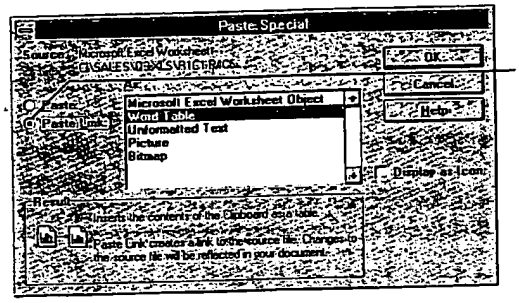
```
{LINK ExcelWorksheet "Hard Drive:Excel:Sales" "R1C1:R9C5"}
```

To specify whether the object itself or the field code is displayed, choose Options from the Tools menu, select the View tab, and then select or clear the Field Codes check box.

Creating a Link

Creating a link is as easy as copying and pasting. You copy a selection from a file (the *source*) and paste it into your Word document (the *destination*) by using the Paste Special command on the Edit menu. Before you can establish a link, you must save the source file to disk.

Important To create a link between a Word document and another application, you must be running both applications, and the other application must support dynamic data exchange (DDE) or object linking and embedding (OLE). On the Macintosh, you must be running both applications with System 7 or later, and your computer must have enough memory to run both applications at the same time.



Select Paste Link to create a link.

The Paste Special dialog box

▶ **To create a link to another file or Word document**

1. Make sure that you save the source file before you link the information.
2. In the application in which the information you want to link was created, open the source file and then select the information you want to link.
3. From the Edit menu, choose Copy.
4. Switch to the Word document, and then position the insertion point where you want to insert the linked information.
5. From the Edit menu, choose Paste Special.
6. Select the Paste Link option button.
7. Under As, select the format you want, and then choose the OK button.

▶ **To create a link to another file or Word document without leaving Word**

1. From the Insert menu, choose Object.
2. Select the Create From File tab.
3. In the File Name box, type or select the name of the file to which you want to link.
4. Select the Link To File check box, and then choose the OK button.

Using this method, you can create a link only to an entire file; you cannot link to a selection in a file.

Note Word creates automatic links by default. Word updates automatic links each time you open the Word document, whereas it updates manual links only when you specify. For information, see "Updating a Link," later in this chapter.

Reducing File Size of Documents Containing Linked Graphics

When a Word document contains links to graphics files, you can reduce the file size of the Word document by storing only the links. By default, Word stores in the Word document a "picture" of the linked graphic, which increases the file size of the Word document by the number of bytes taken up by the picture. To reduce the size of the Word file, you can specify that Word store only the link itself and not the picture.

If you store only the link, the file size of the Word document will not increase appreciably. However, if the source file is not available, you will see only a rectangular placeholder in your document and the linked data will not be printed. If the source file is available, Word displays a picture of the object based on data in the source file, but it does not store the picture in the Word document. Because the picture is created from the source file itself, you may notice that it takes longer to display the picture than if it were stored in the Word document.

If the picture is stored in the Word document, Word displays a picture of the linked graphic regardless of whether or not the source file is available.

▶ **To reduce the file size of a document containing linked graphics**

1. From the Edit menu, choose Links.
2. In the Links dialog box, select the link or links to the graphics files.
3. Clear the Save Picture In Document check box, and then choose the OK button.

Reconnecting or Changing a Link

You may lose a link if you rename or move the source file. If this happens, you must reconnect the link to the original source file or redirect the link to a different file.

▶ **To reconnect or change a link**

1. From the Edit menu, choose Links.
2. From the list, select the link you want to reconnect or change. To select multiple links, hold down CTRL (Windows) or SHIFT (Macintosh) while you click each link.
3. Choose the Change Source button.
4. In the File Name box, type or select the name of the file to which you want to reconnect or change the link, and then choose the OK button.

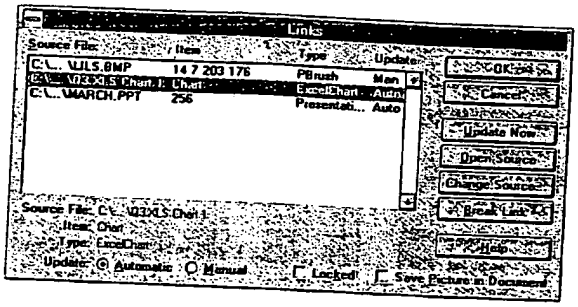
   If you do not see the file you want to open, select a different drive, directory, or folder, or choose the Find File button to search for the file you want.

If you have other links to the same source file, Word asks you to confirm that you want to change all links from the previous source file to the new source file.

Updating a Link

When information in the source document changes, Word can update the information in the Word document. You can specify either manual or automatic updating for each link. By default, newly created links are set to automatic updating, but you can easily change a link to manual updating.

- Word updates automatic links when you open the Word document and any time the source document is changed while the Word document is open.
- Word updates manual links only when you choose the Update Now button in the Links dialog box (Edit menu) or when you position the insertion point in the linked object and press F9.



The Links dialog box

If you change any editable text or numbers in the linked object while working in the Word document, the changes will be overwritten when Word updates the linked material. You can, however, apply formatting—such as bold or italic format or centered paragraph alignment—in the linked object. Word retains such formatting and reapplies it to the text or numbers when they are updated.

▶ To control how links are updated

1. From the Edit menu, choose Links.
1. From the list, select the link to the information you want to update. To select multiple links, hold down CTRL (Windows) or SHIFT (Macintosh) while you click each link.
3. Do one of the following:
   - To update linked information every time there's a change in the source file, select the Automatic option button next to Update.
   - To update linked information only when you choose, select the Manual option button next to Update.
4. Choose the OK button.

▶ To update a link manually

1. From the Edit menu, choose Links.
2. From the list, select the link to the information you want to update. To select multiple links, hold down CTRL (Windows) or SHIFT (Macintosh) while you click each link.
3. Choose the Update Now button.

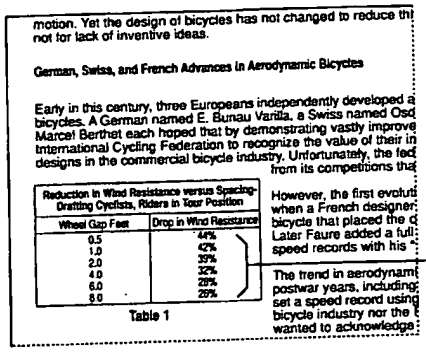
   For each selected link, the destination document reflects any changes made in the source file since the last update.

Examples of Embedding and Linking

The following two scenarios illustrate common uses of embedding and linking.

Embedding a Microsoft Excel Worksheet in Word

Embedding is a way to get fast access to the features of another application. In the following example, a table illustrates the relationship between various bicycle designs and the drop in wind resistance. You can create a Word table to present the data. However, if you embed a Microsoft Excel worksheet instead, you can use the more powerful Microsoft Excel formulas to calculate the results you want to present.


motion. Yet the design of bicycles has not changed to reduce th
not for lack of inventive ideas.

German, Swiss, and French Advances in Aerodynamic Bicycles

Early in this century, three Europeans independently developed a
bicycles. A German named E. Bunau Varilla, a Swiss named Osc
Marcel Berthet each hoped that by demonstrating vastly improve
International Cycling Federation to recognize the value of their in
designs in the commercial bicycle industry. Unfortunately, the fed
from its competitions tha

Reduction in Wind Resistance versus Spacing-Drafting Cyclists, Riders in Tour Position

Wheel Gap Feet	Drop in Wind Resistance
0.5	44%
1.0	42%
2.0	35%
4.0	32%
6.0	28%
8.0	26%

Table 1

However, the first evoluti
when a French designer
bicycle that placed the c
Later Faure added a full
speed records with his

The trend in aerodynam
postwar years, including
set a speed record using
bicycle industry nor the
wanted to acknowledge

Use a formula in Microsoft Excel to calculate the data.

You can also click the Insert Microsoft Excel Worksheet button on the Standard toolbar to embed a worksheet.

To create and embed the worksheet, position the insertion point where you want to embed the worksheet, and then choose Object from the Insert menu. Select the Create New tab, select Microsoft Excel Worksheet in the Object Type box, and then choose the OK button. Microsoft Excel opens, and you can now create the worksheet. When you have finished, choose Exit from the File menu in Microsoft Excel. A message appears, asking if you want to update the worksheet in your document; if you choose the Yes button, the worksheet is inserted into the Word document. You can double-click the worksheet object in the Word document at any time to open Microsoft Excel and make changes.

Linking a Microsoft Excel Worksheet to a Word Document

Linking is a good way to make use of information that's stored and updated in other files. The following example shows a monthly sales report that contains data for a three-month period. The data is maintained by the sales department in a large Microsoft Excel worksheet that also contains a lot of other data. By creating a link to a specific section of the worksheet, you give yourself immediate access to the most recent data needed for the monthly report. If the sales department changes the worksheet, the Word document can be automatically updated.

and linking.

)lication. In the
)us bicycle
·le to present
:ad, you can
sults you want

cel

ere you want
:nu. Select the
/pe box, and
w create the
ıu in Microsoft
ıeet in your
nto the Word
document at

)ocument

. updated in
at contains data
tment in a
ta. By creating
ediate access
epartment
pdated.

The sales data can be automatically updated when the original Microsoft Excel worksheet is changed.

March Sales Report

Hightlights and Major Achievements

- Made 144% of forecast for March, a new monthly record! We attribute the outstanding sales this month to our in-store promotions and to the hard work of our sales force.
- Presented strategy and new product plans to Marketing VP on March 5. See Paul Brach for a summary of comments that came out of that meeting.
- Increased Region 4 sales 50% by distributing a special edition of the spring catalog.

Business Summary

Our March sales continued this quarter's trend of rising revenues. For the first time this year, we exceeded the cumulative year-to-date sales forecast:

	January	*February*	*March*
Month's Sales	40,982	65,832	65,929
Sales Forecast	45,200	78,300	45,900
Cumulative Sales	40,982	106,814	172,743
Cumulative Forecast	45,200	123,500	169,400

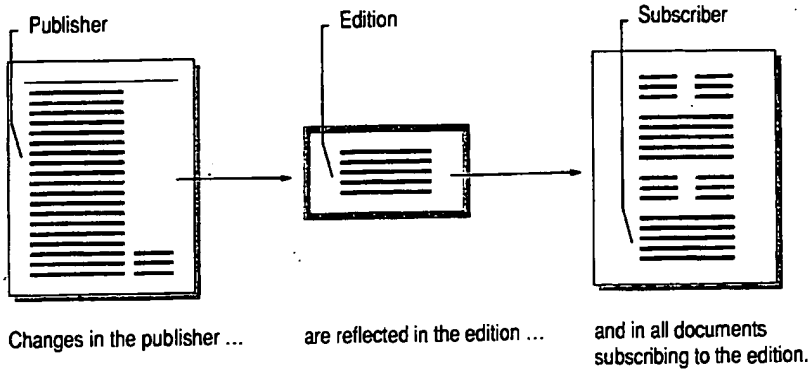
To create the link, open the sales department's Microsoft Excel worksheet, select the data you want to include in the report, and then choose Copy from the Edit menu. Open the monthly report in Word, position the insertion point where you want to insert the data, and then choose Paste Special from the Edit menu. In the Paste Special dialog box, select the Paste Link option button, select Microsoft Excel Worksheet Object in the As box, and then choose the OK button. The worksheet data appears in the Word document. By default, any changes made to the worksheet in Microsoft Excel will automatically appear in the Word document. However, you can prevent automatic updates by choosing Links from the Edit menu, selecting the link, selecting the Manual option button at the bottom of the dialog box, and then choosing the OK button.

You use this same process to link other items—such as graphics or Microsoft Excel charts—to the Word document

Publishing and Subscribing in Word for the Macintosh

With Word for the Macintosh running under System 7 or a later version, you can exchange information between documents in different applications or on different computers connected by a network. To make part of a document available for use with other applications or for other users on a network, you can create a *publisher* for that portion of the document. A publisher contains the part of the document you want to share—text, graphics, spreadsheet data, and so forth.

When you create a publisher, Word automatically creates an intermediate file called an *edition*, which contains a copy of the information that's in the publisher. An edition is a separate file that can be saved on a hard disk or a network server. When you change information in the publisher, these changes are reflected in the edition and in all documents *subscribing* (similar to linking) to the edition. You can specify how often Word sends updated information from the publisher to the edition. You can also specify how often subscribers receive updated information from the edition.



You can name and move an edition just as you would any other file. The connection is maintained even if you change the name of the edition. However, if you want Word to maintain the connection to all publishers and subscribers, do not move the edition off the server volume or hard disk.

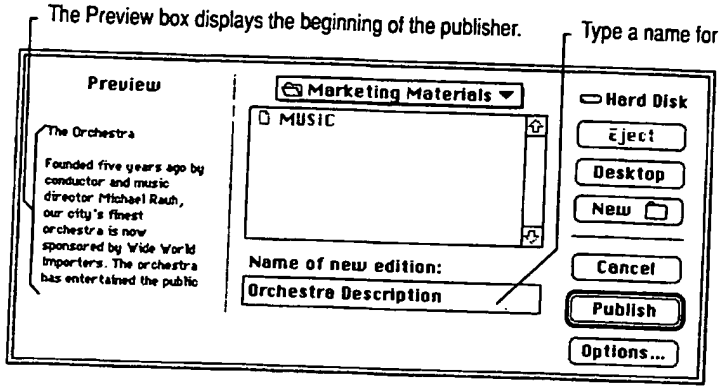
You can publish and subscribe from any application that runs under System 7 or later and supports publishing and subscribing.

Creating a Publisher and an Edition

Use the Create Publisher command to publish information from a Word document. You can edit a publisher the same way you edit other parts of a document. When you create a publisher, Word puts gray brackets around the part of the document that constitutes the publisher. You can see the brackets when you click the Show/Hide ¶ button on the Standard toolbar, but the brackets are not printed.

For information about bookmarks, see Chapter 19, "Cross-references, Captions, and Bookmarks."

Word uses a bookmark to mark a publisher. The name you give the edition when you create a publisher is the name Word uses for the bookmark. If there is already a bookmark by that name, Word adds a number to the end of the bookmark name to make it unique. If you delete the bookmark, the publisher is deleted as well.



The Create Publisher dialog box

▶ To create a publisher

1. Select the information you want to publish.

   If you publish an entire document, including the final paragraph mark, Word includes any material that's added to the document later as part of the publisher. To add information you do not want to publish to the end of the document, exclude the final paragraph mark from the selected information.
2. From the Edit menu, choose Publishing, and then choose Create Publisher.
3. Switch to the disk or open the folder in which you want to store the edition.

   If you want to store the edition on another Macintosh, choose the Desktop button. Word displays all of the available hard disks and servers. Select the computer you want, and then select the folder you want from the list.
4. In the Name Of New Edition box, type a name for the edition.
5. Choose the Publish button.

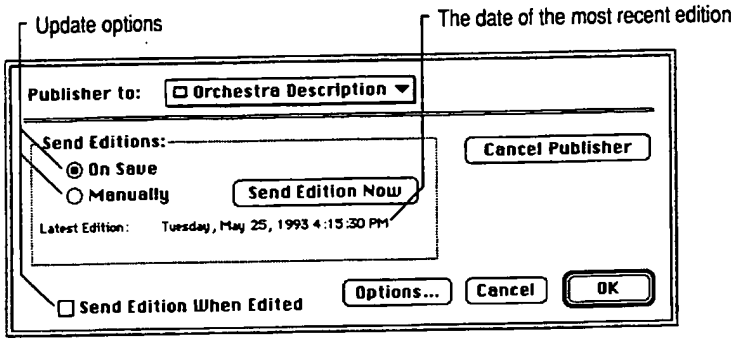
   Word encloses the published information in brackets.

   Each time you create a new publisher, Word creates a new edition. If you publish several parts of the same document, each publisher has its own separate edition.

Note If you want to make editions available to other computers on a network, you must store the editions on your hard disk and use the file sharing option; make sure to share the folder that contains the editions. To share files on the Macintosh under System 7, choose Control Panels from the Apple menu, double-click the Sharing Setup icon, and then choose the Start button under File Sharing. For more information, see your Macintosh system documentation.

Updating an Edition

Once you have created a publisher and an edition, you can specify how frequently you want to update the edition with changes you make to the publisher. Unless you specify otherwise, Word updates the edition as soon as you save changes to the publisher. There are three update options in the Publisher Options dialog box: On Save, Manually, and Send Edition When Edited. These options are described in the procedure following this illustration.



The Publisher Options dialog box

▶ **To control how an edition is updated**

1. Select the publisher whose update frequency you want to change.
2. From the Edit menu, choose Publishing, and then choose Publisher Options.
3. Under Send Editions, do one of the following:

To update the edition	Do this
Whenever you save the publisher	Select the On Save option button.
Only when you choose the Send Edition Now button	Clear the Send Edition When Edited check box, and then select the Manually option button.
Whenever you make changes to the publisher	Select the On Save option button, and then select the Send Edition When Edited check box.

4. Choose the OK button.

▶ **To send an edition manually**

1. Select the publisher you want to update.
2. From the Edit menu, choose Publishing, and then choose Publisher Options.
3. Choose the Send Edition Now button.

Canceling a Publisher

If you decide that you no longer want to publish information in your document, you can cancel the publisher. The contents of the publisher remain in your document. Other users can still subscribe to the edition, but it is no longer updated. To delete the edition, delete it in the Finder just as you would delete any other file.

▶ **To cancel a publisher**

1. Position the insertion point in the publisher you want to cancel.
2. From the Edit menu, choose Publishing, and then choose Publisher Options.
3. Choose the Cancel Publisher button.

   Word asks you to confirm that you want to cancel the publisher. When you cancel a publisher, Word deletes the brackets surrounding the publisher.

Subscribing to an Edition

When you subscribe to an edition, you insert a copy of the edition into your Word document. This copy is called the subscriber. Once you have inserted a subscriber into the Word document, updates received by the edition are automatically sent to the subscriber. As long as the edition remains on the same server volume or hard disk, Word maintains the connection between the edition and the subscriber, even if you change the name of either the edition or the subscriber.

▶ **To subscribe to an edition**

1. Position the insertion point in the document where you want to insert a copy of the edition.

   Make sure that the insertion point is not positioned in another subscriber in the document.

2. From the Edit menu, choose Publishing, and then choose Subscribe To.

3. From the list of files, select the edition you want to subscribe to.

   If the edition is located on another hard disk or another computer, choose the Desktop button. Word displays all available hard disks and computers. Select the hard disk or machine you want, and then select the edition from the list.

4. In the Subscribe With box, select the format you want to use for the subscriber data.
5. Choose the Subscribe button.

Updating a Subscriber

When you've subscribed to an edition, you can specify how frequently you want to receive updated information from the edition. Unless you select another option, Word updates the subscriber automatically as soon as a new edition is available—that is, any time the publisher sends changed information to the edition.

▶ **To control how a subscriber is updated**

1. Select the subscriber whose update frequency you want to change.
2. From the Edit menu, choose Publishing, and then choose Subscriber Options.
3. Under Get Editions, do one of the following:

To update the subscriber	Choose
Whenever a change is made in the edition	The Automatically button
Only when you specify	The Manually button

Note Changing the update frequency of the subscriber affects only how often the subscriber receives updates from the edition, not how frequently the publisher sends updates to the edition. You set the update frequency of the publisher independently. For information, see "Updating an Edition," earlier in this chapter.

▶ **To update a subscriber manually**

1. Select the subscriber you want to update.
2. From the Edit menu, choose Publishing, and then choose Subscriber Options.
3. Choose the Get Edition Now button.

Note You can format a subscriber, but Word overwrites the subscriber each time it receives an updated edition as long as the * MERGEFORMAT switch remains in the field. For more information, double-click the Help button on the Standard toolbar and type **format (*) switch**

Switching from a Subscriber to Its Publisher

If you need to change the contents or formatting of a subscriber, it's best to make the changes in the publisher itself. This way, the changes will be reflected in the subscriber. If you are connected to a network, you must have access to the publisher to perform this procedure.

▶ **To switch from a subscriber to its publisher**

1. Select the subscriber you want to edit.
2. From the Edit menu, choose Publishing, and then choose Subscriber Options.
3. Choose the Open Publisher button.

   Word opens the document that contains the publisher you want.
4. Make the changes in the publisher.

   When you finish making changes in the publisher, save and close the document. Each subscriber reflects the changes according to the update options you've selected in the publisher and subscriber.

Canceling a Subscriber

If you do not have access to the publisher to make changes and you do not need to receive any more updates from the edition, you can cancel the subscriber. You can then edit the information as you would edit any other text, without losing any changes when updates are sent. The contents of the subscriber remain in the document.

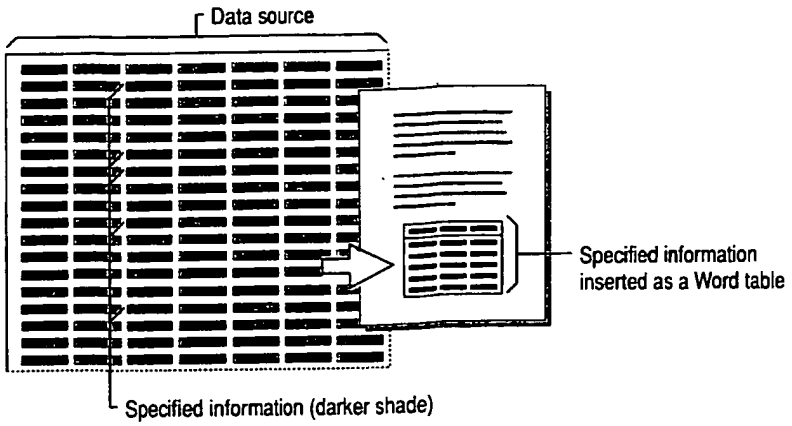
▶ **To cancel a subscriber**

1. Select the subscriber you want to cancel.
2. From the Edit menu, choose Publishing, and then choose Subscriber Options.
3. Choose the Cancel Subscriber button.

   Word asks you to confirm that you want to cancel the subscriber. When you cancel a subscriber, Word deletes the brackets surrounding the subscriber.

Inserting Tables of Information from a Database

Sometimes you may want to include in a Word document information from an existing database, a Microsoft Excel worksheet, or another source of data. By using the Database command on the Insert menu, you can specify the information you want and automatically insert it as a table in a Word document. You can screen, or "filter," the information according to criteria you select. You can also instruct Word to update the information in the Word document if the source file has changed.



Word can retrieve information from the following types of files:

- Files from the following applications that are installed on your system:

  Microsoft Access®
  Microsoft Excel

- Files from single-tier, file-based database applications for which you have an open database connectivity (ODBC) driver installed in the System subdirectory of your Windows directory. ODBC drivers for the following applications are supplied with Word:

  Microsoft Access
  Microsoft FoxPro® (or other Xbase database application such as dBASE®)
  Paradox®

For a list of file converters provided with Word, see Chapter 26, "Converting File Formats."

- Files for which you have a file converter installed. In addition to converters for ASCII text files, Word provides file converters for many applications, including:

Microsoft Word for Windows	WordPerfect 5.*x* for MS-DOS and Windows
Microsoft Word for the Macintosh versions 3.*x*,[1] 4.*x*, and 5.*x*	Microsoft Excel 2.x,[2] 3.0, 4.0,[1] and 5.0[3]
Microsoft Word for MS-DOS 3.0–6.0	Lotus 1-2-3 2.*x*[2] and 3.*x*[1]

[1] Converts only from this format.

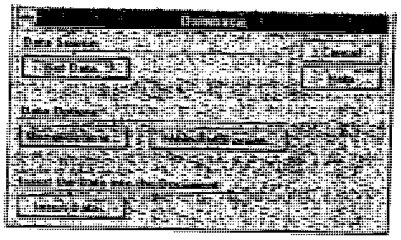
[2] Converter works only with Windows version.

[3] Converter works only with Macintosh version.

You can also insert information from another Word document. For example, you might have set up a membership directory for use as a mail merge data source. Instead of copying and pasting information from various data records, use the Database command to insert just the information you request.

Inserting the Data

When you choose the Database command from the Insert menu, Word displays the Database dialog box. Now you can locate the data source, select the information you want, and format the table in which the information is displayed.



Once you select the data source, the other buttons in the dialog box become available.

By default, Word inserts all of the information from the selected data source. In most cases, however, you'll want to use only some of the available information. For example, from a large personnel file, you might want to list only the names, departments, and hiring dates of all employees who have worked for your company 10 years or longer.

If information in the data source changes frequently and you want to keep your document up to date, you can insert the information as a Word *field*. The field is simply a "placeholder" that represents the table in your document. For more information, see "Keeping the Table Information Up to Date," later in this chapter.

▶ **To insert information from a data source as a table**

1. Position the insertion point where you want the new table of information to be included.
2. From the Insert menu, choose Database.
3. Choose the Get Data button.
4. In the Open Data Source dialog box, type or select the filename of the data source you want to open, and then choose the OK button.

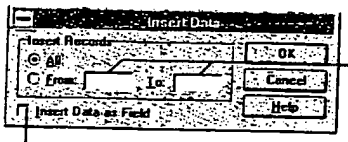
   If the data source is not listed, select the appropriate drive and directory or folder. Then select the appropriate option in the List Files Of Type box.

   If you open a Microsoft Excel worksheet, you can insert the entire worksheet or a range of cells. If you open a Microsoft Access database, you can insert records from a table or a selection of records defined by a query. For more information, see the documentation for the application you are using.
5. To insert specific information from the data source or list the information in a particular order, choose the Query Options button. Do one or more of the following, and then choose the OK button.
   - On the Filter Records tab, specify criteria to select the data records to insert.
   - On the Sort Records tab, select the data fields by which you want to sort the information.
   - On the Select Fields tab, remove any fields you don't want from the Selected Fields list. The order of the fields in the list determines the order in which the fields are inserted left to right.

     If you don't want to insert the field names from the header row with the data records, clear the Include Field Names check box.
6. To format the table, choose the Table AutoFormat button.
7. Choose the Insert Data button.

   In the Insert Data dialog box, you can specify the range of records you want to insert. The range refers only to the records that were selected by the query. If you want to be able to update the information in the table automatically, select the Insert Data As Field check box. Then choose the OK button.

Note If you insert more than 31 data fields, Word inserts tab characters to separate the columns of information.



To specify which of the selected records are inserted, type starting and ending record numbers in the From and To boxes.

Select this check box if you want to keep the table information up to date.

Modifying the table format If you don't select the Insert Data As Field check box, Word inserts the information as an ordinary table. You can resize the table columns and otherwise modify the table by using the commands on the Table menu. If you insert the information as a field, however, you must choose the Database command again to reinsert the table and update the table format by choosing the Table AutoFormat button. Otherwise, the table formatting you've applied is removed the next time you update the DATABASE field. Formatting you've applied to text in the table is also removed. For more information, see "Keeping the Table Information Up to Date," later in this chapter.

Modifying the information in the table You may want to modify the information in the table later. For example, you might want to include another column of information or select a different set of records from the data source. To do this, click in the table and then choose the Database command again to select the information you want in the table. If you insert the information as a field and then edit or format the text in the displayed table, your changes will be deleted the next time you update the DATABASE field. For more information, see "Keeping the Table Information Up to Date," later in this chapter.

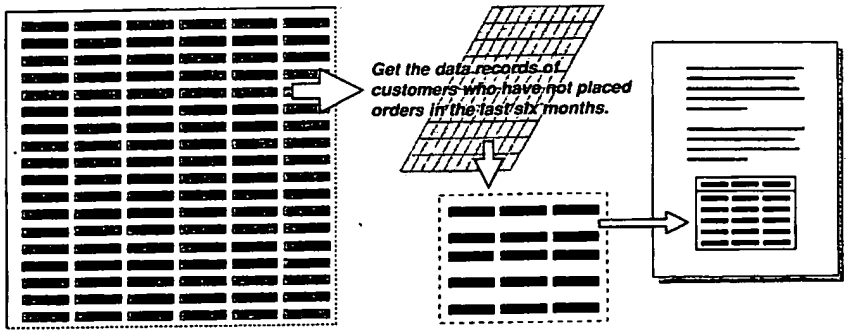
If Word can't recognize the field and record delimiters If Word can't recognize the characters used to separate data fields and data records in text-delimited files (files in which data is separated by commas, tab characters, or other characters), Word asks you to select the separating characters (delimiters). Word recognizes one data field delimiter and one data record delimiter. If a combination of two or more characters is used as a delimiter, then the remaining characters are treated as text in the data fields.

Selecting the Data

To get only the information you want from a data source, you create a *query*. A query is simply a set of instructions, or rules, that describes the information you want from the data source. You can think of the following statement as a query:

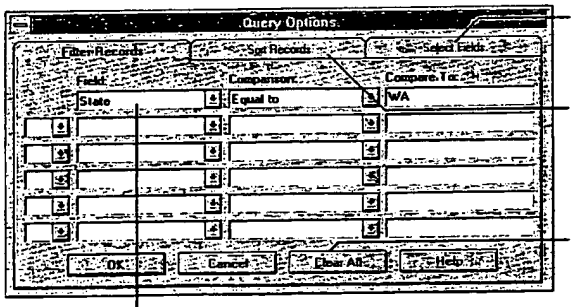
"Give me the names, addresses, and account numbers of all customers who have not placed orders in the last six months."

The first part of the statement identifies the categories of information you want—names, addresses, and account numbers. The second part of the statement indicates that you want information only for certain customers—those who have not ordered anything in the last six months.



A query tells Word which information to select from a data source.

You create queries by selecting options in the Query Options dialog box. You select data fields to specify the categories of information you want. The order in which you select the data fields determines the order of the columns of information in the table, from left to right. To get the information only from certain data records, you specify one or more rules for selecting the records. To list the rows of information in a particular order, you can sort the data records.



Select this tab to specify the categories of information in the table.

Select this tab to specify the order information is listed in the table.

Choose this button to delete the current rules.

Word selects all data records with "WA" in the data field "State."

Specifying the Record-Selection Rules

On the Filter Records tab, you specify the rules that Word uses to retrieve the information you want, based on the contents of selected data fields. When specifying a rule, you can select any data field in the data source—even a data field you don't want to include in the table.

A record-selection rule is made up of three parts:

- A field name corresponding to a data field in the selected data source
- A comparison phrase, such as "Equal To" or "Is Not Blank"
- Text or numbers you want the contents of the data field to be compared with

If you compare text When comparing a data field that contains text, Word compares the sequence of characters based on the ANSI sorting order. The text "apple" is considered "less than" the word "berry" because, alphabetically, "apple" precedes "berry." Whether the text is uppercase or lowercase isn't significant.

For example, to retrieve data records for members of your organization whose last names begin with "A" through "L," you specify the following rule:

LastName Is Less Than M

Any name beginning with "A" through "L" is considered less than "M," so only the data records that contain those names are selected. (The last name must be contained in a separate data field, or else it must precede the first name in the field—for example, "Bendal, Maria".)

If you compare numbers mixed with text If numbers are mixed with letters, hyphens, plus or minus signs, or other nonnumeric characters, Word compares the numbers as though they were a sequence of text characters. For example, a five-digit U.S. ZIP Code is compared as a number, whereas a nine-digit "ZIP+4" code such as 99999-9099 is compared as text, as are non-U.S. postal codes that contain letters.

Comparing sequences of mixed numbers and nonnumeric characters—code numbers, for example—can have different results if some items contain more sequential numerals than others. For example, the following items are sorted in this order:

0002xy, 002, 011y, 1, 1x, 1yz, 22x, 2x

The following items, however, are sorted in this order:

0001, 0001x, 0001yz, 0002, 0002x, 0002xy, 0011y, 0022x

Specifying Multiple Rules

You can specify as many as six selection rules. Using multiple rules allows you to narrow the range of data records that are selected. When you select multiple rules, you must specify AND or OR to connect each additional rule to the preceding rule, as in the following examples.

Example 1		Example 2	
	State (Is) Equal To Oregon		State (Is) Equal To Oregon
AND	City (Is) Equal To Portland	OR	State (Is) Equal To California

The rules connected by AND select only data records that contain both "Oregon" in the State field and "Portland" in the City field. The rules connected by OR select all data records that contain either "Oregon" or "California" in the State field—a potentially larger number of records. The key difference between AND and OR is as follows:

- When you use AND to connect rules, Word selects only those records that satisfy both (or all) rules. Each rule connected by AND *eliminates* more of the records in the data source.
- When you use OR to connect rules, Word selects any record that satisfies at least one of the connected rules. Each rule connected by OR *selects more* of the records in the data source.

AND has precedence You can use AND and OR separately or in combination. In sets of rules that contain both AND and OR, rules connected by AND have precedence over rules connected by OR. This means that the set of rules connected by AND is used to select records before the set of rules connected by OR. How you connect the rules—by using AND or by using OR—affects which data records are selected.

Suppose you want to select data records of all clients who live in either Portland or Salem, Oregon. In the Query Options dialog box, you would specify the following rules to determine the contents of the data fields "City" and "State":

```
      State (Is) Equal To Oregon
AND   City (Is) Equal To Portland
OR    State (Is) Equal To Oregon
AND   City (Is) Equal To Salem
```

Using the first set of rules connected by AND, Word compares the data records to identify the clients who live in Portland, Oregon. Next, Word compares the data records with the next set of rules connected by AND. Word then selects only data records of clients in Oregon who live in either Portland or Salem.

Notice that the following set of rules does *not* produce the same result:

State (Is) Equal To Oregon
AND City (Is) Equal To Portland
OR City (Is) Equal To Salem

Because AND takes precedence, the first set of rules connected by AND selects records of clients who live in Portland, Oregon. However, the rule connected by OR also selects records for clients in any city named Salem—including Salem, Massachusetts, for instance

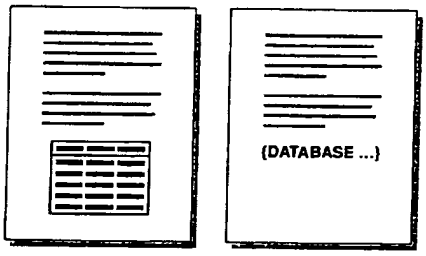
Comparing a range of values You can also use AND to compare a selected field with a range of values rather than a single value. For example, given the following rules, Word selects all data records that have a value of 98001 through 98500 in the PostalCode field.

PostalCode (Is) Greater Than Or Equal (To) 98001
AND PostalCode (Is) Less Than Or Equal (To) 98500

Keeping the Table Information Up to Date

If you select the Insert Data As Field check box when you insert the information, Word does not insert an actual table; instead, it inserts a DATABASE field to represent the table.

With field codes hidden, the information is displayed as a table. With field codes displayed, the information is displayed as a DATABASE field. The field contains all information needed to locate and open the selected data source, carry out the query, and insert the information in your document.



Document with field codes hidden ...

... and with field codes displayed.

To display or hide the field codes, press ALT+F9 (Windows) or OPTION+F9 (Macintosh).

Bruce Schneier

Crypto Bibliography

Citations by First Author - B

A. Back, U. Möller, and A. Stiglic, Traffic Analysis Attacks and Trade-Offs in Anonymity Providing Systems, Proceedings of the 4th Information Hiding Workshop (IHW2001), Springer-Verlag, LNCS v. 2137, pp. 243-254. [.pdf]

S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, A Message Authentication Code based on Latin Squares, Australian Conference on Information Security and Privacy (ACISP '97), Springer-Verlag, LNCS 1270, pp. 194-203, 1997. [.ps.Z]

S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, On Password-Based Authenticated Key Exchange using Collisionful Hash Functions. In Australian Conference on Information Security and Privacy (ACISP '96), Springer-Verlag, LNCS 1172, pp. 299-310, 1996. [.ps.Z]

S. Bakhtiari, R. Safavi-Naini, and J. Piprzyk, On Selectable Collisionful Hash Functions, Australian Conference on Information Security and Privacy (ACISP '96), Springer-Verlag, LNCS 1172, pages 287-298, 1996. [.ps.Z]

T. Baldin, G. Bleumer, and R. Kanne, CryptoManager - Eine intuitive Programmierschnittstelle für kryptographische Systeme; Sicherheitsschnittstellen - Konzepte, Anwendungen und Einsatzbeispiele, Proc. Workshop Security Application Programming Interfaces 94, Deutscher Universitäts Verlag, München 1994, 79-94. [.ps.gz]

T. Baldin and G. Bleumer, CryptoManager++ -- An object oriented software library for cryptographic mechanisms; 12th IFIP International Conference on Information Security (IFIP/Sec '96), Chapman & Hall, London 1996, 489-491. [.ps.gz]

D. Balfanz and L. Gong, Experience with Secure Multi-Processing in Java, Proceedings of the 18th IEEE International Conference on Distributed Computing Systems (ICDCS), Amsterdam, Netherlands, May 1998. [.ps.gz]

J. Bar-Ilan and D. Beaver, Non-Cryptographic Fault-Tolerant Computing in a Constant Expected Number of Rounds of Interaction (extended abstract); Proceedings of **PODC**, ACM, 1989, 201-209. [.pdf]

R. Bar-Yehuda, B. Chor, E. Kushilevitz, and A. Orlitsky, Privacy, Additional Information, and Communication, IEEE IT 39(6), 1993, pp. 1930-1943. [.ps.Z]

N. Baric and B. Pfitzmann, Collision-Free Accumulators and Fail-Stop Signature Schemes Without Trees; Eurocrypt '97, LNCS 1233, Springer-Verlag, Berlin 1997, 480-494. [.ps.gz]

E. Basturk, M. Bellare, C. S. Chow, and R. Guerin, Secure transport protocols for high-speed networks, IBM Research Report 19981, March, 1994.

O. Baudron, H. Gilbert, L. Granboulan, H. Handschuh, A. Joux, P. Nguyen, F. Noilhan, D. Pointcheval, T. Pornin, G. Poupard, J. Stern, and S. Vaudenay, Report on the AES Candidates, Proceedings of the Second AES Candidate Conference, Rome, Italy, 1999. [.pdf]

B. Baum-Waidner, B. Pfitzmann, and M. Waidner, Unconditional Byzantine Agreement with Good Majority; STACS'91, LNCS 480, Springer-Verlag, Heidelberg 1991, 285-295. [.ps.gz]

D. Bayer, S. Haber, and W. Stornetta, Improving the Efficiency and Reliability of Digital Time-Stamping, Sequences II: Methods in Communication, Security, and Computer Science, eds. R. Capocelli, A. DeSantis, and U. Vaccaro, Springer-Verlag, 1993, pp. 329-334. [.pdf]

P. Beauchemin, G. Brassard, C. Crépeau, C. Goutier, and C. Pomerance, Two observations on probabilistic primality testing; In Advances in Cryptology: Proceedings of Crypto '86, volume 263 of Lecture Notes in Computer Science, pages 443-450. Springer-Verlag, 1987. [.ps.gz]

P. Beauchemin, G. Brassard, C. Crépeau, C. Goutier, and C. Pomerance, The generation of random

numbers that are probably prime, Journal of Cryptology, 1(1):53-64, 1988. [.ps]

D. Beaver, S. Micali, and P. Rogaway, The Round Complexity of Secure Protocols (extended abstract); Proceedings of the 22nd **STOC**, ACM, 1990, 503-513. [.ps] [.ps.gz]

D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, Security with Low Communication Overhead (extended abstract), Advances in Cryptology - Crypto '90 Proceedings, Springer-Verlag, 1991, 62-76. [.pdf]

D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, Locally Random Reductions: Improvements and Applications, Journal of Cryptology, 10 (1997), pp. 17-36. [.pdf] [.ps]

D. Beaver, Commodity-Based Cryptography (extended abstract); Proceedings of the 29th **STOC**, ACM, 1997, 446-455. [.pdf]

D. Beaver and S. Haber, Cryptographic Protocols Provably Secure Against Dynamic Adversaries (extended abstract); Advances in Cryptology - Eurocrypt '92, Springer-Verlag, 1993, 307-323. [.pdf]

D. Beaver, J. Feigenbaum, and V. Shoup, Hiding Instances in Zero-Knowledge Proof Systems (extended abstract), in Advances in Cryptology - Crypto '90, Lecture Notes in Computer Science, vol. 537, Springer, Berlin, 1991, pp. 326-338. [.pdf]

D. Beaver, S. Micali, and P. Rogaway, The round complexity of secure protocols; Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing, (STOC 90), 1990, 503-513. [.ps] [.ps.gz]

D. Beaver, Foundations of Secure Interactive Computing (extended abstract); Advances in Cryptology - Crypto '91 Proceedings, Springer-Verlag, 1992, 377-391. [.pdf]

D. Beaver and S. Goldwasser, Multiparty Computation with Faulty Majority, Advances in Cryptology: Crypto '89, ed. Gilles Brassard. [.pdf]

D. Beaver and N. So, Global, Unpredictable Bit Generation Without Broadcast (extended abstract); Advances in Cryptology - Eurocrypt '93, Springer-Verlag, 1994, 424-434. [.pdf]

D. Beaver, Efficient Multiparty Protocols Using Circuit Randomization (extended abstract); Advances in Cryptology - Crypto '91 Proceedings, Springer-Verlag, 1992, 420-432. [.pdf]

D. Beaver, How to Break a "Secure" Oblivious Transfer Protocol (extended abstract); Advances in Cryptology - Eurocrypt '92, Springer-Verlag, 1993, 285-296. [.pdf]

D. Beaver, J. Feigenbaum, R. Ostrovsky, and V. Shoup, Instance-Hiding Proof Systems; submitted for journal publication. Available as DIMACS Technical Report 93-65, Rutgers University, Piscataway, 1993. [.ps.Z]

R. Beigel and J. Feigenbaum, On Being Incoherent Without Being Very Hard, Computational Complexity, 2 (1992), pp. 1-17.

A. Beimel, Y. Isahi, T. Malkin, and E. Kushilevitz, One-way functions are essential for single-server private information retrieval, Proc. of the 31st Annu. ACM Symp. on the Theory of Computing (STOC), pp. 89-98, 1999. [.ps]

A. Beimel and B. Chor, Secret Sharing with Public Reconstruction, IEEE Trans. on Info. Theory, 44 (5):1887-1896, 1998. Extended abstract in Crypto '95. [.ps]

A. Beimel and M. Franklin, Reliable communication over partially authenticated networks, Theoretical Computer Science, (220)1:185--210, 1999. Preliminary version in WDAG '97, volume 1320 of LNCS, pages 245-259, Springer, 1997. [.ps]

A. Beimel, Secure Schemes for Secret Sharing and Key Distribution, Ph.D. Thesis, Dept. of Computer Science, Technion, 1996. [.ps]

A. Beimel, T. Malkin, and S. Micali, The All-or-Nothing Nature of Two-Party Secure Computation, CRYPTO '99., vol. 1666 of LNCS, pages 80 - 97, 1999. [.ps]

A. Beimel and B. Chor, Universally ideal secret sharing schemes. IEEE Trans. on Info. Theory, 40 (3):786-794, 1994. Extended abstract in Crypto '92. [.ps]

Ius mentis
Law and technology explained

Category: Top > Technology > Encryption 01 October 2005

The ElGamal public key system

(Nederlandse versie)

The ElGamal cryptographic algorithm is a public key system like the Diffie-Hellman system. It is mainly used to establish common keys and not to encrypt messages.

In this document
- Introduction
  - Generating the ElGamal public key
  - Encrypting and decrypting messages

See also
- The Diffie-Hellman system
- The RSA public key cryptographic system
- Elliptic curve cryptography

Introduction

The ElGamal cryptographic algorithm is comparable to the Diffie-Hellman system. Although the inventor, Taher Elgamal, did not apply for a patent on his invention, the owners of the Diffie-Hellman patent (US patent 4,200,770) felt this system was covered by their patent. For no apparent reason everyone calls this the "ElGamal" system although Mr. Elgamal's last name does not have a capital letter 'G'.

A disadvantage of the ElGamal system is that the encrypted message becomes very big, about twice the size of the original message m. For this reason it is only used for small messages such as secret keys.

Generating the ElGamal public key

As with Diffie-Hellman, Alice and Bob have a (publicly known) prime number p and a generator g. Alice chooses a random number a and computes $A = g^a$. Bob does the same and computes $B = g^b$.

Alice's public key is A and her private key is a. Similarly, Bob's public key is B and his private key is b.

Encrypting and decrypting messages

If Bob now wants to send a message m to Alice, he randomly picks a number k which is smaller than p. He then computes:

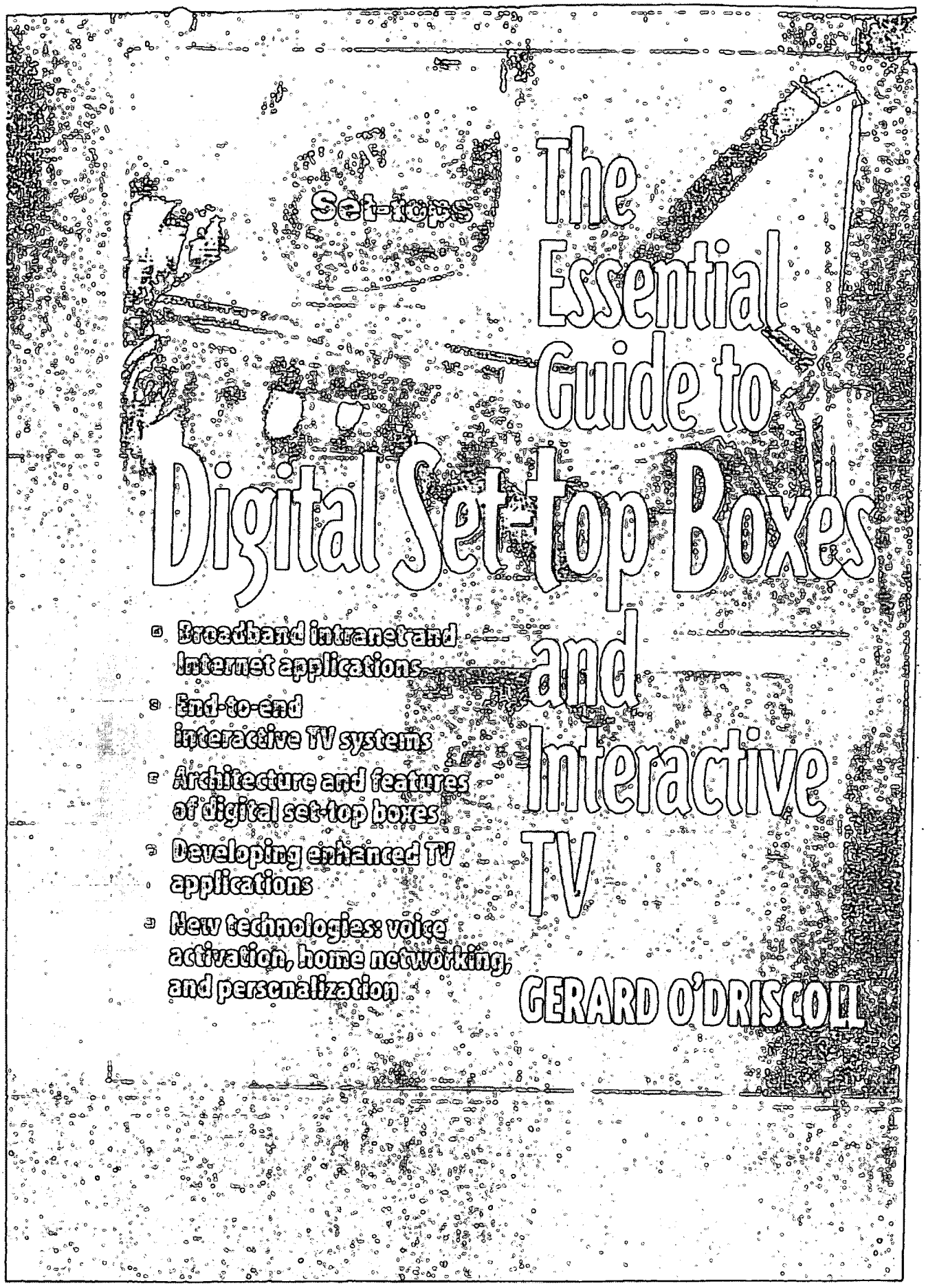
$$c_1 = g^k \bmod p$$
$$c_2 = A^k * m \bmod p$$

and sends c_1 and c_2 to Alice. Alice can use this to reconstruct the message m by computing

$$c_1^{-a} * c_2 \bmod p = m$$

because

$$c_1^{-a} * c_2 \bmod p = (g^k)^{-a} * A^k * m = g^{-a*k} * A^k * m = (g^a)^{-k} * A^k * m = A^{-k} * A^k * m = 1 * m = m$$



passes ETSI projects, technical committees, and special committees. More than 3,500 experts are at present working for ETSI in over 200 groups. (Additional information about ETSI is available from their web site at http://www.etsi.org/).

Digital Video Broadcasting (DVB)

The DVB project was conceived in 1991 and was formally inaugurated in 1993 with approximately 80 members. Today, the DVB project has made huge advancements and boasts a membership of over 230 organizations in more than 30 countries worldwide.

Members of the group include electronic manufacturers, network operators, broadcasters, software companies, and various regulatory bodies.

The DVB project has been a big success and has generated various standards for delivering digital TV to people throughout Europe, Asia, Australia, and North America.

The work of the DVB project has resulted in a comprehensive list of technical and nontechnical documents that describe solutions for implementing digital television in a variety of different environments.

The international standards and solutions developed by DVB over the past few years can be classified and summarized as follows:

1. DVB-S—An international standard for transmitting digital television using satellites.
2. DVB-C—An international standard for transmitting digital television using digital cable systems.
3. DVB-T—An international standard for transmitting digital television in a terrestrial environment.
4. DVB-MC/S—An international standard for transmitting digital television using microwave multipoint video distribution systems.
5. DVB-SI—An international standard that defines the data structures that accompany a digital television signal.
6. DVB-CA—An international standard that defines digital television security standards.
7. DVB-CI—An international standard that defines a common interface to the digital TV security system.
8. DVB-I—An international standard for deploying interactive TV.
9. DVB-Data—An international standard designed to allow operators to deliver software downloads and high speed data services to their customers.
10. Interfaces—An international standard that defines digital TV interfaces to high speed backbone networks.

Figure 1.1
DVB Logo

Copies of these standards are available for download on ETSI's web site.

DVB-compliant digital equipment is widely available and is easily identified by the DVB logo illustrated in Figure 1.1. The DVB has had its greatest success in Europe, however the standard has implementations in North and South America, Africa, Asia, and Australia. For additional information about DVB, visit their web site at http://www.dvb.org/.

Advanced Television Systems Committee (ATSC)

The ATSC committee was formed to establish a set of technical standards for broadcasting standard and High Definition Television (HDTV). Pictures based on this standard can have 3 to 5 times the sharpness of today's analog broadcasts.

The committee is composed of 136 member organizations, standard bodies, IT corporations, educational institutions, and electronic manufacturers. It has been formally adopted in the United States, where an aggressive implementation of digital TV has already begun. In addition to the U.S., Canada, South Korea, Taiwan, and Argentina have also adopted the ATSC digital TV standard for terrestrial broadcasts. A sample of the ATSC standards are outlined in Table 1.1.

Table 1.1 ATSC Standard Documents

Document Number	Standard Description	Brief Overview	Web Address of Detailed Document
A/52	ATSC Digital Audio Compression	Specifies coded representation of audio information and the decoding process, as well as information on the encoding process	www.atsc.org/Standards/A52/
A/53	ATSC Digital Television Standard	Specifications and characteristics for an advanced TV (ATV) system	www.atsc.org/Standards/A53/
A/54	ATSC Guide	Description of ATV system	www.atsc.org/Standards/A54/
A/64	Transmission measurement and compliance for digital television	Description of measurement and ATSC compliance system	www.atsc.org/Standards/A64/

This table only displays a snapshot of the ATSC standards. To review the complete listings of ATSC standards, we recommend you visit the ATSC Web page at http://www.atsc.org/Standards/stan_rps.html for a more detailed listing.

For the latest information and updates about ATSC, visit their web site at http://www.atsc.org/.

Digital Audio Visual Council (DAVIC)

The organization was formed in 1994 with the aim of defining standards for the end-to-end transfer of digital audio, video, and Internet-based content.

DAVIC is a nonprofit standards organization currently located in Switzerland. The organization currently has a membership of over 180 companies from 25 countries around the globe, representing companies and individuals from all sectors of the audio-visual industry. DAVIC members meet on a regular basis to define specifications and use their web site (www.davic.org) to collaborate and implement various international projects.

European Cable Communications Association (ECCA)

ECCA is the European Association of cable operators. The main goal of the Association is to foster cooperation between operators, and to promote their interests

at a European level. ECCA gathers European cable operators, consisting of more than 40 million subscribers. The first informal cooperation between European cable operators started in 1949. As these informal meetings became more frequent, a formal structure for European cooperation was required and on September 2, 1955, the Alliance Internationale de la Distribution par câble (AID) was set up by representatives of Switzerland, Belgium, and The Netherlands. In 1993, AID was renamed the European Cable Communications Association, thus stressing the communication role of its members as well as its European goals.

ECCA now has 29 members in 17 countries. It also has 5 associate members in central and eastern Europe. ECCA has considerably contributed to European policies related to cable on the regulatory as well as on the technical standards field.

On the regulatory, ECCA has done a lot of work on areas such as digital TV, copyright, must-carry, and open-access issues. In addition to these projects, ECCA members have also compiled the following technical specifications.

Eurobox

On initiative of the ECCA organization, a common specification for cable set-top boxes following DVB standards was agreed upon by a large number of cable operators in Europe (the Eurobox platform).

The Eurobox platform was set up in 1997, and has more than 5.5 million subscribers. A more detailed description of the Eurobox is available in Chapter 5 of this book.

Euromodem

A collective resolution to develop a global standard for high speed cable modems was signed at the ECCA Cable Forum in November 1998. The standard fully complies with European standards and with several DVB specifications. The ECCA group has considered two different types of modems: class A and class B. Class A modems are capable of transmitting data at very high speeds in a downstream direction (maximum of 50.8 Mbits/sec) and 3 Mbits/sec in the upstream direction. They are capable of accessing the Internet at high speeds and support a number of security technologies. Class B is the second type of modem considered by the group. It extends the functionality of class A devices through the support of time critical services such as video conferencing and telephony. At the time of going to press, a number of electronic manufacturing companies were invited to submit plans to manufacture modems compliant with the Euromodem standard.

Cable telephony

On the basis of the full liberalization of the telecommunications sector in Europe, cable companies, satellite providers, and terrestrial broadcasters in different countries are planning to become competitors to the local telephony companies. Therefore, their networks are being or have been upgraded to broadband telecommunications networks, which are able to provide all kinds of services from telephony and local Internet access to high speed broadband connections. ECCA is also actively working in this area. For additional information about ECCA, visit their web site at http://www.ecca.be/.

CableLabs

Cable Television Laboratories, Incorporated (CableLabs), was originally established in May 1988 as a research and development consortium of cable television system operators. To qualify as a member of CableLabs, a company needs to be a cable television system operator. CableLabs currently represents more than 85 percent of the cable subscribers in the United States, 70 percent of the subscribers in Canada, and 10 percent of the subscribers in Mexico. CableLabs plans, funds, and implements a number of research and projects that help cable companies take advantage of future opportunities in the areas of digital TV, telephony, and high speed Internet. For additional information about CableLabs, visit their web site at http://www.cablelabs.com/.

W3 Consortium (W3C)

The W3 Consortium (W3C) was originally founded in 1994 to lead the World Wide Web to its full potential by developing common protocols that promote its evolution and ensure its interoperability. The organization is an international consortium, jointly hosted by the Massachusetts Institute of Technology in the U.S; an organization in Europe called the Institut National de Recherche en Informatique et en Automatique, and Keio University in Japan.

The consortium provides a range of services, including: a repository of information about the World Wide Web for developers and users; reference code implementations to embody and promote standards; and various prototype and sample applications to demonstrate use of new technology. For detailed information about the W3C, visit their web site at http://www.w3c.org/.

Federal Communications Commission (FCC)

The Federal Communications Commission (FCC) is an independent United States government agency, directly responsible to Congress. The FCC was established by the

Communications Act of 1934 and is charged with regulating interstate and international communications by radio, television, wire, satellite, and cable. The FCC's jurisdiction covers the 50 states, the District of Columbia, and U.S. possessions. There are six operating bureaus. The bureaus are: Mass Media, Cable Services, Common Carrier, Compliance and Information, Wireless Telecommunications, and International. These bureaus are responsible for developing and implementing regulatory programs; processing applications for licenses or other filings, analyzing complaints, conducting investigations, and taking part in FCC hearings.

The Cable Services Bureau was established in 1993 to administer the cable Television Consumer Protection and Competition Act of 1992. The Bureau enforces regulations designed to ensure that cable rates are reasonable under the law. It is also responsible for regulations concerning "must carry," retransmission consent, customer services, technical standards, home wiring, consumer electronics, equipment compatibility, indecency, leased access, and program access provisions. The Bureau also analyzes trends and developments in the industry to assess the effectiveness of the cable regulations. For additional information about the FCC, visit their web site at http://www.fcc.gov/.

BUILDING BLOCKS OF A DIGITAL TV SYSTEM

A TV operator normally receives content from a variety of sources, including local video, cable, and satellite channels. The content needs to be prepared for transmission to the customer's home by passing the signal through a digital broadcasting system. The diagram in Figure 1.2 depicts the basic building blocks of a digital broadcasting system.

Note that the components shown in this diagram are logical units and do not necessarily correspond to the number of physical devices that are deployed in a total end-to-end digital solution. The role of each component shown in Figure 1.2 is briefly outlined in the following categories.

Compression and Encoding

Central to a digital video-broadcasting network is the compression system, whose job is to deliver high quality video and audio to consumers using a small amount of network bandwidth. The main goal of any compression system is to minimize the storage capacity of information. This is particularly useful for service providers who want to "squeeze" many digital channels into a digital stream.

A compression system consists of *encoders* and *multiplexers*. Encoders are devices used to digitize, compress, and scramble a range of audio, video, and data channels. Digital encoders allow TV operators to broadcast several high quality video

Figure 1.2
Simplified block diagram depicting the basic building blocks of a digital broadcasting system

programs over the same bandwidth that was formerly used to broadcast just one analog video program.

Once the signal is encoded and compressed, an MPEG-2 stream is transmitted to the multiplexer (MPEG-2 is an acronym for Moving Pictures Experts Group). This group has defined a range of compression standards and file formats, including the MPEG-2 video animation system. MPEG-2 is generally accepted in 190 countries worldwide as the standard for digital video compression. There are two major MPEG standards available on the market today: MPEG-1 and MPEG-2.

The MPEG-1 file format is normally used by interactive TV developers to create TV "stills" and has a quality level slightly less than conventional video cassette recorders. The MPEG-2 file format is used in a digital broadcasting environment and features CD-quality audio complemented with a high screen resolution. Once the signal has been compressed into MPEG-2 format, the multiplexer combines the outputs from the various encoders together with the security and program information and data into a single digital stream.

Modulation

Once the digital signal has been processed by the multiplexer, it is now time to amalgamate the video, audio, and data with the carrier signal in a process called *modulation*. The unmodulated digital signal outputted from the multiplexer has only two possible states, either a "zero" or a "one." By passing the signal through a modulation process, a number of states are added, which increases the data transfer rate. The modulation technique used by TV operators will depend on the geography of the franchise area and the overall network architecture.

The three major types of digital modulation are Quadrature Amplitude Modulation, Quadrature Phase Shift Keying, and Coded Orthogonal Frequency Division Multiplexing.

Quadrature Amplitude Modulation (QAM)

QAM is a relatively simple technique for carrying digital information from the TV operator's broadcast center to the customer. This form of modulation modifies the amplitude and phase of a signal to transmit the MPEG-2 transport stream. QAM is the preferred modulation scheme for cable companies because it can achieve transfer rates up to 40 Mbits/sec.

Simplified block diagram depicting the basic building blocks of a digital broadcasting system

Quadrature Phase Shift Keying (QPSK)

QPSK is more immune then QAM to electromagnetic noise and is normally used in a satellite environment or on the return path for a cable television network. QPSK works on the principle of shifting the digital signal so that it is out of phase with the incoming signal. QPSK will improve the robustness of a network, however, this modulation scheme is only capable of transmitting data at 10 Mbits/sec.

Coded Orthogonal Frequency Division Multiplexing (COFDM)

COFDM operates extremely well in heavily built-up areas where digital transmissions become distorted by obstacles such as buildings, bridges, and hills. COFDM is different to QAM because it uses multiple signal carriers to transfer information from one node on the network to another. At the moment, COFDM may be implemented with either 2,000 (2K) or 8,000 (8K) carrier signals. European terrestrial and MMDS operators mainly use the COFDM modulation scheme. In contrast, COFDM has not been deployed in the United States because the ATSC (Advanced Television Systems Committee) has defined a digital terrestrial system that meets the needs of a less-rugged geographical terrain.

Conditional Access System

Broadcast and TV operators are now interacting with their viewers on many levels, offering them a greater program choice than ever before. Additionally, the deployment of a security system or conditional access (CA), as it is commonly called, provides them with unprecedented control over what they watch and when. A CA system is best described as a virtual gateway that allows viewers to access a new world of digital services.

The main goal of any CA system is to control subscribers' access to digital TV pay services and secure the operators revenue streams. Consequently, only customers that have a valid contract with the network operator can access a particular service. Using today's CA systems, network operators are able to directly target programming, advertisements, and promotions to subscribers by geographical area, market segment, or according to personal preferences. The CA system is therefore a vital aspect of the digital TV business. In technical terms, the key elements of the CA system are illustrated in Figure 1.3.

Restricting access to a particular service is accomplished by using a technique called cryptography. It protects the digital service by transforming the signal into an unreadable format. The transformation process is known as "encryption" in a digital environment and "scrambling" in an analog domain. Once the signal is encrypted, it can only be decrypted by means of a digital set-top box. Decryption is the process

Figure 1.3
Basic principle of an end-to-end conditional access system

used to convert the message back to its original format. This is carried out using a decryption key. A key is best described as a secret value, consisting of a random string of bits, which is used by a computer in conjunction with mathematical formulas called algorithms to encrypt and decrypt information.

The box incorporates the necessary hardware and software subsystems to receive and decrypt the signal. These components are comprised of a de-encryption chip, a secure processor, and some appropriate hardware drivers. The de-encryption chip is responsible for holding the algorithm section of the CA. The secure processor can either be soldered onto the set-top box's printed circuit board or else attached to a smart card. Smart cards are plastic cards that look like credit cards. This processor contains the necessary keys needed to decrypt the various services. Chapter 11 discusses the cryptography aspects of smart card security in more detail.

A given subscriber may decrypt and access the digital signal only if the subscriber has purchased the relevant entitlement. As an example, the entitlement may be provided in the form of an electronic smart card that is plugged into the set-top box. Alternatively, in a pay-per-view scenario, the entitlement may be delivered electronically by entitlement management messages (EMMs) and entitlement control messages (ECMs) within the broadcast stream. An EMM is used to carry authorization details and are subscriber-specific. Consequently, the number of EMMs that need to be sent over the broadband network is proportional to the number of set-tops on the network. In addition to sending EMMs to specific customers, operators can also broadcast EMMs to groups of subscribers in different geographical areas. ECMs, on the other hand, carry program- and service-specific information, including control words that are used by the smart card to decrypt the relevant program. However, if a subscriber is not entitled to watch the program, then a signal is sent to the set-top box to indicate that this program has not been authorized for de-encryption. ECMs and EMMs are generated and broadcasted at the TV operations center using specialized hardware devices. They are then transmitted to the viewer's smart card. The card will check access rights and descramble the requested digital services. It is possible to change the value of an ECM every 10 seconds in order to maximize security on a digital network. A typical smart card is capable of storing up to a hundred entitlement messages, which means that each subscriber on the network is capable of ordering 100 pay-TV events at any one time.

In addition to encrypting digital services, the CA also interfaces with the following subsystems:

Subscriber Management System (SMS)

To exploit the commercial potential of digital broadcasting, TV operators need to interface their technical systems with a subscriber management system (SMS). The SMS provides the support required to accurately manage the digital TV business model. It handles the customer database and sends requests to the subscriber autho-

rization system (SAS)—the technical management part of the CA system. Functions typically provided by an SMS software application system include:

- register, modify, and cancel subscriber records;
- targeted marketing campaigns;
- inventory management of set-tops and smart cards;
- customer experience tracking;
- cross-selling of services;
- interfacing with banks and credit card companies;
- fault management;
- multilingual and mulficurrency capability;
- bill preparation and formatting;
- presentation of bills in electronic formats; and
- accounting and auditing facilities.

Many of the software solutions currently available in the marketplace are capable of supporting the increasing variety of interactive services offered to subscribers. The main goal of any SMS system is to ensure that subscribers view exactly what they pay for.

Subscriber Authorization System (SAS)

The main task of the SAS is to translate the requests coming from the SMS into EMMs. These authorization messages are then sent via the digital multiplex to the smart card, which is located in the set-top box. They are sent to customers on a regular interval (for example, every month) to renew subscription rights on the smart card. In the case of Pay Per View (PPV) applications, the SAS sends a certain amount of electronic tokens to the smart card that will allow customers to purchase a variety of PPV events. The SAS contains database(s) that are capable of storing the following items of information:

- pay TV product information,
- data to support the electronic TV guide,
- identification numbers of smart cards,
- customer profiles, and
- scheduling data.

Additionally, SAS security can be enhanced by periodically changing the authorization keys broadcasted to the subscriber base. Some well-known CA systems include:

- CryptoWorks from Philips,
- Viaccess from France Telecom,
- Nagra from NagraVision,
- MediaGuard from Canal+ Technologies,
- VideoGuard from NDS,
- DigiCipher from General Instruments, and
- Iredeto from MindPort.

Network Transmission Technologies

Several different technologies have been deployed to bring broadband entertainment services from a central point to customers on a digital TV network. The different distribution systems (or mix of systems) adopted to broadcast digital TV services in countries around the world has largely been a function of each market's unique characteristics, including elements such as topography, population density, existing broadcast infrastructure, as well as social and cultural factors.

The most popular of these technologies are detailed in the following subsections.

Digital Via Hybrid Fiber-Coax (HFC)

Hybrid fiber-coax (HFC) technology refers to any network configuration of fiber-optic and coaxial cable that may be used to redistribute a variety of broadband entertainment services. These broadband services include telephony, interactive multimedia, high speed Internet access, video-on-demand, and distance learning. The types of services provided to consumers will vary between cable companies.

Many of the major cable television companies in the United States, Europe, Latin America, and Southeast Asia are already using it. Networks built using HFC technology have many characteristics that make it ideal for handling the next generation of communication services. First and foremost, HFC networks can simultaneously transmit broadband analog and digital services. This is extremely important for network operators who are rolling out digital TV to their subscribers on a phased basis. Additionally, HFC meets the expandable capacity and reliability requirements of a new digital TV system. HFC's expandable capacity allows network operators to add services incrementally without major changes to the overall plant infrastructure. HFC is essentially a "pay as you go" architecture that matches infrastructure investment with new revenue streams, operational savings, and reliability enhancements. The HFC network architecture is comprised of fiber transmitters, optical nodes, fiber and coaxial cables, and distribution hubs. An end-to-end HFC network is illustrated in Figure 1.4.

band entertainmen
. The different dig
tal TV services im
rket's unique char
ity, existing broad

owing subsection

ation of fiber-optic
band entertainment
multimedia, high
e types of service

ed States, Europe
s built using HFC
ig the next genera
can simultaneous
important for net
on a phased basis
requirements of a
k operators to add
nfrastructure. HFC
ructure investmen
nhancements. The
al nodes, fiber and
rk is illustrated in

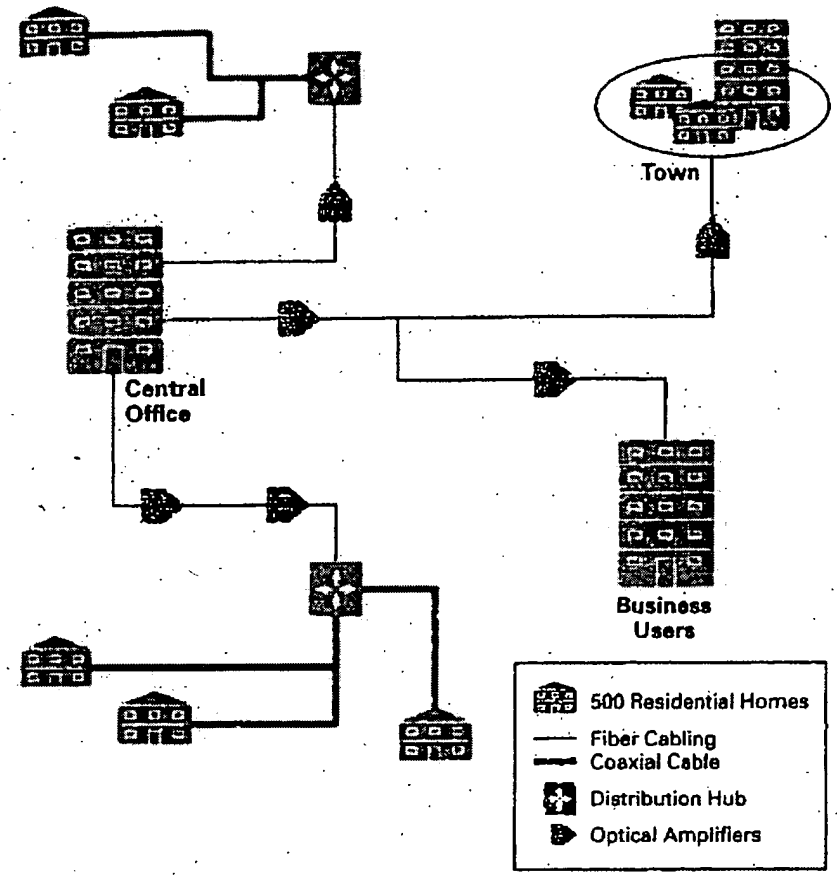


Figure 1.4
End-to-end HFC Network

From the diagram we can see that the signal is transmitted from the central office in a star-like fashion to the fiber nodes using fiber-optic feeders. The fiber node, in turn, distributes the signals over coaxial cable, RF amplifiers, and taps throughout the customer serving area. In conclusion, HFC is the lowest-cost alternative available in terms of cost-per-home-passed. This fact, combined with the other advantages already discussed, ensures that HFC will remain the primary technology for distributing advanced broadband services in a cabled environment.

Digital via Wireless Cable

Wireless cable is a relatively new service used to broadcast TV signals at microwave frequencies from a central point or head-end to small antennas located on the subscriber's roof. It is enabled through the use of two distribution technologies: multichannel multipoint distribution system (MMDS) and local multipoint distribution system (LMDS).

MMDS

Analog-based MMDS began in the mid-1970s with the allocation of two channels for sending business data. The service, however, became very popular for TV subscriber programming and applications were made to allocate part of the ITFS (Instructional Television Fixed Service) band to wireless cable TV. Once the regulations had been amended, it became possible for a wireless cable system to offer up to thirty-one 6 MHz channels in the 2.5 to 2.7 GHz band. During this timeframe, the system was used by nonprofit organizations to broadcast educational and religious programs. In 1983, the FCC allocated frequencies in both of these spectrums, providing 200 MHz bandwidth for licensed network providers. The basic components of an end-to-end digital MMDS system is shown in Figure 1.5.

An MMDS system consists of a head-end that receives signals from satellites, fiber optic cable, off-the-air TV stations, and local programming. At the head-end, the signals are mixed with commercials and other inserts, scrambled, converted to the 2.1 and 2.7 GHz frequency range, and sent to microwave towers. The signals are then rebroadcast from low-powered base stations within a 35-mile diameter of the subscriber's home. Signals are received with home rooftop antennas, which are 18 to 36 inches wide. The receiving antenna should have a clear line of site to the transmitting antenna. A down converter, usually a part of the antenna, converts the microwave signals into standard cable channel frequencies. From the antenna, the signal travels to a set-top box where it is decrypted and from there the signal passes into the television. If the subscriber requires interactivity, then the digital set-top box is also connected to the public telephone network.

Today, there are systems in use all around the U.S. and in many other countries, including Australia, South Africa, South America, Ireland, and Canada. Currently, MMDS is an analog service providing about 20 channels of programming to subscribers. Digital MMDS increases the number of channels to between 130 and 180. Digital MMDS also reduces the line-of-sight restrictions by providing a more efficient signal that will require less signal strength at the set-top box. Digital signals will need about 100 times less signal strength than analog signals, which translates to a substantial increase in the range of service area. Where an analog signal degrades with distance, the digital signal will remain constant and perfect as long as it can be received. In addition to more channels, digital MMDS customers will also be able to receive a variety of Internet, telephony, and interactive TV-based services. MMDS is presently using a standard phone line for the return path, but trials are under way to utilize a portion of the wireless bandwidth for return capabilities.

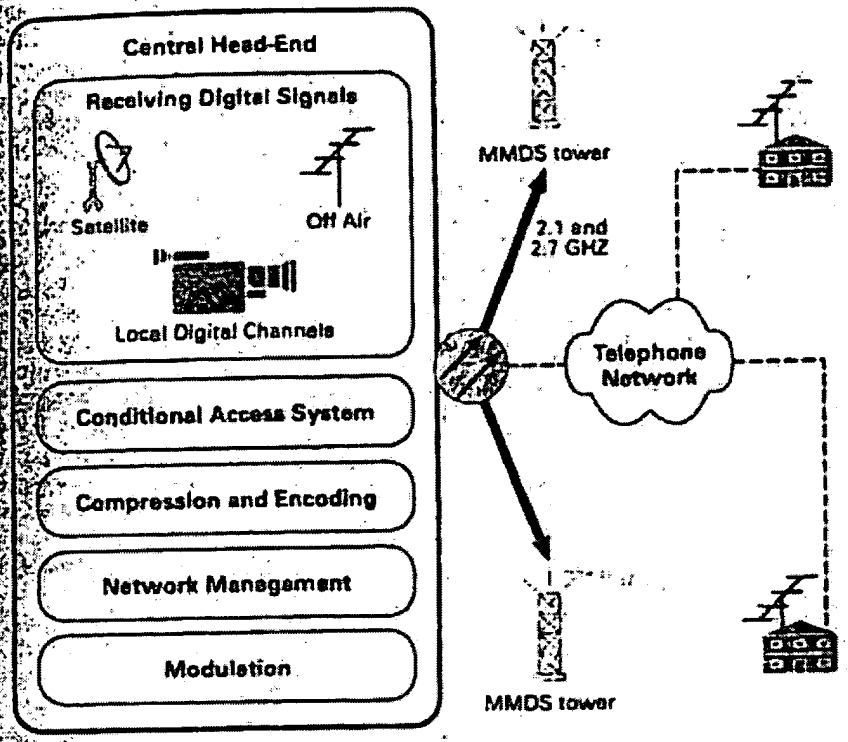


Figure 1.5
End-to-end digital MMDS solution

In Ireland for example, MMDS operators are currently very active in testing and delivering a diversity of advanced digital TV and Internet services using MMDS network transmission techniques to customers across the island. The services on offer to customers include:

- high speed access to the Internet;
- private data networks for companies on the island;
- broadcast video and Pay Per View television;
- Plain Old Telephone Service (POTS); and
- fractional and full leased lines

Future services discussed by Irish operators include video conferencing and delivering multimedia training courses to remote parts of the country using advanced MMDS digital technologies. MMDS operators across the world are adopting similar approaches to their Irish counterparts and are poised to take advantage of the exciting new digital MMDS broadcasting revolution, allowing the delivery of a variety of services to their customer bases.

LMDS

LMDS uses microwave frequencies in the 28 GHz frequency range to send and receive broadband signals, which are suitable for the transmission of video, voice, and multimedia data. Digital LMDS has been commercially deployed and is used to deliver video programming from local and cable channels. Additionally, it is also capable of delivering a plethora of Internet- and telephony-based services to consumers. The system architecture for LMDS is very similar to the MMDS system. The reception and processing of programming and other head-end functions are the same. The signals are then rebroadcasted from low-powered base stations in a 4–6 mile radius of the subscriber's home. Signals are then received using six square-inch antennas, which can be mounted either inside or outside the home. As with the MMDS, the signal travels to the set-top box, decrypted, and formatted for display on the customer's television. In addition to a high video and audio quality, other benefits of LMDS include its bandwidth range of 1 GHz and the availability of a return channel for interactive TV services.

Digital via Terrestrial

Commercially launched in the U.K. in November 1998, terrestrial communications, or DTT as it is commonly called, can also be used to broadcast a range of digital services.

Elements of a terrestrial communications network include:

1. Transmission medium—Services are normally provided via the ultra high frequency band (UHF). The frequencies in this band range from 300 MHz up to 3 GHz. Standard 8 MHz channels are used and shared with analog transmissions.
2. Modulation scheme—DTT uses the COFDM modulation scheme. The main purpose of COFDM is to make the terrestrial signal immune to multipath reflections. In other words, the signal needs to be robust enough to traverse geographical areas that include mountains, trees, and large buildings.
3. Transmission infrastructure—Uses an existing network of broadcast stations and transmitters.
4. Customer's premises equipment—With a modern aerial, there should be no need to replace it to receive the DTT service. If the aerial is a very old one, the viewer would certainly benefit from updating. Additionally, DTT

necessitates the purchase of a new digital set-top box to receive and decode the digital signal.

Digital via Direct Broadcast Satellite (DBS)

Digital television is also available through direct broadcast satellite (DBS), which can provide higher bandwidth than terrestrial, MMDS, or cable transmission.

Direct Broadcast Satellite (DBS) is a service whereby you receive subscription television from a single high-powered satellite. This satellite is typically located about 22,000 miles above the surface of the earth. At the moment, when you subscribe to an analog service you receive a state-of-the art mini-dish that is maintained and owned by the local distributor, along with a decoder for your television set that unscrambles the signals received from the satellite. This year, consumers will be able to receive digital satellite service by installing a new and smaller digital satellite dish and buying a new digital satellite set-top box. Digital via DBS brings consumers more channels to choose from, new features, and new services.

Network Management

As you can see the broadcasting center is made up of many complex components. As these components handle more and more services, network problems must be quickly detected and resolved. To maximize system uptime and monitor the services delivered to customers, a network monitoring and control system is installed at the broadcasting center. The main goal of such a system is to minimize service interruptions to digital TV customers. Features of a typical head-end control system include:

- monitoring the availability of devices,
- gathering statistics,
- reporting alarms and problems to support personnel, and
- remote diagnostics.

The systems available at present are vendor-specific and will run on either Windows NT or UNIX platforms.

SUMMARY

Digital television brings about many challenges, but with those challenges come a lot of opportunities. Advances in technology over the past few years have meant that the

possibility of delivering digital television services to billions of people around the globe has moved from the realms of fantasy into reality.

Digital TV offers a potential mechanism through which every home, school, business, and community center in the world could be included in the information society.

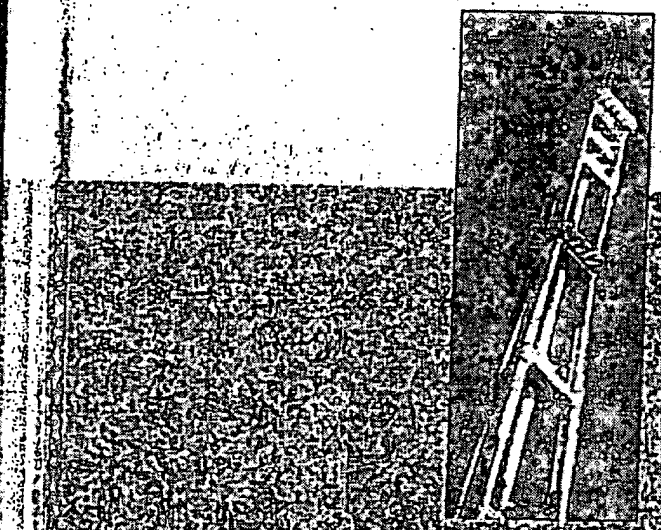
It opens up a new world of opportunity for companies to develop and utilize their existing network infrastructures. This includes broadcasters; cable and satellite companies; the creative community in television and film; Internet content providers; web site producers; and new, innovative companies that will form around the future of digital TV. The broadcast of digital TV and multimedia data works well because of the agreements and partnerships forged by a number of organizations around the world. A complete digital broadcasting system is comprised of a number of building blocks including the compression, encoding, and modulation-system, a CA system for security purposes; network transmission media to deliver the digital services; and, finally, a network management system to detect and resolve problems.

2

In t

- Evolu
- Set-tc
- Basic
- How a
- Under
- Install
- Troubl
- Summ

UNDERSTANDING DIGITAL TERRESTRIAL BROADCASTING



AH DIGITAL AUDIO AND VIDEO SERIES

measure such a low bit error rate at point 3. This long measurement time is not practical, and this is one of the reasons why point 2 (instead of point 3) is used as a measurement point for transmission system quality.

However measuring the BER alone could mask underlying transmission problems if the carrier-to-noise ratio is not also measured. This is because digital systems employing Reed Solomon error correction can be close to total system failure and yet display very little residual errors. This phenomenon is due to the very steep nature of the characteristic BER versus C/N ratio curve for systems measured at point 2. It is often termed the cliff-effect as the BER is seen to fall from very high levels to extremely low levels within a few dB variation of C/N ratio. The huge variation of BER for a relatively small change in carrier-to-noise ratio means that a system, which is not showing significant degradations (low BER), could be on the verge of failure (on the cliff edge). If then, for any reason the carrier-to-noise ratio were degraded by a small amount the overall transmission channel could fail.

9.6.1.3 Measurement Point 3

Measurements made at point 3 are at the output of the demodulator and are the least useful for technical evaluation of the transmission system and channel. This is because these measurements are made after using all of the powerful error correction techniques that DTV can employ to remove errors. There is a risk that the error correction techniques could mask any channel impairments or transmission system degradations. Point 3 should not be used for transmission quality evaluation as it will inevitably lead to problems when link budgets become reduced due to climatic effects or transmission system degradations [4].

9.5 Set Top Boxes (STB) and Integrated Receiver Decoders (IRD)

A set top box (STB) or set top unit (STU) is a device which can translate digital television signals to a format that can be interpreted and displayed by existing analog television receivers. STBs are sometimes referred to as digital set top boxes (DSTB). It includes all of the hardware, middleware, and access entitlement software to allow the consumer to decode all the available digital video, audio, and data. It allows access to all of the digital

based content and services using existing analog television receivers as the display unit. An integrated receiver decoder (IRD) contains also a display unit. Most IRDs available to date can only demodulate the terrestrial COFDM modulation, as traditionally receivers were manufactured to decode off-air or terrestrial signals. Also they are very new additions to the consumer electronics market and at present most can only decode unencrypted DTV signals. However they should become more popular as the DTV market consolidates.

9.5.1 Set Top Box Functionality

Set top boxes are widely available for DTV and in the case of the COFDM based DVB-T system, both the 2k and the 8k systems are now supported. A typical STB will connect to an existing analog television receiver using standard interconnections such as a SCART connector and/or an RF loop through connection, and in a similar manner connect to a video cassette recorder. (The predominant interconnection standard for domestic equipment is the SCART connector, which derives its name from the French committee that has promoted its use.) Many STBs use telephone modems as the return channel, however other terrestrial based systems will be deployed in the future for the return channel. Stereo audio in either digital or analog format can be output to, for instance, a HiFi system. The internal MPEG-2 decoder typically supports Main Profile@Main Level with data rates per service of up to 15 Mbps (see Chapter 4). Current STB units generally support widescreen video format. The hardware can be manufactured to support various application programming interfaces, and conditional access systems. At the moment the network operator financially supports most STB deployment and this is leading to the emergence of proprietary STB systems for different transmission media. In Europe the DVB common interface (CI) is not mandatory for STB. This interface allows for the interconnection of DVB equipment and subassemblies.

9.5.2 Integrated Receiver Decoder (IRD) Functionality

At present, a few manufacturers have begun offering complete DTV receivers, a configuration that is called an integrated receiver decoder. These devices allow for the reception and decoding of DTV without the need for a STB. Some offer flat screen displays and all can decode the unencrypted signals at the moment. They incorporate primarily the demodulation of the terrestrial based transmission standards.

9.6 Middleware and Application Programming Interfaces (API)

There is no doubt that one of the most important topics for broadcasters and service providers at the present time is the topic of set top box middleware and application programming interface (API). In order to provide data services the set top box requires a layer of software that resides between the STB hardware and the application which is to be consumed. This software is called "middleware" and is often referred to as an API when used in connection with DTV. The term API is commonly used in software engineering but in DTV engineering some middleware solutions incorporate the API, but some do not. Some commentators refer to the API as the equivalent to "Windows" for the STB. It can be a helpful analogy for those new to this terminology.

It should be noted that STBs having different middleware might not be able to access the same data or interactive services. Each service must be designed or authored for each particular middleware system that it is intended to run on. This is a major problem for traditional analog broadcasting organizations entering the DTV market [5]. Clearly a fragmented STB market with different STB manufacturers using different middleware will mean that the same material will need to be authored many times to run on different STBs. This is recognized as a potential disaster for DTV and as a result DVB wish to adopt a standard for middleware known as the DVB multimedia home platform (DVB-MHP). This API will meet the need for the next generation of STB, including interactive applications, and Internet access from the STB. Other middleware solutions are proposed in the meantime until the DVB-MHP is developed and standardized. It is beyond the scope of this book to explore the issues relating to middleware and APIs.

9.6.1 DVB and Java

The DVB group has decided to use the Java programming language as the core specification for the software of the DVB-MHP (multimedia home platform). Java is a powerful programming language that should allow the implementation of new applications in a platform independent manner. Providing a so-called virtual machine environment, that is, a software entity that processes applications in the same way on any microprocessor in which it is implemented, does this. This will allow many different hardware realizations of the same applications. With Java, the virtual machine is commonly called a Java virtual machine (JVM).

9.6.2 The MHEG API

MHEG is a sister organization of MPEG within ISO/IEC JTC1 (see Section 3.2). The multimedia and hypermedia information expert coding group (MHEG) develops standards primarily for the transmission and representation of data and applications to allow for interactive services on set top boxes. The MHEG-5 standard has been chosen in the United Kingdom implementation of DTV and is now in use. It is an open and nonproprietary standard for the set top box API.

9.6.3. The EuroMHEG System

The EuroMHEG is a development of the MHEG-5 API to include extensions to the standard. The purpose of these extensions is to provide greater functionality to the consumer, within an open standard API. These functions include provision for a return path in various ways, and a financial toolkit to allow for home shopping and e-commerce. Downloading of different text fonts and text input from a remote control or keyboard are also supported. The EuroMHEG API has other functions which are a development of the original open standard MHEG-5 API, and as such is regarded as a migratory step from MHEG-5 toward the DVB-MHP. This is useful for broadcasters and network operators who wish to launch a DTV service prior to the standardization of the DVB-MHP.

9.6.4 Data Carousels

Data carousels are of interest to DTV service providers and network operators. A carousel is defined as a rotating magazine, for example slides in a projector, or luggage on a rotating conveyor belt. In broadcasting carousels have been used for a long period of time in conjunction with analog services such as teletext, where the contents of the carousel are cyclically repeated and the repetition rate is quite low. The data is broadcast to some defined playout, and the simplest playout is a carousel or rotation of information. If a receiver wants to access a particular module it simply awaits the next time the data from that module is broadcast. These simple data broadcasting techniques are popular with the public, despite the basic nature of the service provided.

It is expected that DTV carousels will be able to improve on these carousels and approach near multimedia presentation, through the better delivery of text, support for audio-visual clips and bitmaps, easier navigation, better graphics, and better scheduling.

Data broadcast according to the data carousel specification of the MPEG-2 DSM-CC is transmitted in a data storage media command and control (DSMCC) data carousel. The DVB data broadcast specification for data carousels supports data broadcast services, which require the periodic transmission of data modules through DVB compliant networks. Again, the application decoder must await the transmission of a particular module to access the data contained within that module.

For the support of interactive services DVB has adopted another part of the DSM-CC specification, known as the object carousel, which provides the facility to transmit structured groups of objects from a broadcast server to specific receivers. This provides enhanced capability as compared to the above mentioned data carousel [6].

9.6.5 Resident and Interactive Applications

A part of the middleware is often referred to as an application, and consists of software code which is used to allow a user to interact with various services and products. If the application is resident, that is, resides within the set top box, then it can be loaded into memory when the set top box is initially switched on. It may allow for tuning of the STB to various channels, set-up configuring, and also manage any interactions between the user and the service provider. Alternatively interactive applications require the user to authenticate the requested service or product from the provider before downloading, and hence control of access to the application remains with the service provider.

9.6.5.1 Electronic Program Guide (EPG)

The electronic program guide has been available in limited form with analog television, however it is expected to change dramatically with DTV. With DTV it allows the consumer to navigate through the various services and programs offered by the network operator or service provider from a graphical user interface (GUI). The EPG is regarded as a very powerful tool in guiding the consumer through the vast amount of services and programs offered in a DTV platform. The EPG will be able to present choices to the consumer based not only on a channel by channel basis (as is the case with analog television), but also by categorization of program content into different interest groups. It enables the consumer to create a viewing schedule based on content. It contains some of the information contained within a traditional paper based television listing service.

The EPG will use service information (SI) as the basis of most of the information that it will display (see Chapter 5). However, it will need to be collated and presented to the consumer in a responsible and coherent manner. Competition issues may arise on a platform between different service providers if undue prominence of a particular service is shown over others. Therefore some degree of regulation in the provision of the EPG is expected.

Some of the features that will be available from an EPG include program guides for a number of days in advance. Program browsers will display the current and next programs on all available services and allow for immediate switching of channels. These browsers will be available for both audio and video services. It will also allow for notification of events on other channels while the consumer is viewing a completely separate channel. Interactive services will be accessed through the EPG, for instance access to the Internet, email, e-commerce, and games, to mention but a few. The power of the EPG is realized when the complete platform is available through a single EPG.

References

[1] ETR 290. "Digital Video Broadcasting (DVB); Measurement Guidelines for DVB Systems." May 1997.

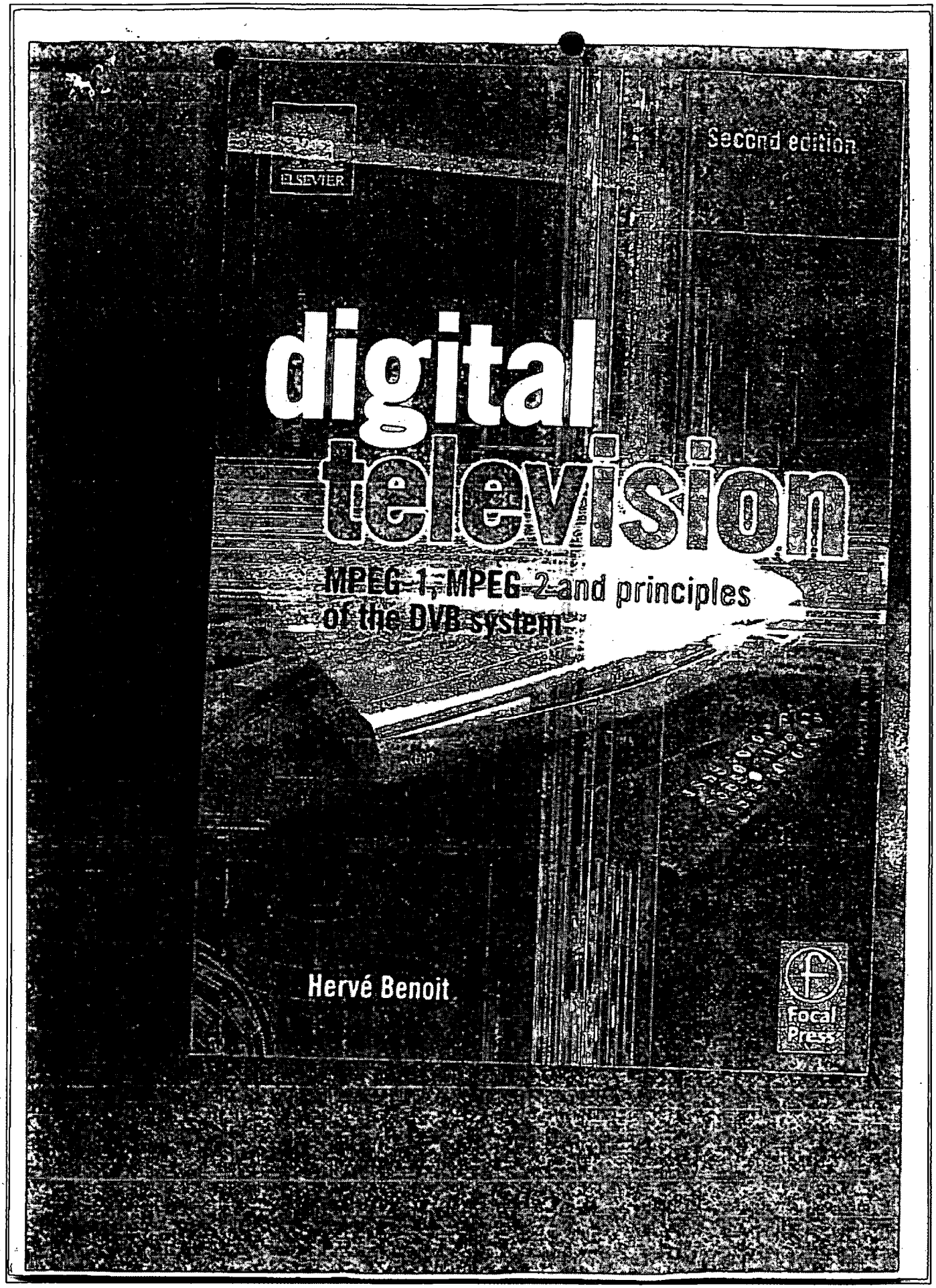
[2] DVB Document TM1748, Draft MG 119 Rev. 1, "DVB-T Measurement Guidelines." September, 1996.

[3] ITU-T Recommendation 0.151: "Error Performance Measuring Equipment Operating at the Primary Rate and Above."

[4] Nokes, C., "Bit Error Rates for DVB-T Signal," *Digital News*, Digital TV Group, No. 7, February 1999, p. 18.

[5] AGITS/WG7 Document, "Digital Television—A Preliminary Guide," 1999, http:/www.teltec.dcu.ie/agitswg7/dtv/

[6] Horst, H., "DVB Data Broadcasting: Building the Info Highway," *World Broadcast News*, Special Supplement, November 1998, pp. 16–21.


ELSEVIER
Second edition
digital
television
MPEG-1, MPEG-2 and principles
of the DVB system
Hervé Benoit
Focal
Press

5 Scrambling and conditional access

The proportion of free access programmes among analogue TV transmissions by cable or satellite is decreasing continuously, at the same time as their number increases; hence, it is almost certain that the vast majority of digital TV programmes will be pay-TV services, in order to recover as quickly as possible the high investments required to launch these services. Billing forms will be much more diversified (conventional subscription, pay per view, near video on demand) than what we know today, made easier by the high available bit-rate of the system and a 'return channel' (to the broadcaster or a bank) provided by a modem.

The DVB standard, as explained in the previous chapter, envisages the transmission of access control data carried by the conditional access table (CAT) and other private data packets indicated by the program map table (PMT). The standard also defines a common scrambling algorithm (CSA) for which the trade-off between cost and complexity has been chosen in order that piracy can be resisted for an appropriate length of time (of the same order as the expected lifetime of the system).

The conditional access (CA) itself is not defined by the standard, as most operators did not want a common system, everyone guarding jealously their own system for both commercial (management of the subscribers' data base) and security reasons (the more open the system, the more likely it is to be 'cracked' quickly). However, in order to avoid the problem of the subscriber who wishes to access networks using different conditional access systems having a stack of boxes (one set-top box per network), the DVB standard envisages the following two options:

1. **Simulcrypt.** This technique, which requires an agreement between networks using different conditional access systems but the same scrambling algorithm (for instance the CSA of the DVB), allows access to a given service or programme by any of the conditional access systems which are part of the agreement. In this case, the transport multiplex will have to carry the conditional access packets for each of the systems that can be used to access this programme.

2. **Multicrypt.** In this case, all the functions required for conditional access and descrambling are contained in a *detachable module* in a PCMCIA form factor which is inserted into the transport stream data path. This is done by means of a standardized interface (common interface, DVB-CI) which also includes the processor bus for information exchange between the module and the set-top box. The set-top box can have more than one DVB-CI slot, to allow connection of many conditional access modules. For each different conditional access and/or scrambling system required, the user can connect a module generally containing a smart card interface and a suitable descrambler.

The multicrypt approach has the advantage that it does not require agreements between networks, but it is more expensive to implement (cost of the connectors, housing of the modules, etc.). The DVB-CI connector may also be used for other purposes (data transfers for instance).

Only the future will tell us which of these options will be used in practice, and how it will be used.

5.1 Principles of the scrambling system in the DVB standard

Given the very delicate nature of this part of the standard, it is understandable that only its very general principles are available, implementation details only being accessible to network operators and equipment manufacturers under non-disclosure agreements.

The scrambling algorithm envisaged to resist attacks from hackers for as long as possible consists of a cipher with two layers, each palliating the weaknesses of the other:

- a *block layer* using blocks of 8 bytes (reverse cipher block chaining mode);
- a *stream layer* (pseudo-random byte generator).

Table 5.1 Meaning of transport_scrambling_flag bits

Transport_scrambling_flags	Meaning
00	No scrambling
01	Scrambling with the DEFAULT control word
10	Scrambling with the EVEN control word
11	Scrambling with the ODD control word

The scrambling algorithm uses two control words (even and odd) alternated with a frequency of the order of 2 s in order to make the pirate's task more difficult. One of the two encrypted control words is transmitted in the entitlement control messages (ECM) during the period that the other one is in use, so that the control words have to be stored temporarily in the registers of the descrambling device. There is also a *default* control word (which could be used for free access scrambled transmission) but it is of little interest.

The DVB standard foresees the possibility of scrambling at two different levels (transport level and PES level) which cannot be used simultaneously.

Scrambling at the transport level

We have seen in the preceding chapter (Fig. 4.6) that the transport packet header includes a 2-bit field called 'transport_scrambling_flags'. These bits are used to indicate whether the transport packet is scrambled and with which control word, according to Table 5.1 above.

Scrambling at transport level is performed after multiplexing the whole payload of the transport packet, the PES at the input of the multiplexer being 'in the clear'. As a transport packet may only contain data coming from one PES, it is therefore possible to scramble at transport level all or only a part of the PES forming part of a programme of the multiplex.

Scrambling at the PES level

In this case, scrambling generally takes place at the source, before multiplexing, and its presence and control word are indicated by the 2-bit PES_scrambling_control in the PES packet header, the format of which is indicated in Fig. 4.4. Table 5.2 indicates the possible options.

Table 5.2 Meaning of PES_scrambling_control bits

PES_scrambling_control	Meaning
00	No scrambling
01	No scrambling
10	Scrambling with the EVEN control word
11	Scrambling with the ODD control word

The following limitations apply to scrambling at the PES level:

- the header itself is, of course, not scrambled; the descrambling device knows where to start descrambling due to information contained in the PES_header length field, and where to stop due to the packet_length field;
- scrambling should be applied to 184-byte portions, and only the last transport packet may include an adaptation field;
- the PES packet header should not exceed 184 bytes, so that it will fit into one transport packet;
- the default scrambling word is not allowed in scrambling at the PES level.

5.2 Conditional access mechanisms

The information required for descrambling is transmitted in specific conditional access messages (CAM), which are of two types: entitlement control messages (ECM) and entitlement management messages (EMM). These messages are generated from three different types of input data:

- a *control_word*, which is used to initialize the descrambling sequence;
- a *service_key*, used to scramble the control word for a group of one or more users;
- a *user_key*, used for scrambling the service key.

ECM are a function of the control_word and the service_key, and are transmitted approximately every 2 s. EMM are a function of the service_key and the user_key, and are transmitted approximately every 10 s. The process for generating ECM and EMM is illustrated in Fig. 5.1

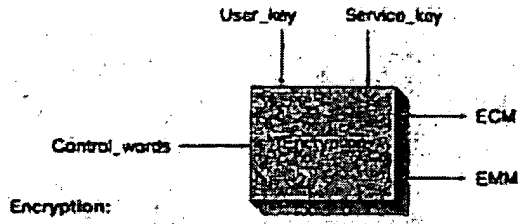


Fig. 5.1 Schematic illustration of the ECM and EMM generation process

In the set-top box, the principle of decryption consists of recovering the service_key from the EMM and the user_key, contained for instance in a smart card. The service_key is then used to decrypt the ECM in order to recover the control_word allowing initialization of the descrambling device. Figure 5.2 illustrates schematically the process for recovering control_words from the ECM and the EMM.

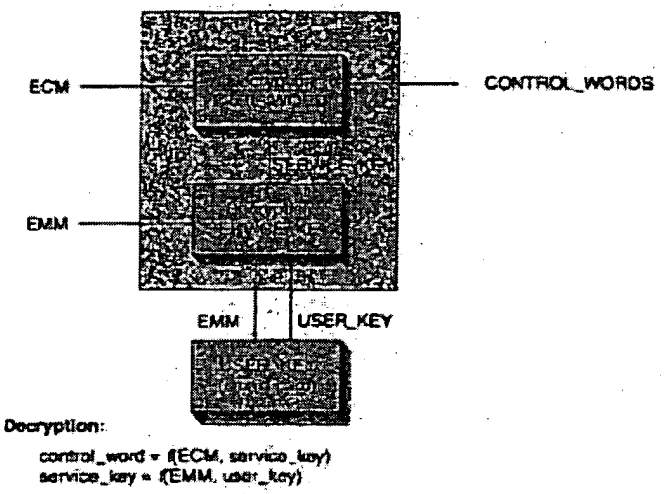


Fig. 5.2 Principle of decryption of the control words from the ECM and the EMM

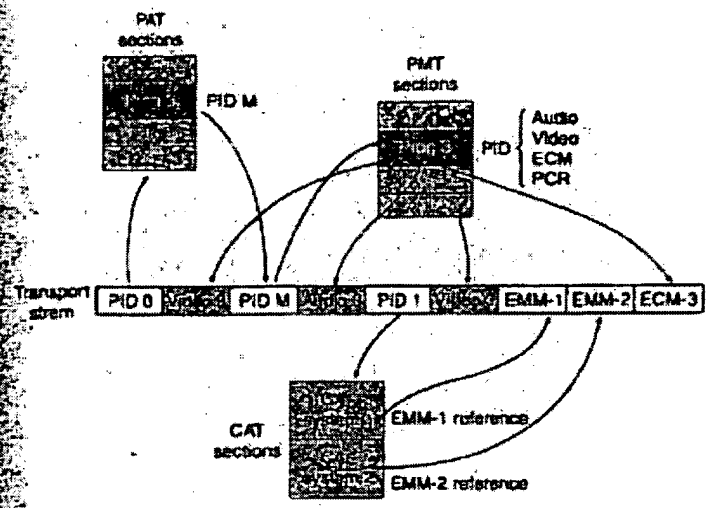


Fig. 5.3 Process by which the EMM and the ECM are found in the transport stream

Figure 5.3 illustrates the process followed to find the ECM and EMM required to descramble a given programme (here programme no. 3):

1. the program allocation table (PAT), rebuilt from sections in packets with PID = 0 x 0000, indicates the PID (M) of the packets carrying the program map table (PMT) sections;
2. the PMT indicates, in addition to the PID of the packets carrying the video and audio PESs and the PCR, the PID of packets carrying the ECM;
3. the conditional access table (CAT), rebuilt from sections in packets with PID = 0 x 0001, indicates which packets carry the EMM for one (or more) access control system(s);
4. from this information and the user_key contained in the smart card, the descrambling system can calculate the control_word required to descramble the next series of packets (PES or transport depending on the scrambling mode).

The above-described process is indeed very schematic; the support containing the user_key and the real implementation of the system can vary from one operator to another. The details of these systems are, of course, not in the public domain, but their principles are similar.

5.3 Main conditional access systems

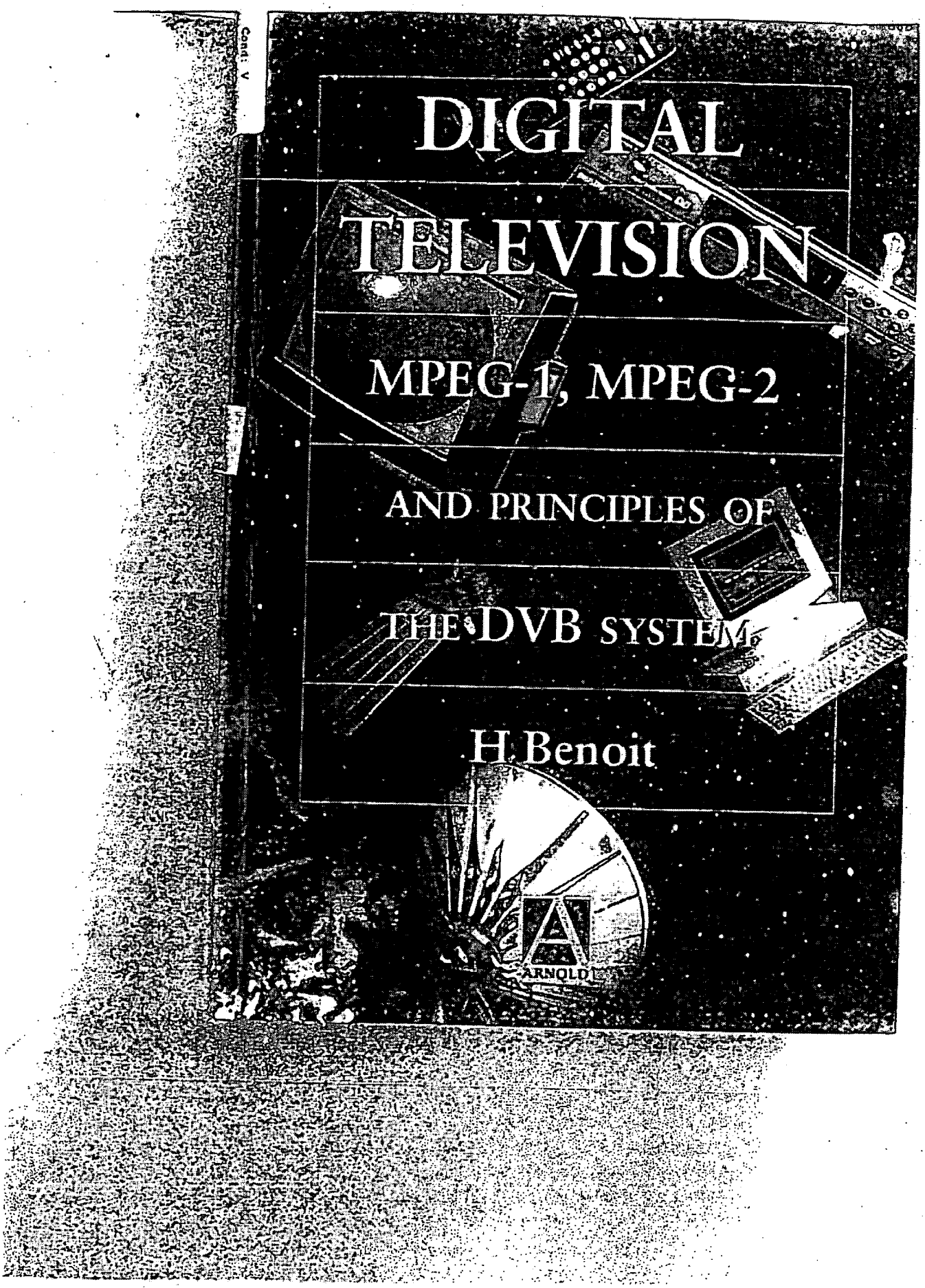
Table 5.3 indicates the main conditional access systems used by European digital pay TV service providers.

Most of these systems use the DVB-CSA scrambling standard specified by the DVB. The receiver has an internal descrambler controlled by an embedded conditional access software which calculates the descrambler control words from the ECM messages and keys contained in a subscriber smart card with valid access rights updated by the EMM messages.

Systems allowing pay per view often have a second card reader slot for a banking card as well as a modem to order the programmes as well as charging the bank account.

Table 5.3 Main conditional access systems

System	Origin	Service providers
Betacrypt	Betaresearch/IrDETO	Premiere World, German cable
CryptoWorks	Philips	Viacom, MTV Networks
IrDETO	IrDETO Access	Telepiu, Stream, Multichoice
Mediaguard	SECA (Canal+)	Canal+, Canal Satellite, ITV Digital
Nagravision	Kudelski SA	Via Digital, Quiero, Dish Network
Viaccess	France Telecom	TPS, AB-Sat, Arabesque, SSR/SRG
Videoguard	News Datacom (NDS)	BSkyB, Stream


DIGITAL
TELEVISION
MPEG-1, MPEG-2
AND PRINCIPLES OF
THE DVB SYSTEM
H.Benoit
ARNOLD

isily the large number of

s not as quick as with
hronization process can
the complex operations
top box, i.e.:

g part (only when chan-

the system clock of the

eal decoding (this alone

5 Scrambling and conditional access

The proportion of free access programmes among analogue TV transmissions by cable or satellite is decreasing continuously, at the same time as their number increases; hence, it is almost certain that the vast majority of digital TV programmes will be pay-TV services, in order to recover as quickly as possible the high investments required to launch these services. Billing forms will be much more diversified (conventional subscription, pay per view, near video on demand) than what we know today, made easier by the high available bit-rate of the system and a 'return channel' (to the broadcaster or a bank) provided by a modem.

The DVB standard, as explained in the previous chapter, envisages the transmission of access control data carried by the conditional access table (CAT) and other private data packets indicated by the program map table (PMT). The standard also defines a common scrambling algorithm (CSA) for which the trade-off between cost and complexity has been chosen in order that piracy can be resisted for an appropriate length of time (of the same order as the expected lifetime of the system).

The conditional access (CA) itself is not defined by the standard, as most operators did not want a common system, everyone guarding jealously their own system for both commercial (management of the subscribers' data base) and security reasons (the more open the system, the more likely it is to be 'cracked' quickly). However, in order to avoid the problem of the subscriber who wishes to access networks using different conditional access systems having a stack of boxes (one set-top box per network), the DVB standard envisages the following two options:

1. **Simulcrypt.** This technique, which requires an agreement between networks using different conditional access systems but the same scrambling algorithm (for instance the CSA of the DVB), allows access to a given service or programme by any of the conditional access systems which are part of the agreement. In this case, the transport multiplex will have to carry the conditional access packets for each of the systems that can be used to access this programme.
2. **Multicrypt.** In this case, all the functions required for conditional access and descrambling are contained in a *detachable module* in a **PCMCIA** form factor which is inserted into the transport stream data path. This is done by means of a standardized interface (common interface, **DVB-CI**) which also includes the processor bus for information exchange between the module and the set-top box. The set-top box can have more than one DVB-CI slot, to allow connection of many conditional access modules. For each different conditional access and/or scrambling system required, the user can connect a module generally containing a smart card interface and a suitable descrambler.

The multicrypt approach has the advantage that it does not require agreements between networks, but it is more expensive to implement (cost of the connectors, housing of the modules, etc.). The DVB-CI connector may also be used for other purposes (data transfers for instance).

Only the future will tell us which of these options will be used in practice, and how it will be used.

5.1 Principles of the scrambling system in the DVB standard

Given the very delicate nature of this part of the standard, it is understandable that only its very general principles are available, implementation details only being accessible to network operators and equipment manufacturers under non-disclosure agreements.

The scrambling algorithm envisaged to resist attacks from hackers for as long as possible consists of a cipher with two layers, each palliating the weaknesses of the other:

- a *block layer* using blocks of 8 bytes (reverse cipher block chaining mode);
- a *stream layer* (pseudo-random byte generator).

The scrambling algorith
odd) alternated with a fre
make the pirate's task mo
control words is transmitt
(ECM) during the period
control words have to be s
descrambling device. Ther
could be used for free acc
little interest.

The DVB standard fores
different levels (transport
used simultaneously.

Scrambling at the transp

We have seen in the prece
packet header includes a 2
flags'. These bits are used
is scrambled and with whic
below.

Table 5.1 Meaning of transport

Transport_scrambling_flags
00
01
10
11

Scrambling at transport
the whole payload of the tr
the multiplexer being 'in
only contain data coming fi
scramble at transport level
part of a programme of th

Scrambling at the PES le

In this case, scrambling ge
multiplexing, and its prese
the 2 bit PES_scrambling_
format of which is indicat
possible options.

The scrambling algorithm uses two control words (even and odd) alternated with a frequency of the order of 2 s in order to make the pirate's task more difficult. One of the two encrypted control words is transmitted in the entitlement control messages (ECM) during the period that the other one is in use, so that the control words have to be stored temporarily in the registers of the descrambling device. There is also a *default* control word (which could be used for free access scrambled transmission) but it is of little interest.

The DVB standard foresees the possibility of scrambling at two different levels (transport level and PES level) which cannot be used simultaneously.

Scrambling at the transport level

We have seen in the preceding chapter (Fig. 4.6) that the transport packet header includes a 2 bit field called 'transport_scrambling_flags'. These bits are used to indicate whether the transport packet is scrambled and with which control word, according to Table 5.1 below.

Table 5.1 Meaning of transport_scrambling_flag bits

Transport_scrambling_flags	Meaning
00	No scrambling
01	Scrambling with the DEFAULT control word
10	Scrabling with the EVEN control word
11	Scrambling with the ODD control word

Scrambling at transport level is performed after multiplexing the whole payload of the transport packet, the PES at the input of the multiplexer being 'in the clear'. As a transport packet may only contain data coming from one PES, it is therefore possible to scramble at transport level all or only a part of the PES forming part of a programme of the multiplex.

Scrambling at the PES level

In this case, scrambling generally takes place at the source, before multiplexing, and its presence and control word are indicated by the 2 bit PES_scrambling_control in the PES packet header, the format of which is indicated in Fig. 4.4. Table 5.2 indicates the possible options.

Table 5.2 Meaning of PES_scrambling_control bits

PES_scrambling_control	Meaning
00	No scrambling
01	No scrambling
10	Scrambling with the EVEN control word
11	Scrambling with the ODD control word

The following limitations apply to scrambling at the PES level:

- the header itself is of course, not scrambled; the descrambling device knows where to start descrambling due to information contained in the PES_header length field, and where to stop due to the packet_length field;
- scrambling should be applied to 184 byte portions, and only the last transport packet may include an adaptation field;
- the PES packet header should not exceed 184 bytes, so that it will fit into one transport packet;
- the default scrambling word is not allowed in scrambling at the PES level.

5.2 Conditional access mechanisms

The information required for descrambling is transmitted in specific conditional access messages (CAM), which are of two types: entitlement control messages (ECM) and entitlement management messages (EMM). These messages are generated from three different types of input data:

- a *control_word*, which is used to initialize the descrambling sequence;
- a *service_key*, used to scramble the control word for a group of one or more users;
- a *user_key*, used for scrambling the service key.

ECM are a function of the control_word and the service_key, and are transmitted approximately every 2 s. EMM are a function of the service_key and the user_key, and are transmitted approximately every 10 s. The process for generating ECM and EMM is illustrated in Fig. 5.1

In the set-top box, the principle of decryption consists of recovering the service_key from the EMM and the user_key, contained

Control_words

Encryption:

ECM = f(control_word,

EMM = f(service_key, u

Fig. 5.1 Schematic illustratio

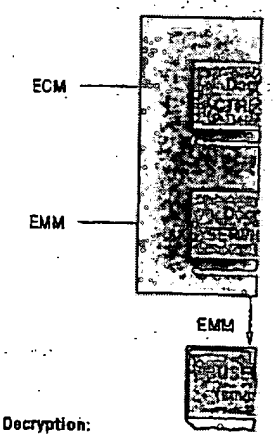


Fig. 5.2 Principle of decryption o

for instance in a smart c
decrypt the ECM in order
initialization of the descran
matically the process for re
and the EMM.

Fig. 5.3 illustrates the p
EMM required to descra
gramme no. 3):

1. the program allocation
   packets with PID = 0 >
   packets carrying the prc

control bits

th the EVEN control word
th the ODD control word

mbling at the PES level:

nbled; the descrambling
ling due to information
d, and where to stop due

te portions, and only the
laptation field;

ed 184 bytes, so that it

ved in scrambling at the

nisms

bling is transmitted in
AM), which are of two
M) and entitlement man-
ages are generated from

tialize the descrambling

ıtrol word for a group of

rvice key.

ord and the service_key,
2 s. EMM are a function
are transmitted approxi-
ating ECM and EMM is

yption consists of recov-
l the user_key, contained

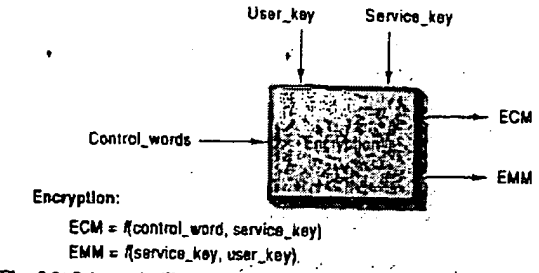


Fig. 5.1 Schematic illustration of the ECM and EMM generation process

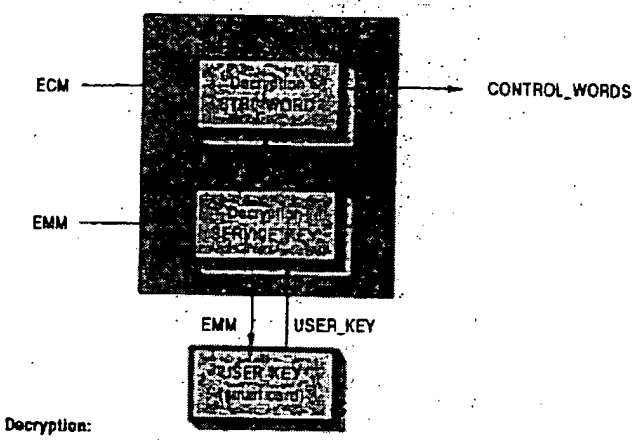


Fig. 5.2 Principle of decryption of the control words from the ECM and the EMM

for instance in a smart card. The service_key is then used to decrypt the ECM in order to recover the control_word allowing initialization of the descrambling device. Fig. 5.2 illustrates schematically the process for recovering control_words from the ECM and the EMM.

Fig. 5.3 illustrates the process followed to find the ECM and EMM required to descramble a given programme (here programme no. 3):

1. the program allocation table (PAT), rebuilt from sections in packets with PID = 0 × 0000, indicates the PID (M) of the packets carrying the program map table (PMT) sections;

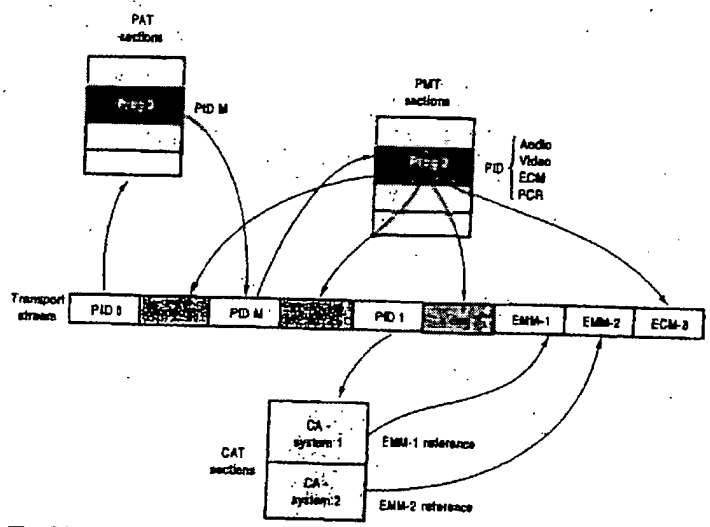


Fig. 5.3 Process by which the EMM and the ECM are found in the transport stream

2. the PMT indicates, in addition to the PID of the packets carrying the video and audio PESs and the PCR, the PID of packets carrying the ECM;
3. the conditional access table (CAT), rebuilt from sections in packets with PID = 0 × 0001, indicates which packets carry the EMM for one (or more) access control system(s);
4. from this information and the user_key contained in the smart card, the descrambling system can calculate the control_word required to descramble the next series of packets (PES or transport depending on the scrambling mode).

The above-described process is indeed very schematic; the support containing the user_key and the real implementation of the system can vary from an operator to another. The details of these systems are, of course, not in the public domain, but their principles are similar.

6 Channe (forwar correct

Once the source coding ope multiplexing and eventuall of 188 byte packets is availa a radiofrequency link (satel

We previously indicated unfortunately, not error-fre disturbances which can c interference, echoes). Hov when almost all its redunda low bit error rate (BER) fo of 10^{-10}–10^{-12}, correspon for a bit-rate of 30 Mb/s) called *quasi-error-free* (QI

It is therefore necessar modulation in order to all correction in the receiver c cal transmission channel. consist of reintroducing a (which obviously reduces t grouped under the terms 'channel coding' (this ter cess). Such measures will physical transmission med

The *virtual channel* thus encoder on the transmiss decoder on the receiver si referred to as a *super cha*

9 Future prospects

The fully digital era in consumer video applications is only just starting, and rapid and numerous changes in the services offered to the public are to be expected in the coming months. Due to the high investments required, it is understandable that these transmissions will start with pay TV services in nearly all countries.

However, even if the DVB standard has a powerful unifying role (27 countries had adopted the standard by mid-1996), in a very similar manner to that pioneered by the GSM standard for the mobile telephone some years ago, it could not impose a common conditional access control in the way GSM did at that time. As a result, a 'war of boxes' has already started between the various European and extra-European players, everyone trying to impose their technology in the field of conditional access and user interface (electronic program guide), and surprising alliances can sometimes be observed.

We will not, therefore, attempt to make any predictions in this field, as the risk of being contradicted by events taking place between the time of writing and the time of reading is far from negligible, and the person who could predict the winner would be very clever indeed. Nevertheless, we will try to list the main foreseeable technical changes in this field up to the turn of the century.

9.1 Terrestrial digital TV

Within Europe, the United Kingdom seems to be the first country willing to start a terrestrial service (as early as 1998, with the 2K

variant of DVB-T). This
the current PAL analog
coexistence between bot

One of the proposals f
'simulcasting' on the sam
of each of the four or five
one possibility would th
versions, one channel at

This would allow the
bandwidth, which could
multiplexes to increase
new bandwidth-hungry
mainly for cost reasons
OFDM modulation sche
the digital European ra
DAB). Its detractors wo
scheme similar to the
forward error correction

A similar approach is
HDTV 'Grand Alliance
replacement of analogu
However, the choice of
will certainly differ fron
states of AM modulatio
of ASK modulation wit
sidebands is almost co
halve the required banc

As long as the techn
ment of analogue tran
chance that the digital
TV set. As a first step,
ably be hybrid (capabl
missions), unless the IC
beginning.

9.2 Evolution of

9.2.1 Functional i

The diagram in Fig. 8.
European IRD of the fi
market in the first half
corresponds to a majo

pects

:o applications is only just
:ges in the services offered
coming months. Due to the
'standable that these trans-
:es in nearly all countries.
d has a powerful unifying
andard by mid-1996), in a
l by the GSM standard for
;o, it could not impose a
ı the way GSM did at that
already started between the
players, everyone trying to
conditional access and user
ınd surprising alliances can

ıake any predictions in this
ed by events taking place
time of reading is far from
›redict the winner would be
/ill try to list the main fore-
ıp to the turn of the century.

eems to be the first country
early as 1998, with the 2K

variant of DVB-T). This new system would eventually replace the current PAL analogue transmissions after 10–15 years of coexistence between both systems.

One of the proposals for the transition period would consist of 'simulcasting' on the same RF channel of 8 MHz a digital version of each of the four or five existing analogue channels; in this case, one possibility would then be to stop transmitting the analogue versions, one channel at a time, over a period of 15 years or so.

This would allow the progressive freeing up of an important bandwidth, which could be reallocated either to new digital TV multiplexes to increase the number of programmes, or even to new bandwidth-hungry communication services. However, mainly for cost reasons, there has been some criticism of the OFDM modulation scheme, already adopted some years ago for the digital European radio system (Digital Audio Broadcasting, DAB). Its detractors would prefer a less robust but cheaper QAM scheme similar to the one used for DVB-C with a reinforced forward error correction.

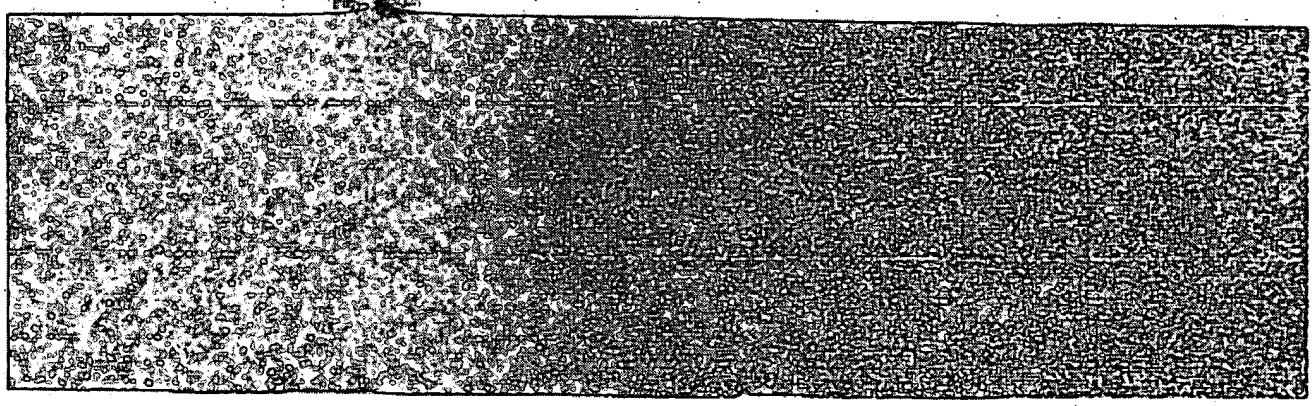
A similar approach is envisaged in the USA, where the digital HDTV 'Grand Alliance' project had as one of its objectives the replacement of analogue NTSC transmissions in the year 2008. However, the choice of the modulation for terrestrial and cable will certainly differ from DVB, since it will probably be 8-VSB (8 states of AM modulation with a vestigial sideband). This is a kind of ASK modulation with more than two states, where one of the sidebands is almost completely removed by filtering in order to halve the required bandwidth for transmission.

As long as the technical and political choices for the replacement of analogue transmissions remain unclear, there is little chance that the digital TV decoders will be integrated into the TV set. As a first step, the sets integrating digital TV will probably be hybrid (capable of receiving analogue and digital transmissions), unless the 100% simulcast approach is chosen from the beginning.

9.2 Evolution of the set-top box

9.2.1 Functional integration

The diagram in Fig. 8.2 represents the functional partitioning of a European IRD of the first generation (available on the commercial market in the first half of 1996), and each of its functional blocks corresponds to a major integrated circuit.



Most of these ICs are fabricated with a C-MOS technology, with geometries of 0.6 or 0.5 µ. As a result of the very rapid progress of the IC technology, we will soon be able to group these functions in circuits with 0.35 µ geometries according to the possible following scheme:

- reduction of the (S)DRAM packages from four of 256 K × 16 to one of 1 M × 16;
- use of a RISC processor integrating a part of the external interfaces (IEEE1284 etc.);
- grouping of the transport stream processing (demultiplexer and descrambler functions) with the RISC processor or perhaps the MPEG-2 audio/video decoder;
- use of a monochip for channel decoding (for satellite as well as for cable), grouping the demodulator and the error correction, and possibly the input ADC(s).

By 1988–1989, in addition to the power supply, the tuner and the necessary memory, such a decoder (a block diagram of which is shown in Fig. 9.1) could consist of three main ICs:

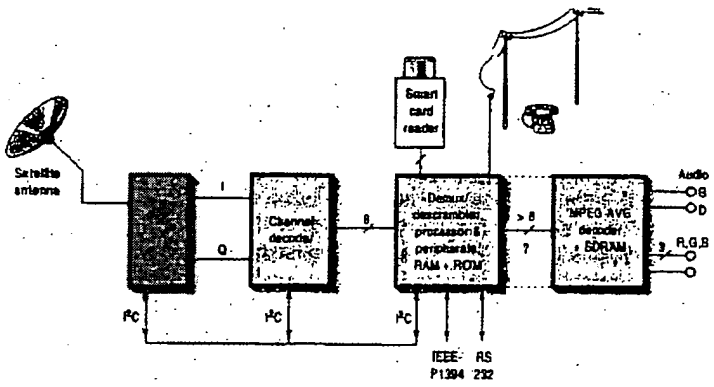


Fig. 9.1 Possible architecture of a DVB satellite receiver in 1997/1998 (doted lines represent possible further integration options)

- channel decoder
- RISC processor with transport stream processing
- source decoder.

Integration will continue ever smaller geometries (0. the end of the century, to th PAL/RGB encoder with t processing. In addition, pr memory management will size required for decodin processor, leaving more r EPG or other application unthinkable that all the ac nel and source decoding) IC'. Only the power suppl of software would then be

9.2.2 Functional evol

In addition to the integrat mainly to reduce the cost added which will increa with the models at the e

- Interfaces to the exter that the analogue PA recording will be enha digital I/O, for instanc der (the IEEE1394 int some important manu perhaps replace the cu is used for data interc
- For IRDs connected to preferred means of ac high speed *return cha* simple telephone line opening the door to co net access could the functions that could modification to the se TV set-top box are, am the international Digit
- Among the list of ne set-top box, the newl digital versatile disk) the existing MPEG-2

th a C-MOS technology,
result of the very rapid
ɔon be able to group these
netries according to the

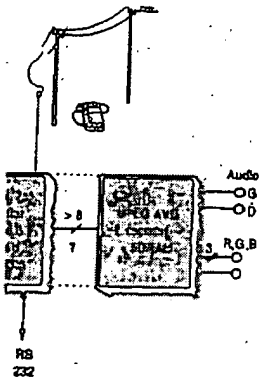
rom four of 256 K × 16

a part of the external

ssing (demultiplexer and
processor or perhaps the

g (for satellite as well as
and the error correction,

supply, the tuner and the
ɔck diagram of which is
main ICs:



er in 1997/1998 (dotted lines
1 options)

ocessing

Integration will continue inexorably after this step, by the use of ever smaller geometries (0.25 µ or less), which will lead, well before the end of the century, to the integration of the source decoder and the PAL/RGB encoder with the microprocessor and transport stream processing. In addition, progress in the decoding algorithms and in memory management will make possible a reduction in the memory size required for decoding, which could then be shared with the processor, leaving more room for high resolution graphics for the EPG or other applications. In much less than 10 years, it is not unthinkable that all the active functions (processor, memory, channel and source decoding) will be integrated into one single 'super IC'. Only the power supply, a tuner, this super IC and a fair amount of software would then be required to build a set-top box.

9.2.2 Functional evolution of the decoder

In addition to the integration of the existing functions, which aims mainly to reduce the cost of the basic IRD, new functions will be added which will increase the appeal to the consumer, starting with the models at the expensive end of the market:

- Interfaces to the external world will evolve, and it is probable that the analogue PAL or SECAM output used mainly for recording will be enhanced (and later replaced) by a high speed digital I/O, for instance, for connection to a digital video recorder (the **IEEE1394** interface seems to be the preferred choice of some important manufacturers, and when introduced it could perhaps replace the current parallel interface, IEEE1284, which is used for data interchange with a PC).
- For IRDs connected to a cable network, which could become the preferred means of access to the 'information superhighway', a high speed *return channel* using the cable network instead of a simple telephone line will provide a much better interactivity, opening the door to completely new services. High speed Internet access could then be one of the possible value-added functions that could be integrated without major hardware modification to the set-top box. These new features of a digital TV set-top box are, among others, being defined and specified by the international Digital Audio Visual Council (**DAVC**).
- Among the list of new functions which could be added to the set-top box, the newly standardized DVD (digital video disk or digital versatile disk) is a high priority, as it could make use of the existing MPEG-2 decoder to realize, at relatively low cost, a

combined IRD/DVD player (or even recorder) within the body of an existing VCR.

- Last, but not least, digital video broadcasting will ease the integration of TV and PC functions as techniques converge, and the new added functions will be more attractive due to the high data throughput offered by the transport channels. This will perhaps bring about the true home multimedia and multipurpose machine announced some years ago.

9.3 Other changes

9.3.1 The future MPEG-4 standard

Work continues within the MPEG group in order to elaborate new digital audio/video standards. The defunct MPEG-3, initially intended as the basis for a digital HDTV standard, was eventually included in the upper levels of MPEG-2. A new MPEG-4 committee has now been created, with the objective of defining a standard for audio/video coding at a very low bit-rate (from 10 kb/s to 1 Mb/s for moving pictures, and from 2 to 64 kb/s for the associated sound!).

This work is not, in principle, aimed at digital TV broadcasting, but at interactive multimedia applications at the interface between audio-visual and computer (radio)communications, such as videotelephony based on LAN, ISDN and standard telephone or radiotelephone networks. For these applications, MPEG-4 is going to define new coding principles, including being object oriented to ease editability and interactivity, and various scalability possibilities to allow easy adaption to a variety of transmission channel capacities.

The objective is to obtain an international standard by the end of 1998, which means that commercial applications based on MPEG-4 will not be available much before the year 2000.

9.3.2 New trends for signal processing

Despite the complexity of the algorithms used, the constant increase in the processing power of the new processors already allows purely software-based full-screen MPEG decoding in real time. Although this approach is definitely not economically attractive at present for a stand-alone set-top box, it is already of interest in some microcomputer based applications.

In recent years and months, new specialized processors have been developed which integrate a certain number of functional

blocks dedicated to mu
VLIW processor (audi
motion estimators, var
processors are generall
sors in a microcomputer
most probably appear f
ing, for instance, all the

The advantage of thi
that a well designed pi
reprogramming, in ver
same board could be
MPEG decoder or a n
pay being the amount o
application).

In conclusion, we wi
book if readers have a
view of the new techn
which will perhaps give
subject which is still in

In order to satisfy the
books much more de
aspects of this vast sub
paration) will most pro
The short bibliography
existing references.

/en recorder) within the body

broadcasting will ease the
ions as techniques converge,
ill be more attractive due to
d by the transport channels.
ie true home multimedia and
l some years ago.

andard

group in order to elaborate new
ie defunct MPEG-3, initially
HDTV standard, was eventually
G-2. A new MPEG-4 committee
ective of defining a standard for
-rate (from 10 kb/s to 1 Mb/s for
kb/s for the associated sound!).
ed at digital TV broadcasting, but
ns at the interface between audio-
lications, such as videotelephony
telephone or radiotelephone net-
G-4 is going to define new coding
oriented to ease editability and
/ possibilities to allow easy adap-
annel capacities.
ternational standard by the end
mercial applications based on
uch before the year 2000.

processing

algorithms used, the constant
r of the new processors already
l-screen MPEG decoding in real
is definitely not economically
-alone set-top box, it is already
er based applications.
iew specialized processors have
a certain number of functional

blocks dedicated to multimedia processing on top of a RISC or VLIW processor (audio, video and communication interfaces, motion estimators, variable length coder/decoder, etc.). These processors are generally used as dedicated multimedia coprocessors in a microcomputer environment, but specific derivatives will most probably appear for use as stand-alone processors, combining, for instance, all the control and decoding functions of an IRD.

The advantage of this approach is its very high flexibility, so that a well designed piece of hardware will be usable, by simple reprogramming, in very different applications (for instance, the same board could be used as a videoconferencing codec, an MPEG decoder or a music editing sytem, the only penalty to pay being the amount of memory required for the most demanding application).

In conclusion, we will have reached our goal at the end of this book if readers have acquired a (hopefully clear enough) global view of the new techniques used in digital television systems, which will perhaps give them the desire to investigate further this subject which is still in its infancy.

In order to satisfy the thirst for knowledge of interested readers, books much more dedicated to and specialized in particular aspects of this vast subject (including the new standards in preparation) will most probably be published in the coming months. The short bibliography given at the end provides some of the existing references.

ROTL 1. Remote Office Test Line. Provides the capability to originate automatic inter office trunk transmission test calls under the automatic control of CAROT. from a remote location.
2. A popular online abbreviation, shorthand for "Rolling On The Floor Laughing"; an appropriate typed response to a particularly amusing online remark. Other common Net acronyms include IMHO ("In My Humble Opinion") and IMHSHO ("In My Not-So-Humble opinion").
Rotor The rotating part of a motor or other electrical machines.
ROTS Rotary Out-Trunk Switches.
Round Cutter These are used to cut cables. The blades of the cutter are curved so that there is a space between them.
Round Robin This is a method of distributing incoming calls to a bunch of people. This method selects the next agent on the list following the agent that received the last call. See also Top Down and Longest Available.
Roundtrip Propagation Delay Roundtrip propagation delay from a burst modem to a burst modem will be about 470 milliseconds to 570 milliseconds (About half a second). See Satellite Transmission Delay.
Routable Protocols Protocols, such as TCP/IP, DECnet, and XNS, that support Network Layer addressing. Packets constructed using these protocols contain information about how data should move through a network. This information, carried in the NLA (Network Layer Address) field of the packet, is used by internetworking devices to make routing decisions.
Route The path that a message takes. In telephone companyese, a route is the particular trunk group or interconnected trunk groups between two reference points used to establish a path for a call. This term (or this term routing) is also used as a verb to define the act of selecting a route or routes.
Route 66 A colloquial term for the Internet, with something interesting everywhere along the way. Route 66 used to be the way we drove across America before they put in concrete highways.
Route Advance This feature routes outgoing calls over alternate long distance lines when the first choice trunk group is busy. The phone user selects the first choice route by dialing the corresponding access code. The phone equipment automatically advances to alternate trunks and trunk groups, based on the user's class of service. Route advance is a more primitive form of least cost routing. See Least Cost Routing.
Route Daeman A program that runs under 4.2 or 4.3BSD UNIX systems (and derived operating systems) to propagate routes among machines in a local area network. Pronounced "route-dee."
Route Discovery Process through which a brouter can learn LAN topology by passing information about its address and the LANs it connects and receiving the same information from others.
Route Diversity See Diversity.
Route Flap Frequent change in state (addition-removal) of customer Internet route announcements. Route flaps are typically caused by BGP session resets; changes in state (on-off) of router; changes in state (up-down) of private line; change to router filter list; high circuit error rate.
Route Guide A map showing how calls are to be routed at the switch: first choice group of circuits, second choice, third choice, and fourth choice.
Route Indicator An address or group of characters in the heading of a message defining the final circuit or terminal to which the message is to be delivered.
Route Length The actual length of a route, or path, between transmitter and receiver. The length of the route is one factor that determines propagation delay, i.e., the delay associated with a signal as it transverses a network. Route length is determined by measuring the actual length of a the path, rather than the distance "as the crow flies."
Route List A sequence of trunk groups that can be searched for a particular route. This list is comprised of trunk groups and configuration attributes (e.g. Class of Service) governing the use of a particular trunk group.
Route Mile Let's say that you have two sheaths of fiber, each of which contains ten fibers and runs for one mile. That is one route mile (total distance of all fibers), two sheath miles (2 sheaths running one mile), and twenty fiber miles (20 fibers running one mile).
Route Optimization Another way of saying Least Cost Routing.
Route Server An ATM term. A physical device that runs one or more network layer routing protocols, and which uses a route query protocol in order to provide network layer routing forwarding descriptions to clients.
Route Xpander Card A board manufactured by IBM for insertion into a PC which provides the PC with a wide area interface to a frame relay network, including handling all of the necessary encapsulation.
Routed Protocol A protocol that can be routed by a router. In do so a routes must understand the logical internetwork as perceived by that routed protocol. Examples of routed protocols include DECnet, AppleTalk, and IP.
Router 1. As in software, router is a system level function that directs a call to an application.
2. As in hardware, routers are the central switching offices of the Internet and corporate Intranets and WANs. Routers are bought by everybody
—from backbone service providers to local Internet Service Providers (ISPs), from corporations to Universities. The crazie provider of routers in the world is Cisco. It has built its gigantic business on selling routers
—from small ones, connecting a simple corporate LAN to the Internet, to corporate enterprise wide networks, to huge ones connecting the largest of the largest backbone service providers. A router is, in the strictest terms, an interface between two networks. Routers are highly intelligent devices which connect like and unlike LANs (Local Area Networks). They connect to MANs (Metropolitan Area Networks) and WANs (Wide Area Networks), such as X.25, Frame Relay and ATM. Routers are protocol-sensitive, typically supporting multiple protocols. Routers most commonly operate at the bottom 3 layers of the OSI model, using the Physical, Link and Network Layers to provide addressing and switching. Routers also may operate at Layer 4, the Transport Layer, in order to assure end-to-end reliability of data transfer.
Routers are much more capable devices than are bridges, which operate primarily at Layer 1, and switches, which operate primarily at Layer 2. Routers send their traffic based on a high level of intelligence inside themselves. This intelligence allows them to consider the network as a whole. How they route (also called routing considerations) might include destination address, packet priority level, least-cost route, minimum route delay, shortest route distance, route congestion level, and community of interest. Routers are unique in their ability to consider an enterprise network as comprising multiple physical and logical subnets (subnetworks). Thereby, they are quite capable of confining data traffic to a subnet, on the basis of privilege as defined in a policy-based routing table. In a switched router topology, each router port defines a physical subnet, and each subnet is a broadcast domain. Within that domain, all connected devices share broadcast traffic; devices outside of that domain can neither see that traffic, nor can they respond to it. Contemporary routers have the ability to define subnets on a logical basis, based on logical address (MAC or IP address) information contained within the packet header, and acted upon in consultation with a programmed routing table. In addition to standalone routers designed specifically for that purpose, server-based routers can be implemented. Such routers, in the form of high-performance PCs with routing software. As software will perform relatively and efficiently than firmware, such devices generally are considered to be less desirable for large enterprise-wide application, although they do serve well in smaller remote offices and less-intensive applications. Routers also are self-learning; they can communicate their existence and can learn of the existence of new routers and LAN segments. Routers constantly monitor the condition of the network so that in order to dynamically adapt to changes in network conditions.
Characteristics of routers can include:
- LAN Extension
- Store & Forward
- Support for Multiple Media
- Support for Multiple LAN Segments
- Support for Disparate LAN Protocols
- Filtering
- Encapsulation
- Accommodation of Various and Large Packet Sizes
- High-Speed Internal Buses (1+ Gbps)
- Self-Learning
- Routing Based on Multiple Factors
- Route Length
- Number of Hops
- Route Congestion
- Traffic Type
- Support for a Community of Interest (VLAN)
- Redundancy
- Network Management via SNMP

Router protocols include both bridging and routing protocols, as

appearance of objects when you roll over them or click on them. See also Rollover Lines.

Rollover Lines You receive many incoming calls. You don't want to miss a call, so you ask your phone company to set your phone lines up to roll over, also called hunt, also called ISG (Incoming Service Group) in telephonese. You order five lines in hunt. The calls come into the first. If the first one is busy, the second rings. If it's busy, the third rings. If they're all busy, then the caller receives a busy. The commonest types of hunting are sequential and circular hunting. Sequential hunting starts at the number dialed, keeps trying one number after another in number order and ends at the last number in the group. It's typically ascending. For example, it starts at 691-8215, goes to 691-8216, then 691-8217, etc. But it can also be ascending — from 691-8217 up. Circular hunting hunts all the lines in the hunting group, regardless of the starting point. Circular hunting, according to our understanding, circles only once (though your phone company may be able to program it circle a couple of times). The differences between sequential and circular are subtle. Circular seems to work better for large groups of numbers. You don't need consecutive phone numbers to do rollovers. Nowadays you can roll lines forwards, backwards and jump around, for example most idle, least idle. Rollovers are now done in software. This also has its downside, since software fails. For example, theoretically if a rollover strikes a dead trunk, it should bounce to the next live trunk. But sometimes it hangs on the dead trunk and many of your incoming calls never get answered. They might ring and ring. They might hit a busy. My recommendation: Test your rollovers at least twice a day. In particular, test that your callers ultimately get a busy if all your lines are busy. Nothing worse your customer should receive a ring-no-answer or a constant busy when calling your company. See also Terminal Number.

ROLM A telephone equipment manufacturer based in Santa Clara, CA, at least once upon a time. ROLM was started in 1969 by four engineers to produce computers for the military. The company introduced one of the first digital PBXs in 1975. It was a great PBX. Later, they developed a line of KTSs (Key Telephone Systems) and hybrid PBX/KTS systems. They were not so good. IBM acquired ROLM in 1984 as part of their plan to integrate the worlds of computers and communications. It didn't work...at all. And IBM lost a lot of money with Rolm. In 1989, IBM sold ROLM to Siemens, at which time it became a ROLM Company. In 1994, the name was changed to Siemens Rolm Communication Inc. In 1996, the name was changed to Siemens Business Communication Systems, Inc. Siemens really doesn't use the name ROLM (or Rolm) anymore, but there are a lot of ROLM systems still in service.

Rolodex A Trademarked product which started life as paper card based device to keep names and address on. Now it has become more of a generic name to connote software to let you look up peoples' phone numbers and addresses. Software to do this is also called PIM — for Personal Information Manager.

ROM Read Only Memory. Computer memory which can only be read from. New data cannot be entered and the existing data is non-volatile. This means it stays there even when power is turned off. A ROM is a memory device which is programmed at the factory and whose contents thereafter cannot be altered. In contrast is the device called RAM, whose contents can be altered. See Read Only Memory and Microprocessor.

ROM Font The ROM Font is your PC's type font. It consists of a set of 256 characters which cannot be edited — unless you are running in video mode, in which case you can design your own type font.

Rom Shadowing 386 and higher CPUs provide memory access on 32 & 64 bit paths. Often they will use a 16 bit data path for system ROM BIOS info. Also some adapter cards (ie. older video, network adapters etc.) with on board BIOS may use an 8 bit path to system memory. For high end computers this is a bottleneck. Like having YIELD signs out on the lanes within a freeway. ROM is very slow, 150ns-200ns. Modern RAM is 60ns or less. Therefore when the system is waiting on this data it generates wait states. For high end computers these wait states slow the entire system down. There is a system developed to transfer the contents of all the slow 8-16 bit ROM chips through out the system into 32 bit faster main memory. "This is ROM SHADOWING". This is accomplished using the MMU, the memory management unit. The MMU takes a copy of the ROM BIOS codes and places it into RAM. To the rest of the system this RAM location looks exactly like the original ROM location. This definition courtesy Charlie Irby, chsirby@foothill.net.

Roofing Filter A low-pass filter used to reduce unwanted higher frequencies.

Room Cut-Off Hotel/motel guest telephones restricted from outgoing calls when the guest room is unoccupied.

Room Status And Selection Provides the capability to store and display the occupancy and cleaning status and the type number of each guest room. This helps housekeeping management, maid locating and room selection. Also, communications between the front desk and the housekeeper are speeded up via real-time maid activity and checkout audit printouts to indicate which rooms need cleaning next. The occupancy status is normally changed by the maid or inspector dialing from the room telephone.

Root The base of a tree. The base of a hard disk. See Root Directory.

Root Directory The top-level directory of a PC disk, hard or floppy. The root directory is created when you format the disk. From the root directory, you can create files and other directories.

Root Web The FrontPage web that is provided by the server by default. To access the root web, you supply the URL of the server without specifying a page name. FrontPage is installed with a default root web named !!!root web@@@. All FrontPage webs are contained by the root FrontPage web.

ROSE Remote Operations Service Element. An application layer protocol that provides the capability to perform remote operations at a remote process. Definition from Bellcore (now Telcordia) in reference to its concept of the Advanced Intelligent Network.

Rostered Staff Factor RSF. A call center term. Alternatively called an Overlay, Shrink Factor or Shrinkage. RSF is a numerical factor that leads to the minimum staff needed on schedule over and above base staff required to achieve your service level and response time objectives. It is calculated after base staffing is determined and before schedules are organized, and accounts for things like breaks, absenteeism and ongoing training.

Rostering A call center term. The practice of rotating employees through all existing schedules in a matrix, or roster, of schedules. This "share the grief" method is prevalent in Europe and Australia, where agents work through an entire roster.

ROT13 A way to encode things that the general Internet community can't read. Each letter in a message is replaced by the letter 13 spaces away from it in the alphabet. There are online decoders to read these. For instance, Harry Newton becomes Uneel Arjgba, which sounds a lot more exotic.

ROTA A call center term. 1. An European term for a rotating shift pattern or rotating schedule, 2. Short form for roster.

Rotary Dial The circular telephone dial. As it returns to its normal position (after being turned) it opens and closes the electrical loop sent by the central office. Rotary dial telephones momentarily break the DC circuit (stop current flow) to represent the digits dialed. The circuit is broken three times for the digit 3. The CO counts these evenly-spaced breaks and determines which digit has been dialed. You can hear the "clicks". The number "seven," for example consists of seven "opens and closes," or seven clicks. You can dial on a rotary phone without using the rotary dial. Simply depress the switch hook quickly, allowing pauses in between to signify that you're about to send a new digit. It's a good party trick.

Rotary Dial Calling The telephone system will accept dialing from conventional rotary dial sets.

Rotary Hunt You buy several phone lines. Let's say 212-691-8215, 212-691-8216, 212-691-8217, 212-691-8218. Someone dials you on your main number — 212-691-8215. It's busy. (That's our number.) The central office slides the call over to 212-691-8216. If that number is busy, it slides it over to 212-691-8217, and so on. This is called rotary hunt. It hunts to the next line in the rotary group. In the old days, the phone lines you could rotary hunt to had to be in numerical sequence. But now with modern stored program control central offices, your lines in rotary hunt can be very different as long as they're all on the same exchange.

Rotary Output To Central Office Most central offices are equipped to provide tone dial service. In cases where the telephone company central office trunks are not designed to accept tone signaling, your on premise phone system (PBX, key system or single line phone) will translate the number entered by a phone in tones into rotary dial pulses which can be processed by the central office.

Rotating Cylinder (Drum) Scanner A scanning technique using a drum and a photocell scan head. The original is attached to the drum, enabling the scan head to travel along the length of the document. Reflected light from the document is concentrated on the scanner photocell, which causes an analog signal.

Rotating Helical Aperture Scanner Original is illuminated by a lamp when fed onto the platen, via a mirror and lens system, the document's image is focused first through a fixed horizontal slot, then through a rotating spiral slit disk series, and finally onto a photocell to generate an analogous electrical current.

Rotational Latency The delay time from when a disk drive's read/write head is on-track and when the requested data rotates under it.

Rotational Mailboxes Information only mailboxes whose information is automatically changed on a time sensitive or usage sensitive basis.

ROTFL I'm "Rolling on the Floor, Laughing." Used in e-mail.

...communications and data transfer. Essentially, NetBIOS is a way for application programs to talk to the network. Other applications interfaces are also being used these days, such as APPC. To run an application that works with NetBIOS, a non-IBM network operating system or network interface card must offer a NetBIOS emulator. More and more hardware and software vendors offer these emulators. They aren't always perfectly compatible, though. Many vendors either provide a version of NetBIOS to interface with their hardware or support its transport layer communications services in their network products.

NETBIOS Emulator An emulator program provided with NetWare that allows applications to run applications that support IBM's NetBIOS calls.

NetBT NBT. NetBios over TCP/IP. A protocol supporting NetBIOS services in a TCP/IP environment, defined by RFCs 1001 and 1002.

Netcasting A method used to deliver Web content automatically to the desktop. Netcasting is referred to as push technology because content is pushed from a Web site to those users who requested receipt of the content. Content can include weather forecasts, stock market quotes, or software updates.

NETCOM A system that provides information on Customer Premises Equipment (CPE), such as equipment, and contracts.

NETDDE Network Dynamic Data Exchange, a feature of Microsoft Windows.

NetFind A white pages service which enables a person to query one service and have it perform search other databases for addresses matching the originally entered query.

NETFX Consortium NetFX consortium is a group of companies that have come together to provide solutions for delivering real-time multimedia over the Internet's World Wide Web. The members are Diamond Multimedia Systems www.diamondmm.com, NETCOM www.netcom.com, SAIC http://mercury.saic.com/demo), Templeton Graphics (http://www-sd.lbl.gov/~template), and Xing Technology Corp www.xingtech.com).

Netgod Also spelled NET.GOD. A person very visible on a network and who played an important role in the development of the network.

NetHead Also spelled net.head. A person who like really loves the Internet.

Nethopper A wonderfully cute name for a device that organizes data to hop from one ...network to another. Nethoppers are also called bridges and routers.

NetHub The home of the TUME network located in Bethesda, MD. All Super-Regional ...on the TUME network call the NetHub to exchange mail packets.

NETI Network Information Table.

NETIPC Network Interprocess Communications. Permits inter-CPU program sharing and ...programs running on different systems to exchange data through a set of program-...calls. This peer-to-peer service is important for developing distributed applications.

Netiquette A pun on "etiquette". Contraction for "Network Etiquette". It means proper ...on Internet. Because the recipients of text-only messages cannot see your face or hear ...nuances of your voice, special care must be taken to avoid misunderstanding and to convey ...flavor" that you intended your messages to have. Some common techniques are the ...(also called emoticons): punctuation that suggests sideways faces, bracketed ...abbreviations like <g> for grin and <G> for Big Grin, and careful capitalization as in I AM NOW ...YOU!

Netizen A citizen of the Internet.

NetKeeper See IPTC.

Netmail A private message transmission capability on FidoNet in which nodes directly ...with one another on a point-to point (in FidoNet) or hub-routed (in PCRelay). ...was originally developed for use by SYSOPS to communicate with one another and ...available on some BBs for regular users. See Echo.

Netmask A colloquial way of saying IP Address Mask. See IP Address Mask.

NetMeeting Microsoft's NetMeeting is a conferencing and collaboration tool ...designed for the Internet or Intranet. I'm giving a lot of space to it in this edition of my dictionary because I believe NetMeeting is an important product that will have a major impact ...the way we communicate (voice, video, data, images, etc.) over the PSTN, over private ...networks and over IP networks, such as the Internet and private IP networks, ...Intranets. NetMeeting, according to Microsoft, has the following features: * ...standards-based voice and video conferencing. Real-time, point-to point audio conferencing over the Internet or corporate Intranet enables a user to make voice calls to associates and organizations around the world. NetMeeting voice conferencing offers many features, including half-duplex and full-duplex audio support for real-time conversations, automatic microphone sensitivity level setting to ensure that meeting participants hear each other clearly, and microphone muting, which lets users control the audio signal sent during a call. This voice conferencing supports network TCP/IP connections. * Support for the H.323 protocol enables interoperability between NetMeeting and other H.323-compatible voice clients. The H.323 protocol supports the ITU G.711 and G.723 audio standards and Internet Engineering Task Force (IETF) RTP and RTCP specifications for controlling audio flow to improve voice quality. On MMX-enabled computers, NetMeeting uses the MMX-enabled voice codecs to improve performance for voice compression and decompression algorithms. This will result in lower CPU use and improved voice quality during a call. With NetMeeting, a user can send and receive real-time visual images with another conference participant using any video for Windows-compatible equipment. They can share ideas and information face-to-face, and use the camera to instantly view items, such as hardware or devices, that the user chooses to display in front of the lens. Combined with the video and data capabilities of NetMeeting, a user can both see and hear the other conference participant, as well as share information and applications. This H.323 standards-based video technology is also compliant with the H.261 and H.263 video codecs. Multipoint data conferencing using T.120. Two or more users can communicate and collaborate as a group in real time. Participants can share applications, exchange information through a shared clipboard, transfer files, collaborate on a shared whiteboard, and use a text-based chat feature. Also, support for the T.120 data conferencing standard enables interoperability with other T.120-based products and services. The following features comprise multipoint data conferencing: Application sharing: A user can share a program running on one computer with other participants in the conference. Participants can review the same data or information, and see the actions as the person sharing the application works on the program (for example, editing content or scrolling through information.) Participants can share Windows-based applications transparently without any special knowledge of the application capabilities. The person sharing the application can choose to collaborate with other conference participants, and they can take turns editing or controlling the application. Only the person sharing the program needs to have the given application installed on their computer. Shared Clipboard: The shared clipboard enables a user to exchange its contents with other participants in a conference using familiar cut, copy, and paste operations. For example, a participant can copy information from a local document and paste the contents into a shared application as part of a group collaboration. File Transfer: With the file transfer capability, a user can send a file in the background to one or all of the conference participants. When one user drags a file into the main window, the file is automatically sent to each person in the conference; they can then accept or decline receipt. This file transfer capability is fully compliant with the T.127 standard. Whiteboard: Multiple users can simultaneously collaborate using the whiteboard to review, create, and update graphic information. The whiteboard is object-oriented (versus pixel-oriented), enabling participants to manipulate the contents by clicking and dragging with the mouse. In addition, they can use a remote pointer or highlighting tool to point out specific contents or sections of shared pages. Chat: A user can type text messages to share common ideas or topics with other conference participants, or record meeting notes and action items as part of a collaborative process. Also, participants in a conference can use chat to communicate in the absence of audio support. A "whisper" feature lets a user have a separate, private conversation with another person during a group chat session. TAPI 3.0 and NetMeeting both support core IP Telephony capabilities. Each platform offers unique benefits: TAPI 3.0 integrates traditional telephony with IP Telephony, providing a COM-based, protocol-independent call-control and data streaming infrastructure. NetMeeting SDK supports T.120 conferencing and application sharing in addition to IP Telephony. Applications using TAPI 3.0 and the NetMeeting API interoperate using H.323 audio and video conferencing. Because TAPI 3.0 and NetMeeting both support core IP Telephony capabilities (including support for H.323), developers may want to consider the following guidelines when choosing an API for their IP Telephony applications: TAPI 3.0. This is the API to use if you are doing IP Telephony in your application. TAPI 3.0 is especially valuable in the world of client/server computer telephony integration, for combining IP Telephony with traditional telephony, and for IP multicast of voice and video. NetMeeting API. This is the API to use if you are doing real-time collaboration and want to integrate voice, video, and data conferencing into your application. The NetMeeting API is useful for applications that want to integrate application sharing, whiteboard functionality, and multipoint file transfer with voice and video sessions. www.microsoft.com/netmeeting.

NETOPS Network Operations System.

NetPC See Network Computer and Thin Client.

Netscape Netscape was once the most famous Web Browser is Netscape. It owes its genesis to a software program called Mosaic, which Marc Andreessen and a small team of student programmers working for $6.85 an hour in 1993 wrote at the National Center for Super-

computing Applications at the University of Illinois at Urbana-Champaign. A year later, after becoming annoyed at the way the University had taken over his Mosaic creation, Mr. Andreessen proposed a "Mosaic Killer" — a new and improved version of his own creation. The team was back at work by April 1994 in a company called Netscape. And by October, they had created a new version of Mosaic, called Netscape Navigator. Netscape went public in August of 1995 in one of the most successful IPOs (Initial Public Offerings) ever. when Microsoft started giving away its Internet Explorer Browser for free, Netscape fell on hard times. And in late 1998, America OnLine (AOL) bought it.

Netserv A file server used for distributing files directly related to the BITNET network.

Netsite The term Netscape Navigator uses to refer to a URL or WWW address.

Netsploitation Flick Any one of the Hollywood films about the Internet.

Netstat A utility program used to show server connections running over TCP/IP (Transmission Control Protocol/Internet Protocol) and statistics, including current connections, failed connection attempts, reset connections, segments received, segments sent, and segments retransmitted.

Netstation In an Internet scenario, thin clients are known as NetPCs or Netstations. The NetStation is reliant on the server, which is provided by your company or a service provider (e.g., America OnLine, CompuServe, or your ISP). In addition to providing some combination of content and Internet access, the service provider's server will provide your client NetPC with access to all necessary applications (e.g., word processing and spreadsheet), will store all your personal files, will provide all significant processing power, and so on. In this Internet example, the Netstation differs from the standard thin client by virtue of the fact that it does contain a modem, a communications port and communications software, all of which are required for Internet access. See also Client, Client/Server, Client/Server Model, Fat Client, Mainframe Server, Media Server, and Thin Client.

Netview An IBM product for management of heterogeneous networks that integrates the functions of three formerly separate Communications Network Management (CNM) software programs: 1. NCCF. Network Communication Control Facility. 2. NLDM. Network Logical Data Manager, which uses functions from NCCF and helps pinpoint problems along the logical connection/path of an SNA session. 3. NPDA. Network Problem Determination Application, which displays various alerts using IBM equipment located at strategic points in the network and allows diagnostic information to be displayed. Also, NetView incorporates some of the functions from two other programs: VNCA (Virtual Telecommunications Access Method/Node Control Application) which monitors the status and current activity of all resources in a domain, and NMPF (Network Management Productivity Facility) which helps the network operator to install, learn and use many network management products. See also Network and Network Management.

NetWare NetWare is an extremely popular and extremely good operating system for a local area network from Novell, Orem, UT. NetWare is actually its own operating system. This means it is the link between machine hardware (file servers, printers, modems, etc.) and people who want it use that hardware. NetWare is neither DOS, nor OS/2 nor Windows though it can be made to look and act like them. That's part (a small part) of its popularity. See Netware MHS, Netware Workstation Files, NETx.COM and Novell.

NetWare Bindery Centralized authentication database for NetWare 3.xx LANs.

NetWare Directory Services. See NDS.

NetWare Global MHS Novell's implementation of MHS as a NetWare Loadable Module (NLM), providing powerful integration with NetWare services. This supports additional modules to connect to X.400, SNA and SMTP systems.

NetWare Loadable Module NLM. An driver that runs in a server on a local area network under Novell's NetWare operating system and can be loaded or unloaded on the fly as it's needed. In other networks, such applications could require dedicated PCs. A telephony NLM might allow a workstation on a LAN to control a PBX attached to a NetWare file server. It might also allow the workstation to control one or more voice processing cards sitting on in a NetWare server. In early 1993, AT&T became the first PBX maker to ink a deal with Novell, the creator of NetWare, to put telephony onto Novell LANs. AT&T created a PC-card resident in a Novell File server. The card connects to the ASAI (Adjunct Switch Applications Interface) BRI port on the AT&T Definity PBX. Anyone with a PC on the Novell network and an AT&T phone on their desk can use telephone features, such as auto-dialing, conference calling and message management (a new term for integrating voice, fax and e-mail on your desktop PC via your LAN). The Novell/AT&T deal intends to create open Application Programming Interfaces (APIs) that third party developers can work with. A Novell/AT&T example of what could be developed: A user could select names from a directory on his PC. He could tell the Definity PBX through the PC over the LAN to place a conference call to those names. At the same time, a program running under Net[illegible] automatically send an e-mail to the people, alerting them to the conference call [illegible] them the agenda. All participants would have access to both the document and [illegible] ference call simultaneously. See Telephony Services.

NetWare MHS Netware MHS, which is software that provides store-and-forward [illegible] bility. Fax and E-mail systems that support MHS format their message transmissions [illegible] to MHS specifications. MHS reads compatible transmissions, determines the intended [illegible] ent and his location, and then sends the message to that location, regardless of [illegible] or E-mail system at the different ends. See MHS.

NetWare Telephony Services See Telephony Services.

Network Networks are common in our lives. Think about trains and phones. A [illegible] ties things together. Computer networks connect all types of computers and [illegible] things — terminals, printers, modems, door entry sensors, temperature monitors. [illegible] networks we're most familiar with are long distance ones, like phones and trains. [illegible] are also Local Area Networks (LANs) which exist within a limited geographic area, [illegible] few hundred feet of a small office, an entire building or even a "campus" such as [illegible] or industrial park. There are also Metropolitan Area Networks (MANs). See also LAN, [illegible]

Network Access Control Electronic circuitry that determines when a [illegible] may transmit next or when a particular workstation may transmit.

Network Access Line NAL. A communications channel between a [illegible] premises and the central office.

Network Access Point NAP. A telephone company AIN term. Software [illegible] a switch capable of recognizing a call that requires processing by AIN-logic which [illegible] ognizing such a call, routes the call to an SSP or ASE switch.

Network Accounting A system or application software module that [illegible] reports on packet-switched data network traffic, generally focusing on IP (Internet [illegible] traffic. Network accounting software captures data packets as they transverse the [illegible] presses them, and stores them on a centralized data repository to which the network [illegible] istrator, cost center managers, and privileged others can gain access to run various [illegible] Much like a call accounting system in the circuit-switched voice domain, a network [illegible] system captures data output from a switch or router. The SDR (Session Detail Record) [illegible] is much like CDR (Call Detail Record) output from a voice PBX. Much like PBX CDR [illegible] identifies the originating and terminating extension/telephone number, a network [illegible] system captures the originating and terminating IP address, and can translate [illegible] the MAC address of the LAN-attached user workstation, and the URL (Uniform [illegible] tor) of the Website the user has visited. Much like a call accounting system keeps [illegible] duration and time of day of a voice call, a network accounting system keeps [illegible] ration and time of day of a data network user's session. Network accounting systems [illegible] effective monitoring systems used by large corporations to ensure that expensive [illegible] sources are neither abused or misused. Such resources include both LAN resources [illegible] switches, servers, and routers) and high-speed (e.g., T-1 and T-3) circuits connecting [illegible] ternet. See also Call Accounting, Electronic Communication Privacy Act, and Keystroke [illegible] toring.

Network ACD Network ACD allows ACD agent groups, at different [illegible] (nodes), to service calls over the network independent of where the call first [illegible] network. NACD uses ISDN D-channel messaging to exchange information between [illegible]

Network Address Every cardEvery node on an Ethernet network has one or [illegible] addresses associated with it, including at least one fixed hardware address such as [illegible] 2c-1d-69-f1" assigned by the device's manufacturer. Most nodes also have [illegible] ific addresses assigned by a network manager.

Network Address Translation Network Address Translation. A [illegible] sion of Port Address Translation (PAT), NAT enables a local area network (LAN) to [illegible] set of IP addresses for internal traffic and a second set of addresses for external [illegible] allows a company to shield internal addresses from the public Internet. According to [illegible] NAT has several applications. You want to connect to the Internet, but not all your [illegible] have globally unique IP addresses. NAT enables private IP internetworks (i.e., [illegible] that use nonregistered IP addresses to connect to the Internet, or another public [illegible] network. NAT is configured on the router at the border of a stub domain (referred to as [illegible] inside network) and a public network such as the Internet (referred to as the outside [illegible] work). NAT translates the internal local addresses to globally unique IP addresses [illegible] sending packets to the outside network. You must change your internal addresses [illegible] of changing them, which can be a considerable amount of work, you can translate [illegible] by using NAT. You want to do basic load sharing of TCP traffic. You can map a single [illegible]

...to many local IP addresses by using the TCP load distribution feature. As a ...connectivity problem, NAT is practical only when relatively few hosts in a ...communicate outside of the domain at the same time. When this is the case, ...subset of the IP addresses in the domain must be translated into globally ...addresses when outside communication is necessary, and these addresses can be ...no longer in use. A significant advantage of NAT, according to Cisco, is that it ...figured without requiring changes to hosts or routers other than those few ...which NAT will be configured. NAT may not be practical if large numbers of hosts ...domain communicate outside of the domain. Furthermore, some applications ...encoded IP addresses in such a way that it is impractical for a NAT device to trans... applications may not work transparently or at all through a NAT device. NAT ...the identity of hosts, which may be an advantage or a disadvantage. A router ...with NAT will have at least one interface to the inside and one to the outside. ...environment, NAT is configured at the exit router between a stub domain and ... When a packet is leaving the domain, NAT translates the locally significant ...address into a globally unique address. When a packet is entering the domain, NAT ...the globally unique destination address into a local address. If more than one ...exists, each NAT must have the same translation table. If the software cannot ...address because it has run out of addresses, it drops the packet and sends an ...unreachable packet. A router configured with NAT must not advertise the local ...to the outside. However, routing information that NAT receives from the outside ...advertised in the stub domain as usual. See also Port Address Translation.

Network Addressable Unit NAU. In IBM's SNA, a logical unit (LU), physi... (PU) or system services control point (SSCP), which is host-based, that is the or... of information transmitted by the path control portion of an SNA network.

Network Agent A network agent is a device, such as a workstation or a router, ...to gather network performance information to send to the network man... agent. See Network Management Agent.

Network Analyzer A microwave test system that characterizes devices in terms ...complex small-signal scattering parameters (S-parameters). Measurements involve ...the ratio of magnitude and phase of input and output signals at the various ...network with the other ports terminated in the specified characteristic impedance ...50 ohms).

Network Application Architecture A generalized architecture allowing ...ability at the application level. Examples are Digital Equipment Corp.'s Network ...Support (NAS) and IBM Corp.'s Systems Application Architecture (SAA).

Network Application Support Digital Equipment Corporation's set of open ... which allegedly allows its customers to integrate, port and distribute applications ...different computer systems, including VMS, UNIX, MS-DOS, OS/2 and Apple ...

Network Architecture The philosophy and organizational concept for enabling ...communications between multiple locations and multiple organizational units. Network ...ture is a structured statement of the terminal devices, switching elements and the ...rules and procedures to be used for the establishment effective telecommunications. ...tecture.

Network Balancing 1. Lumped circuit elements (inductances, capacitances and ...ances) connected so as to simulate the impedance of a uniform cable or open-wire cir... a band of frequencies. 2. Moving circuits around in a multi-node switching net... so the switching loads on each of similar switching modules are roughly equal.

Network Basic Input/Output System NETBIOS. Within the context of ...DOS operating system, the software or software and firmware services that imple... the interface between applications and a network adaptor, such as a CSMA/CD or ...ring adaptor.

Network Board 1. A circuit board installed in each network station to allow stations ...communicate with each other and with the file server. 2. An SCSA term. A board device de... to act as an interface between a computer-based signal processing system and a tele... network.

Network Byte Order The Internet standard way of ordering of bytes corre... to numeric values.

Network Channel Terminating Equipment NCTE. A device or ... at the user's premises used to amplify, match impedance or match network sig... to the customer's equipment connected to the network. Basically, network channel ...ating equipment is a general name for equipment linking the network to a cus... tomer's premises. When NCTE connects to digital circuits, it typically consists of DSUs and CSUs. They are used for balancing of signals and providing for loop-back testing.

Network Computer NC. Larry Ellison of Oracle's idea of a $500 (or so) PC that lacks a hard disk and may lack a monitor but can be used to browse the Internet and run applications on a server on the Internet or corporate intranet. Ellison, who is Oracle's chairman, sees the NC as a "universal digital appliance." The New Yorker of September 8, 1997 discussed the implications of the network computer thus: Microsoft's worries about Ellison and NCs are not trivial. After a prolonged period of being in denial about the rise of the Internet, Gates and his team now understand that it is the central fact of the next phase of computing, and that it poses a real threat to Microsoft's power. In 1995, Sun Microsystems introduced an Internet-centric programming language called Java, which creates programs that can run on any operating system and is fast becoming the standard lingo of the Net. In a Java-fuelled future, the reign of the PC might be challenged by the NC which would let users "borrow" programs from the Net and would have no need for Microsoft's Windows — developments that would create enormous upheaval in many of the software markets that Gate's firm now dominates. See also Internet Terminal, NetPC and NetStation.

Network Computing System NCS. A RPC (Remote Procedure Call) system developed by Apollo, and used in DEC and Hewlett-Packard computer systems. The NCS protocol later was adopted by the Open Software Foundation (OSF). See also OSF.

Network Control Center A physical point within a network where various management and control functions are implemented.

Network Control Program NCP. An IBM Systems Network Architecture (SNA) term. This is the program that switches the virtual circuit connections into place, implements path control, and operates the Synchronous Data Link Control (SDLC) link. The Network Control Program is normally resident in the communications controller or the host processor.

Network Control Signaling The transmission of signals used in the telecommunications system which perform functions such as supervision, address signaling and audible tone signals to control the operation of switching machines in the telecommunication system.

Network Control Signaling Unit A telephone set that controls the transmission of signals into the telephone system which will perform supervision, number identification and control of the switching machines.

Network Controller A powerful microprocessor device designed to perform communications protocol translations between various terminals and computers and an X.25 packet switching network.

Network Data Management Protocol NDMP. An Internet draft specification from the IETF (Internet Engineering Task Force), NDMP is an open protocol for enterprise-wide, network-based data backup. NDMP is a secure backup technique which makes use of the TCP/IP protocol, running on networked file servers.

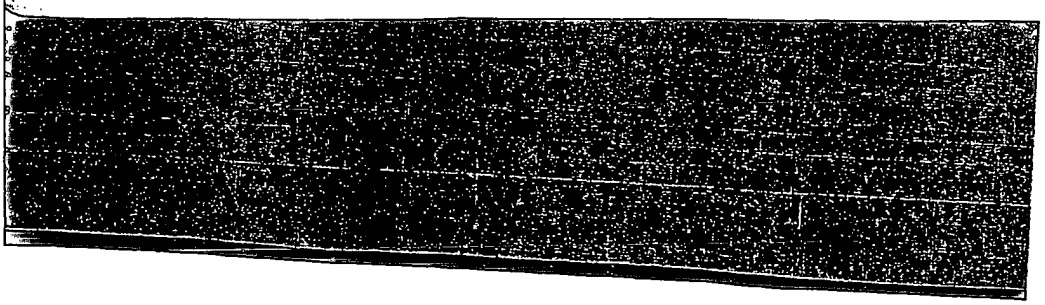
Network DDE Service A Windows NT definition. The Network DDE (dynamic data exchange) service manages shared DDE conversations. It is used by the Network DDE service.

Network Demarcation Point The network demarcation point is the point of interconnection between the local exchange carrier's facilities and the wiring and equipment at the end user's facilities. The demarcation point is located on the subscriber's side of the telephone company's protector.

Network Design and Optimization Network design and optimization is a process which balances network performance (availability) against cost. There are two fundamental tools in network design and optimization: a traffic usage recorder and software to interpret the results and make recommendations. A traffic usage recorder (TUR) is a device which connects to a network element in order to capture and record traffic statistics. Most network elements (e.g., PBXs, ACDs, data switches and routers) have special ports to which such a device can connect, usually via a RS-232 cable. As traffic flows through the network element, various information about that traffic is sent to the TUR in real time. The TUR holds that raw data in buffer memory until such time as it is polled by a centralized computer and the data is downloaded to that centralized computer. Later, the data is processed and reports are generated by traffic analysis software. That software will help you figure out which circuits you need, what speeds, to where, etc.

Network Design Order NDO. See Telephone Equipment Order.

Network Device Driver Software that coordinates communication between the network adapter card and the computer's hardware and other software, controlling the physical function of the local area network adapter cards.



Network Distribution The physical facilities of a communications carrier which are located outside the carrier's buildings. It includes cables, pole lines, conduit, etc.

Network Diversity A simply concept that says if you have a network which is important you need to have multiple ways of moving information around that network. Some of those ways should be provided by circuits from one vendor and some should be provided by circuits from several vendors. In other words, there should be diversity in both circuits and vendors. As an example of the necessity of lacking network diversity very seriously, consider the year 1991. In 1991, nearly 30 million residential and business customers, more than one hundred thousand airline passengers and an alphabet soup of state and federal agencies were temporarily crippled by nine major telephone network outages. Also in a 12-month period, the FAA (Federal Aviation Commission) reported a total of 114 phone service failures that disrupted air traffic facilities around the nation. Between June 10 and July 1, 1991, six major outages in three states comprised the largest series of network outages in history.

Network Drive A disk drive that is available to multiple users and computers on a network. Network drives often store data files for many people in a work group.

Network Economy The network economy is a term economists use when they apply the law of increasing returns to our entire economy. The idea is that one fax machine is worth nothing. Connect a hundred of them together and they're now worth something. Connect one million of them and suddenly you have something truly valuable. By the end of 1998, there were 350 million email addresses worldwide. Bingo, the network economy.

Network Element Control Protocol See NECP.

Network Elements NE. Processor controlled entities of the telecommunications network that primarily provide switching and transport network functions and contain network operations functions. Examples are: Non-ADN switching systems, digital cross-connect systems, ADN switching systems, and Signaling Transfer Points (STPs). In SONET, five basic network elements are: add/drop multiplexer; broadband digital cross-connect; wideband digital cross-connect; digital loop carrier; and switch interface.

Network Entity Identifier NEI. A cellular radio term. An Internet protocol address, or ConnectionLess Networking Protocol (CLNP) address, or any other protocol addressing used by the provider in transmission and receipt of Cellular Digital Packet Data (CDPD) services. The address that identifies that a user is authorized to use the service.

Network Equalizing A device connected to a transmission path in order to alter the characteristics of that path in a specified way. Often used to equalize the frequency response characteristic of a circuit for data transmission.

Network Extension Unit NEU. An AT&T thing which sits in a telephone satellite closet (their words) which links up to 11 Starlan daisy chained clusters of PCs, etc. in a star configuration through standard telephone wiring, modular cords and plugs.

Network Externalities First seen in an article by John Cassidy in The New Yorker Jan 12, 1998 issue. In plain English, "network externalities" means that the value of a product increases along with the number of other people using it. Few people care about how many people are buying corn flakes or pizzas. But it usually applies to high-tech goods for two reasons. They have to be compatible with one and other (how useless is a Betamax videocassette today?). Second, the buyers are often linked to a network. The more people on the network, the more valuable the product becomes. Clearly, one person and one phone has no value. Two people and two phones have more value. 10,000 people and 10,000 phones have even more value, etc.

Network Fax Server An network fax server is typically a PC on LAN. The PC has one or more fax/modem cards. Its job is to send and receive faxes for everyone on the LAN. As a result it is connected to dial out phone lines. See Server.

Network File System NFS. A method developed by Sun Microsystems Inc. for distributing files within a heterogeneous network. NFS has become a de facto standard. It requires special software drivers to suit specific vendor hardware and operating system software. NFS hardware and software gateway products are also available.

Network Horrors The reasoning behind the Bell System's insistence that the public could not hook its own equipment to the telephone line for fear it would produce "Irreparable Harm To The Network". A Bell System publication quoted the National Academy of Sciences as identifying four areas of potential harm. They were: Excessive signal power, hazardous voltage, improper network control signaling and line imbalance. Since the Carterfone Decision in the summer of 1968, the FCC has seen fit to allow devices, which pass an FCC Registration program, to be connected to the network, and while the quality of the stuff is occasionally poor, the fears of Network Horrors have proven groundless. The fears were, of course, raised to preserve the Bell System's erstwhile almost monopoly on the manufacture of telephone equipment in the U.S.

Network Information Center NIC. Centers providing user assistance, document service, training on the Internet.

Network Information Management System NIMS. Provides customers with browsing capability in their 800 and VNET databases.

Network Interface The point of interconnection between Telephone Company communications facilities and terminal equipment, protective apparatus or wiring at a subscriber's premises. The network interface or demarcation point shall be located on the subscriber's side of the Telephone Company's protector, or the equivalent thereof in cases where a protector is not employed, as provided under the local telephone company's reasonable and nondiscriminatory standard operating practices.

Network Interface Card Also called a NIC card. A printed circuit board consisting electronic circuitry for the purpose of connecting a workstation to a LAN. A NIC usually is in the form of a card that fits into one of the expansion slots inside a PC. Alternatively, it can fit into a slot of a MAU (Multistation Access Unit), which serves multiple LAN-attached devices, such as workstations and printers. In the context of IEEE standards, NICs operate at the MAC (Medium Access Control) layer. In the context of the OSI Reference Model, NICs operate at Layers 1 (Physical Layer) and 2 (Data Link Layer). The basic job of the NIC is to take data from the transmitting workstation, form it into the specific packet format demanded by the LAN protocol you are running (e.g., Ethernet or Token Ring), and present it to the shared medium (usually a cable). On the receiving end, the process is reversed, of course. Hard-coded into the NIC at the time of manufacture is a MAC address, unique in all the world to that NIC card; the MAC address effectively identifies the LAN-attached device with which it is associated. A NIC works with the network software and computer operating system to transmit and receive messages on the network.

Network Interface Controller Same as a Network Interface Card. See above definition.

Network Interface Device 1. NID. A device wired between a telephone protector and the inside wiring to isolate the customer's equipment from the network. 2. A device that performs, functions such as code and protocol conversion, and buffering required for communications to and from a network. 3. A device used primarily within a local area network to allow a number of independent devices, with varying protocols, to communicate with each other. This communication is accomplished by converting each device protocol into a common transmission protocol.

Network Interface Module Electronic circuitry connecting a system (usually a PC) to the telephone network. Network interface modules come in as many varieties as there are ways of connecting to the telephone network. They range from a simple, single start telephone line to complex ISDN PRI circuits. Usually the network interface module slides into one of the expansion slots inside a PC. The card transmits and receives messages from the resource modules and provides access to the telephone network.

Network Interface Unit 1. A Network Interface Card (NIC) or Multistation Access Unit (MAU) that attaches to a LAN. It implements the local network protocol and provides an interface for device attachment. See also NIC and MAU for much more. 2. NIU. A semi-intelligent device that serves as the point of physical and logical demarcation between the LEC (Local Exchange Carrier) and your residential or small business premise. The NIU includes a silicon-based protector that trips the circuit in the event of a lightening strike or some other form of aberrant voltage that otherwise would fry your telephone and burn down your house. The NIU also has enough intelligence to perform a master to conduct an automated loopback test, which tests the integrity of the twisted-pair, wire-based, twisted-pair, local loop from the central office (CO) to your premises - and is also called Loopback and NIU.

Network Internetworking A Frame Relay/ATM term. Network internetworking is a method of connecting two frame relay devices over an ATM backbone network. This approach leaves the entire frame relay frame intact, including header, payload, and the ATM devices act to set up the connection on a cut-through basis through the ATM tunneling protocol. Network internetworking is defined in the FRF.5 specification of the Frame Relay Forum, and is recognized by the ATM Forum. Contrast with Service Interworking.

Network Inward Dialing NID. A service feature of an automatic telephone network that allows a calling user to dial directly to an extension on a called user facility without operator intervention.

Network Layer Third layer of the OSI model of data communications.

Appendex III - Compatibility with HTML

HTML documents can be easily converted into the HTML+ format, and only a few changes are needed. Most documents won't need any changes at all. HTML+ browsers should be able to view HTML documents with very little effort. Older browsers will be able to view HTML+ documents which don't contain, tables or forms.

Lists

<menu>	becomes	<ul compact>
<dir>	becomes	<ul narrow>

Emphasis

HTML+ replaces the various tags used by HTML with a single tag. It may be worth changing the name for the emphasis tag in HTML+ from EM to EM, to gain compatibility with this common form. However, using EM might be confused with the typographical term *em* as in em dash (you also get en dash). EM has the merit of being unambigous. **I would like to get peoples views on this.**

	becomes	
<tt>	becomes	<em tt>
	becomes	<em b>
<b>	becomes	<em b>
<i>	becomes	<em i>
<u>	becomes	<em u>
<code>	becomes	
<samp>	becomes	
<kbd>	becomes	
<var>	becomes	
<dfn>	becomes	
<cite>	becomes	

Miscellaneous

Some tags which are deprecated in HTML are now obsolete, and should be mapped to preformatted text:

<plaintext>	becomes	<pre>
<xmp>	becomes	<pre>
<listing>	becomes	<pre>

The following two tags have been absorbed into the standard mechanism for paragraphs:

<address>	becomes	<p role="byline" align="right">
<blockquote>	becomes	<p role="quote">

References

This is missing the appropriate references to work on the syntax and name service for URNs. The HTTP definition needs updating to cover the encoding of form data (and *ismap* ?).

[Berners-Lee 93a] "*Hypertext Markup Language (HTML)*", Tim Berners-Lee, March 1993.
URL=ftp://info.cern.ch/pub/www/doc/http-spec.ps

[Berners-Lee 93b] "*Uniform Resource Locators*", Tim Berners-Lee, January 1992.
URL=ftp://info.cern.ch/pub/ietf/url4.ps

[Berners-Lee 93c] "*Protocol for the Retrieval and Manipulation of Textual and Hypermedia Information*", Tim Berners-Lee, 1993.
URL=ftp://info.cern.ch/pub/www/doc/html-spec.ps

[Raisch 93] "*Style sheets for HTML*", Robert Raisch, June 1993, O'Reilly & Associates
email: raisch.ora.com

[Kimber 93] Article in comp.text.sgml newsgroup, 24th May 1993 by Elliot Kimber (drmacro@vnet.almaden.ibm.com),
URL=news:19930524.152345.29@almaden.ibm.com

SoftLock Services Introduces

V

```
*************************************************************************
*      This message was requested from SoftLock Services' Email-Robot.  *
* * * * * * * * * *  We do not send "junk" mail. * * * * * * * * * * * * *
* If you receive any unsolicited or unwanted Email from SoftLock Services, *
*     please inform Jonathan Schull, Schull@SoftLock.Com, 215-993-9900   *
*************************************************************************
```

SoftLock Services introduces...SoftLock Services

Jonathan Schull, President

Schull@SoftLock.com

716-242-0348

SoftLock Services is a new kind of business for a new kind of product. While we offer an automated credit card processing service that will be a boon to Shareware authors (and many other people), the big news is our innovative suite of patent-pending Services and Tools for SoftLocking and selling freely-copyable computer programs and data.

In a nutshell, a SoftLocked computer program or document is a full-featured, freely copyable computer program or digital document, with a "lock" on certain "advanced features" (such as the ability to print, or to run at full speed). When a User tries to access one of those advanced features, the product displays a message like the following.

> In order to access the advanced features of this program in their full capacity, just...
>
> 1. Contact SoftLock Services any time, day or night, by telephone, modem, electronic mail, or fax.
>
> 2. Tell them you want a password for Product Number 87654321
>
> 3. Tell them the unique ID for that product on your computer is 12345678
>
> 4. Give them your credit card number (or SoftLock Voucher Number)
>
> And within 30 seconds, they'll give you a password that will unlock the advanced features, on this hard drive, forever!
>
> (And by the way, please give copies of this product to your friends. It will be of value to them, and if they want to unlock the advanced features they can just call us for the unique password they will need for THEIR hard drives).

MORE

All of this can be customized to suit the requirements of a particular product, but basically, that's all there is to it. In a typical SoftLocked application program, the developer might elect to SoftLock an advanced feature such as the ability to print professional-quality hard copy at full speed. In a SoftLocked electronic journal, some articles, and abstracts of all encrypted articles would be freely readable in any text browser or mail reader. Viewed through a SoftLock-aware text browser, an instant password could unlock one or more specific articles (and render this material readable, or printable, or modifiable, etc. at the author's discretion). Thus, SoftLocking can be used to strike the appropriate balance between "free sample" and "purchased product" all the while preserving the user's ability to backup, copy and pass on to others the entire SoftLocked application or document.

The business of SoftLock Services rests on three foundations:

1. SoftLock's (patent pending) technology: product- and machine- specific IDs can create a "lock" that requires a unique and unpredictable password available from SoftLock's central password "dispensaries".

2. A suite of commercial services, to make life easier for consumers and producers of SoftLocked products.

3. A philosophy: digital products should be virtually free until their value to the user is clear. Then, the creator of the product should be compensated fairly.

In keeping with this philosophy, our software tools are available to potential developers at cost, in order to create the market for our reasonably priced Services. By the same token, our client's products can be made available at nominal cost to Users who will (hopefully) find them useful when locked, and invaluable when unlocked. Users can "try before they buy" and then purchase passwords to instantly unlock any and all advanced features.

SoftLock's essential Service (and the one for which we take a modest commission) is the quick and convenient sale of Passwords and other products, round the clock, and around the world. Payment can be tendered on-line via credit card, or via SoftLock Vouchers through a variety of convenient channels. We provide information producers with sales, distribution, physical fulfillment and customer registration services and a variety of other support services, as well as prompt and accurate accounting and payment. We can provide these services to our Clients for far less than it would cost them to do it themselves.

SoftLock Services thus creates a whole new venue for the sale and distribution of freely copyable information. We do not pretend to know how our tools and services will best be exploited, but we are confident that we are in a position to help our clients and their customers find out.

With SoftLock Services...

Programmers and authors can now implement a business operation the same way one implements a computer operation -- by adding a few lines of code to their products. The resulting SoftLocked product can be cast upon the electronic waters, and (if Users find them valuable) checks for 80% of the retail cost will be deposited in the developers account.

Users can obtain software at little or no cost, "try before they buy", and unleash the full power of the software within minutes of deciding to make their purchase.

Software Producers can release complete programs and documentation, without copy protection, and still be assured that serious users will purchase the programs they use.

Hardware manufacturers can put computers and storage media into the hands of the people at cost (or below) by loading their goods with SoftLocked software and sharing in the profits from Password sales (by arrangement with SoftLock's clients). Thus, the computer becomes a "software vending machine", and digital technology becomes an inexpensive but invaluable commodity.

To make all of this possible we provide the following:

SoftLock Programmers' ToolKits

SoftLock Toolkits for C and Pascal programmers on PC and Macintosh computers, make it easy for programmers to create SoftLocked application programs that are unlocked by the Passwords we sell. Other ToolKits are under development, and SoftLock is eager to support those who would like to help produce them.

SoftLock Writers' Tools

Anyone who can use a word processor or spreadsheet can create a SoftLocked document. SoftLock Services has sponsored the development of freely-copyable SoftLock-aware text browsers and document-encrypting programs for DOS and Macintosh computers Using these and other tools we provide at nominal cost, SoftLocking a document can be done as easily as writing one.

Thus, the ability to SoftLock text files puts a virtual printing press, distribution channel, and sales operation into the hands of anyone with knowledge to sell.

SoftLock Fulfillment Services

We expect that many of our Clients' customers will want printed documentation, backup disks, etc. to follow up on their instant password purchases. SoftLock provides inexpensive on-demand fulfillment services for Clients who want to concentrate on what they do best -- create valuable digital products (not stuff envelopes).

SoftLock Customer's Tools

Privacy Enhanced Email and Digital Certification

There is now a standard for Privacy Enhanced electronic Mail which is so secure that it is subject to export restrictions by the US government. SoftLock Services is among the first mass-market information providers to empower users with the same kind of encryption technology used by banks and major corporations to protect their own business transactions. We will be making available to our customers two version of the RIPEM public-key encryption package: the original written by Mark Riordan, and a Macintosh version written by Ray Lau, author of the popular data compression program, Stuffit. SoftLock will honor and respond in kind to confidential messages, including credit-card based password purchases, through unsecured public Email channels. By endorsing and promoting this public standard, and by honoring Privacy-Enhanced electronic orders, SoftLock Services Inc., offers secure means for all of us to mind our own electronic business dealings. As usual (for us), all of this is free of charge. SoftLock Services will also provide "digital certificates" for attaching highly-secure "digital signatures" to PEM communications. Electronically "signed and sealed envelopes" are now a reality for everyone.

SoftLock communication tools

Our customers would be well served by a variety of communication tools to facilitate and automate their interactions with SoftLock Services. So we hereby invite developers to create some SoftLocked products for the purpose, which we will happily promote on their behalf, and of course sell under the auspices of our "standard deal".

Among the extensions we have planned are
--simple communications robots for the PC and Mac that will log in to the SoftLock Central bulletin board system, "squirt out" a pre-formatted order form, receive a password, and give the password back to the calling program (which will install it), and then hang up.
--a communications robot for bulletin board SYSOPs which will accomplish the same thing, and allow them to serve as on-line SoftLock resellers for their callers.
--an analogous internet-daemon to interact with SoftLock.com
--and so on.
Developers, get in touch!

SoftLock Vouchers

SoftLock Vouchers are analogous to "gift certificates" for SoftLocked Products. Each Voucher has a unique serial number and a specific monetary value, and can be used in lieu of a credit card number to purchase products from SoftLock Central. Each Voucher number is "retired" as soon as it is used. SoftLock Vouchers are useful in a variety of situations.

Software manufacturers can include a SoftLock Voucher in their shrink-wrapped packages, and thereby entitle their pre-paid customers to one (or more) pre-paid passwords.

Authorized SoftLock Resellers can sell SoftLock Vouchers to customers for whom credit card purchases are inconvenient or impractical. We offer quantity discounts on Vouchers to Authorized SoftLock Resellers in order to insure that customers anywhere in the world will be able to purchase passwords with local currency. Vouchers can be purchased and redeemed electronically, so we expect Authorized SoftLock Resellers to be able to help convey passwords to customers without convenient electronic access to SoftLock Central.

SoftLock Central

SoftLock Central is our virtual Customer Service headquarters, established to provide real-time processing of purchases from anywhere in the world at any time, through a variety of communication channels.

Touch-Tone Telephone. (1-800-SOFTLOCK)

Anyone with a touch-tone telephone can use our voice-response system to punch in a few numbers (ProductNumber, SoftLockID, Credit Card or SoftLock Voucher Number) and receive a password in approximately 60 seconds. Dongles and other consumer products can be ordered in a similar fashion.

Modem

Users with modems and communications software can log in to the SoftLock Central Bulletin Board System and purchase passwords on-line, order Dongles or ToolKits, etc. We are eager to encourage the development of automated communication tools that would log our system, submit SoftLock IDs and receive passwords

Email, Fax, and paper mail

SoftLock will respond within minutes to Email messages (including credit card and voucher-based purchases) addressed to SoftLock.com and to faxes sent to our Fax number. (We will also respond in kind to paper mail.) Many SoftLocked products will have the ability to generate order forms suitable for mailing or faxing to us. And as noted above, we are ready and able to respond to PEM-encrypted Email messages.

Voice

We don't want anyone left out. Customer Assistants at 215-993-9900 await your calls during business hours.

SoftLock Client Services

SoftLock's success depends upon the success of its clients -- the programmers and authors who produce SoftLocked products. We offer them a variety of services at minimal cost.

Developers can order ToolKits, from SoftLock Central, any time. They will soon be available at ftp sites and on Bulletin Board Systems worldwide.

SoftLock Certifications

SoftLock-aware text browsers must follow certain guidelines designed to protect the wishes of text-authors. SoftLock will only license SLX decryption technology to developers and products that are demonstrated to adhere to those guidelines.

SoftLock Resellers are enrolled in our certification program so that our customers can be confident they are buying valid SoftLock Vouchers from those able to provide appropriate customer support.

Certified Digital Signatures for Privacy Enhanced Mail are currently available at nominal cost to employees of many high-tech organizations. We will make this key to electronic security available to everyone else.

MORE.

SoftLock Central BBS Developer's Forum

This branch of the SoftLock Central BBS exists to register and serve SoftLock clients, and to provide them with the unique Product Numbers and Feature IDs they need to produce their own products. Developers without modem access to our bulletin board can receive the same services from our Developer's Assistants during business hours.

The BBS will allow Developers to:
- --set the prices of their products
- --see how their products are selling
- --access account information
- --collect user registration information
- --provide our fulfillment operation with documentation files, etc.
- --trade tips and referrals with each other and SoftLock Services.

The SoftLock Central BBS also hopes to be able to offer developers of large-volume products their own sub-forums on the BBS which they can use to offer support to their own customers.

We are eager to facilitate any other collaborative interactions on the internet or on the BBS that will put tools, inspiration and support into the hands of our clients, customers and collaborators.

In conclusion

We think we have created a new niche in cyberspace.

We do not pretend to know what will evolve in that niche.

We hope you'll join us.

[Submitted by: ANDREW WILLIAMSON (cijs26@vaxb.strathclyde.ac.uk)
Fri, 28 Jan 94 11:04 GMT]

A4

Intellectual Property Rights For Digital Library And Hypertext Publishing Systems: An Analysis of Xanadu

Pamela Samuelson

University of Pittsburgh School of Law

Robert J. Glushko

Hypertext Engineering
Pittsburgh, PA

ABSTRACT

Copyright law is being applied to works in digital form. The special character of digital media will inevitably require some adjustments in the copyright model if digital libraries and hypertext publishing environments are to become as commercially viable as the print industries have been. An intellectual property system works only when it embodies a reasonably accurate model of how people are likely to behave, but it is hard to predict author and reader behavior in an environment that has yet to be built. By far the most ambitious proposal for a digital library and hypertext publishing environment is Ted Nelson's Xanadu system. This paper reviews the intellectual property scheme in Xanadu and contrasts it with current copyright law. Xanadu's predictions about reader and author behavior are examined in light of how people currently behave in computer conferencing, electronic mail, and similar existing systems. These analyses identify some respects in which intellectual property systems might have to be changed to make digital libraries and hypertext publishing systems viable.

INTRODUCTION

An intellectual property system works only when it embodies a reasonably accurate model of how people are likely to behave. Copyright law is based on a relatively simple and straightforward model of author and reader behavior. Authors are assumed to be motivated to produce interesting and valuable texts, and to make these works available to others by copyright's reassurance that authors can control the sale of copies of their works. Readers are motivated to purchase the texts, or to urge institutions, such as libraries, to purchase the texts, so that they can have access to the work. Authors have generally had little control over what uses readers make of the copies after the first sale of the work to the public, and U.S. copyright law has sometimes regarded this lack of control over uses as a virtue. But while it can be said that the absence of use control promotes the dissemination of knowledge, the truth may be that in the print world it is infeasible to maintain meaningful control over uses anyway.

Copyright should be accounted a great success at modeling author and reader behavior, for the basic framework of this law has lasted nearly three hundred years. During this period, copyright industries have flourished and copyright law has broadened to include a wide variety of intellectual products besides those manufactured by printing presses.

Computers and the concomitant capability they have provided for making copyrighted texts available in digital form have created many new and exciting opportunities,

including the potential to create digital libraries and hypertext publishing systems. Active development of such systems is now underway ([Arms90]; [Enge90]; [Kahn88]; [Neuw90]). While there are many difficult technical problems that must be solved to build these systems, they are thought to be surmountable. Less clear, however, is what kind of intellectual property scheme is needed to make digital library or hypertext publishing systems commercially viable. While the copyright model is still being utilized for all manner of texts in digital form, the behavior of authors and readers is being changed by the new digital technologies. It is becoming increasingly likely that some adjustments will have to be made in the copyright model to make digital libraries and hypertext publishing environments as commercially viable as the print industries have been. But few new models have yet been constructed, and work in this direction has only just begun ([Kahn89]; [Zahr89]).

DIGITAL MEDIA AND INTELLECTUAL PROPERTY LAW

Elsewhere the first author has identified six characteristics of works in digital form that seem likely to change significantly the contours of copyright law [Samu90b]. The first and second of these, namely, the ease of replication of works in digital form, and the ease with which such works can be transmitted and accessed by multiple users, will create strong incentives for copyright industries to move away from their traditional focus on the sale of copies, and toward greater control over uses of protected works. That it is now feasible to control uses through controlling access to computer systems containing works in digital form will also affect this trend.

A third characteristic of digital works is the ease with which they can be manipulated and modified. While this plasticity offers users some important advantages over the print medium (printed works are sometimes *too* fixed to be maximally usable), copyright law is more experienced dealing with works that are permanently fixed. The law may need to be adjusted to cope with the new benefits and new problems that this plasticity will entail.

A fourth is that the traditional copyright distinctions among different kinds of works tend to break down when the works are in digital form. Federal copyright law recognizes seven categories of copyrighted works and provides each with different degrees of protection [USC88a]. Is a hypertext version of Mozart's "Magic Flute" that contains the music, the libretto, textual commentary, pictures of Mozart, and other media a "literary work", a "musical work", a "sound recording", a "pictorial work", or an "audiovisual work"? The answer to this question under copyright law cannot be all of the above--even if it is. Copyright's classification scheme, oriented as it is toward the appearance of works, seems in need of adjustment if the statutory differences are absent from the digital representation.

A fifth is that digital works are so compact as to be virtually invisible to user/readers. Consequently, user/readers are more dependent on user interfaces and navigation aids of a sort that the print world has not needed to provide. Intellectual property protection for interfaces and navigation aids are already a source of controversy, both on copyright and patent fronts, and seem likely to be more so in the future. Despite repeated Supreme Court rulings that algorithms are unpatentable [Samu90a] and evidence that practitioners believe strong protection by copyright and patent is bad for the software industry [Samu89], the U.S. Patent Office has been issuing many software patents in recent years. Many of these claim rights to certain functions and user interfaces for hypertext systems (see, for example, [Garb90]).

A sixth characteristic of digital media is the potential they provide for new search and linking activities, which may give rise to new classes of protected intellectual property products.

THE INTELLECTUAL PROPERTY SYSTEM IN XANADU

The most complete proposal for making digital library or hypertext publishing systems commercially viable has come from Ted Nelson, who coined the word "hypertext" and is often--and rightly--perceived as a hypertext visionary. For over two decades Nelson has been writing and talking about a proposed system called Xanadu, a vast digital library containing all of the world's literature [Nels87].[1] Because Xanadu will allow users to create new and derivative documents via links, Xanadu is also a hypertext publishing system. Xanadu can usefully be understood as an attempt to create an institution that will be writing environment, publishing environment, library, and bookstore in one.

Despite his visionary reputation, Nelson is practical enough to realize that the commercial success of the Xanadu proposal critically depends on the way it deals with intellectual property issues. The intellectual property system in Xanadu has sometimes been summarized in writings about the Xanadu system in popular magazines [Fraa87], but has been subject to little serious analysis.

After an introduction to the intellectual property system in Xanadu, this paper will discuss some respects in which the Xanadu proposal differs from the existing copyright system. While Xanadu contains some interesting ideas about how to solve certain problems with digital library and hypertext publishing systems, some aspects of the Xanadu model of author and user behavior may be unworkable. This analysis suggests some respects in which intellectual property systems might have to be changed to make digital libraries and hypertext publishing systems viable.

The Xanadu system builds on the foundation of copyright law, but goes beyond it to include some features that differ significantly from the standard copyright model. Nelson proposes to contract with all authors whose works are stored in the Xanadu system about derivative uses that can be made of documents in the system. Varying the "default setting" of copyright by contract is not, in itself, a novel thing. The motion picture industry is an example of a copyright industry that has historically depended for commercial success on contract-based distributions of copies, rather than on the outright sale of copies which has typified most copyright industries. Nelson's scheme is novel in proposing to use a contract-based scheme for commercial distribution of written texts, the prototypical subject matter of copyright.

Revenue and Royalty Incentives and Mechanisms

Revenues are generated in Xanadu from two sources: one, as author fees for renting space for their documents in the Xanadu system, and two, as user fees for their usage of the system. A portion of the usage fee (estimated at 10-20%) is to go to authors whose documents are accessed by users; the rest will go to the system to recoup costs and make profits. (If public domain documents, such as Shakespeare's plays, are accessed, the author portion of the fee will go into an "author's fund" for scholarships and the like.)

Nelson expects that authors will want to put their documents into the Xanadu system because once the documents are in the system, authors will be able to earn royalties whenever users make use of their documents. For the sake of administrative convenience,

---

[1] Nelson has described Xanadu in numerous publications, presentations, and interviews. Many of the publications have appeared in multiple editions, so it is hard to identify any one work as the definitive specification for Xanadu. Furthermore, Xanadu is being commercialized by Autodesk, a highly profitable firm with a track record of successful products. A commercial version is likely to differ from Nelson's vision, but it is instructive to consider Nelson's proposal in its "pure" form to understand some of the changes and compromises Autodesk is likely to make.

Nelson intends for the usage fees, and consequently the royalties as well, to be set on a per byte delivery basis. Nelson expects that people will pay to use the Xanadu system, because not only will it contain as much of the world's literature as Nelson can get into it, but there will also be legion opportunities in the Xanadu system for users/browsers/readers to make money by adding value to the system through their creative uses of the system.

There are two main ways Nelson intends to let users make money in the Xanadu system. One is by making derivative works of documents already in the system, such as new versions of other authors' documents, compound documents consisting of portions of a number of different documents, or commentaries on other documents in the system. By creating derivative documents, users would become system authors themselves, and thereby become able to earn royalties when other users access their derivative documents. No special permission would be needed to make derivative documents from other authors' documents, for Nelson will make it a condition of storing documents in Xanadu that authors agree to allow others to make whatever derivative uses they want of published documents in the system.

Nelson relies on two factors to motivate authors to agree to allowing derivatives to be made of their documents. One is that they will then be able to do to others' documents what others can do to theirs. But more importantly, when a third party accesses the derivative document on Xanadu, the author of the underlying document, as well as the author of the derivative document, will earn a royalty because the derivative document will be connected to the original document; bytes from both will be called up when third parties access the derivative document. Hence, both authors will receive royalties.

A second way for users to generate revenues when using the Xanadu system will be by creating links between (or among) documents in the system. Nelson expects some links to be very elaborate, such as a specialized index to certain classes of documents in the system; others may be modest, such as a connector between two documents. User links between documents, in effect, become new documents in the system. Each time other users traverse a set of links, the link author will receive a royalty, as will the authors of the documents on either end of the link. Although Vannevar Bush was the first to perceive that information trailblazers would be needed for computerized information systems [Bush45], Nelson deserves credit for recognizing the need to give incentives to information pioneers to cut paths through the invisible contents of a digital library.

Nelson's scheme would also provide authors with the opportunity to store private as well as published documents in the Xanadu system. Authors will be able to define who can have access to the private documents and under what conditions. Private documents can be withdrawn without difficulty from Xanadu by their authors. The same will not be true for published documents because of the effect withdrawal would have on the interests of authors who have linked to or otherwise built upon the foundation of the published document. Nelson attempts to create a strong incentive for authors to publish their documents in the Xanadu system by making system royalties unavailable to authors for private documents, even those with unrestricted distribution (i.e., from which derivatives can be made, and to which links can be constructed). Because publication imposes obligations on the Xanadu operator and the author, publication of a document in the Xanadu system is a formal event, requiring a signature of the author on a form affirming the intent to publish the work.

HOW THE XANADU INTELLECTUAL PROPERTY SYSTEM DIFFERS FROM THE COPYRIGHT SYSTEM

Nelson refers to copyright in a positive way in a number of passages in his book, and takes great care to establish a plausible case that nothing in Xanadu violates existing copyright law. Xanadu gives authors new ways to generate revenues from their works--even some that copyright might not provide--and so aims to create incentives to authorship, revealing a predisposition in keeping with traditional copyright incentives.

But the Xanadu system is more different from copyright than might be apparent from a cursory examination.

Accounting by Uses, not by Copies

One difference between the Xanadu intellectual property system and traditional copyright is that Xanadu aims to derive revenues for authors by charging for each and every *use* of their documents, rather than, as has traditionally been done in copyright industries, on the sale or other commercial distribution of *copies* of copyrighted works. Numerous other commercial computer data bases do much the same thing. Such arrangements seem likely to become increasingly common for works in digital form.

Blurring the "Idea" and "Expression" Distinction and Eliminating the "Fair Use" Provision

More novel are the set of differences from copyright that flow from Xanadu's treatment of links. Fundamental to the copyright regime is a distinction between "ideas" (which are unprotected by copyright) and "expression" (which is what copyright protects). Under the copyright regime, authors generally do not expect remuneration whenever other authors comment on, quote from, use ideas from, or make reference to their work. The "fair use" provision allows even literal copying of copyrighted text if the amount is small and for research, educational, or critical purposes. Only if other authors take a fairly hefty chunk of "expression" from the protected work do copyright holders expect compensation. Even for printed works, however, there is no exact boundary between "small" and "hefty" copying under fair use provisions, and some authors and publishers avoid the issue by obtaining rights to use even a handful of words.

In Xanadu, because information can be included in a document by linking, the definition of what information to count as a single work becomes unclear. This would complicate the determination of what constitutes fair use in any case. Nevertheless, Xanadu allows no fair use copying, and authors in the Xanadu system are expected to get varying royalties based on how many bytes were linked to, merely for being linked to. While Nelson presents arguments for this scheme, an intellectual property system that compensates authors without regard to whether chunks of expression have been appropriated may tend to undermine the "idea/expression" distinction that has been a staple part of the copyright system.

Treating Linking as Authorship

Nelson's decision to treat linking as a kind of authorship--an intellectual activity that should be encouraged, that should serve as the basis for earning royalties when users traverse the links, but that should not be controllable by authors of the documents being linked to--diverges somewhat from the traditional copyright model [Samu90b]. While an extensive set of links, such as an index, might readily be protectable by traditional copyright law as a compilation, many of the kinds of links that Nelson would treat as works of authorship might be unprotectable under traditional copyright law. A link between a passage in document A and a passage in document B might, for example, be considered a "discovery" that the statute says copyright cannot protect [USC88b]. Additionally, traditional copyright law would not regard it as a compensable use of a copyrighted work for readers to traverse the links among documents referred to in a printed article [Samu90b]. Yet link authors in Nelson's scheme would be compensated for link traversal.

Defining "Rights to Do" Rather than "Rights to Exclude"

It seems natural for people to think of intellectual property rights in terms of what authors should be able to get compensation for, what users should be able to do with documents in the system, and the like. This intuitive "rights to do" framework is used by

Xanadu. The law tends to define intellectual property rights in a somewhat different way. The law focuses on what rights owners have to *exclude* other people from doing certain kinds of things with the protected work. (The law, in general, tends to identify certain conduct as prohibited, leaving all else as legal conduct.)

Copyright law defines the ownership rights of authors by saying what kinds of activities they can stop unauthorized people from doing, which chiefly are: making copies of the work, making derivative works, and selling unauthorized copies or derivative works. The only exclusive right Xanadu seems to contemplate is whether or not to put a document into the Xanadu system in the first place. Xanadu is more like a compulsory license system than an exclusive rights system. While U.S. copyright law does contain some compulsory license provisions, compulsory licenses are generally an anathema to owners of intellectual property rights because the license fee generally bears little or no relation to what the market would bear if the issue were left to the market.

Extending the Duration of Rights Indefinitely

The Xanadu system seems to contemplate no end to the duration of author rights. As long as authors (or their heirs) continue to pay for storage on his system, Xanadu will to continue to pay royalties for uses of the documents. Copyright must, under the U.S. Constitution, only grant authors exclusive rights for limited times. Upon expiration of the copyright, the work is in the public domain. While Nelson may intend to include this aspect of copyright in the Xanadu system, he makes no mention of it. Certainly, he does not intend to reduce the usage fee for accessing public domain materials; royalties from them go into the "author's fund" over which he undoubtedly will exercise some control.

Making Publication a Formal Event (again)

Publication is an important formal event in the Xanadu system. Under "old" copyright law, an author only "copyrighted" his or her work when the work was published. Since 1976, federal copyright law has protected works of authorship from the moment of their first fixation in a tangible medium. Between 1976 and 1989, publication was mainly important because authors had to attach a copyright notice to published copies of the work. In 1989, this notice requirement was dropped, which made publication into a nonevent in copyright law. By making publication into a significant event, Nelson's scheme resembles "old" copyright more than "new" copyright.

Making publication a formal event in Xanadu is necessary because it creates a contract between the Xanadu operator and the author to guarantee the existence of the published document for a period of time. This provides an integrity to links and citations generally absent in the print world, where only law reviews, with their armies of student citation-checkers, assure the reader that the cited document exists and supports the proposition for which it was cited.

In short, the Xanadu intellectual property system is more different from copyright than one might think from reading Nelson's books. Nelson's insights about linking--the need to create incentives to do it, a willingness to treat linking as authorship and to treat the traversing of links by users as deserving of compensation to link authors, and the inability of authors to control who can link to their documents--are his most important and original contributions to current thought about how intellectual property issues should be handled for digital library and hypertext publishing systems. But some aspects of the Xanadu intellectual property system depend on assumptions about how authors and readers will behave that may be incorrect.

MODELS OF AUTHOR AND READER BEHAVIOR IN EXISTING COMPUTER INFORMATION AND MAIL SYSTEMS

How can one question a model of a system that has yet to be built? Some clues about how authors and readers might behave in digital libraries and hypertext publishing systems come from how people use computer bulletin boards, information services, and electronic mail systems. Instead of viewing these systems as technical precedents, it is instructive to consider them as experiments to develop appropriate models for intellectual property and human behavior to be applied to more ambitious applications like Xanadu.

Prodigy and CompuServe

Prodigy and CompuServe are commercial services that provide a variety of information services, bulletin boards, electronic mail, and entertainment. They embody significantly different intellectual property models and the behavior of their users is markedly different. Prodigy is targeted to the consumer and home market, and treats its users as relatively passive information consumers who do not interact much with each other. Prodigy is marketed in part for its entertainment value, and Prodigy's services are made available for a fixed monthly fee; usage-insensitive pricing is made possible by the paid advertising that Prodigy presents along with nearly every screen of information displayed by users. When Prodigy imposed a usage-pricing scheme for sending electronic mail, many users felt that their contract with Prodigy had been violated.

In contrast, CompuServe is oriented toward business and professional users and has always had usage-based pricing based on connect time. CompuServe information services are specifically focused, organized into a complex hierarchy of bulletin boards and databases, many of which are moderated by an expert, who in some circumstances is compensated by CompuServe. This finer-grained categorization enables CompuServe to impose surcharges for supposedly more timely or valuable information, but its user population is presumably used to paying for information according to its value. Users engage in heated electronic dialogues with each other on bulletin boards, commenting on and criticizing each other's postings.

The Internet

The Internet is a vast network of networks that interconnect thousands of computing sites in government, industry, and academia. The Internet has evolved from primarily providing electronic mail services to become the infrastructure for significantly broader services of information exchange and collaborative work. Like CompuServe, the heart of the Internet is a vast collection of newsgroups in which participants from around the world post and comment on messages. Some people take on the role of newsgroup moderators, but the overwhelming majority of newsgroups are unmoderated.

Author and reader behavior on the Internet are governed by norms or "netiquette" that have evolved over time and that are enforced both by system administrators and by the informal but effective sanctions of "flames" (critical messages) directed at violators. Included in these norms are rules about selecting newsgroups in which to post messages, choosing titles, sensitivity to authors of cited messages, and other topics that improve the lot of both authors and readers.

Users of the Internet vary greatly in their perception of intellectual property laws as they apply to this new kind of publishing system. Some users (especially new users who are college students) act as if the Internet services and the information it contains are completely free, and copyrighted material from newspapers or books often is posted

without permission.[2] At the same time, other authors explicitly assert copyrights on the messages that they post.

Authors are not paid to publish and receive no royalties, and readers do not have to pay to read, but it is fair to state that these activities are often being paid for (or at least subsidized) by their employers. Hence, it can be argued that anything posted on the Internet that is work-related is the intellectual property of the employer who provides access to the Internet by paying for the computers and telecommunications infrastructure. Employers may feel that the value of the information their employees glean from the Internet outweighs the costs of the time to obtain it, but it is unlikely that few employers explicitly make this analysis.

THE XANADU MODEL OF AUTHOR BEHAVIOR

One fundamental question raised by the Xanadu system is whether authors, particularly good ones, will be willing to pay to publish their works in Xanadu. Some authors may publish documents in Xanadu out of misplaced confidence in the value of their work, just as authors now post messages of dubious information content to CompuServe or Internet newsgroups. They, of course, will get feedback at the end of the first rental period when no royalties are credited to their account. Some authors also seem likely to decide not to renew their document rental space in Xanadu if no one linked to them during the first rental period, even though if they had stayed in the system, their documents would have eventually have been discovered and made them a fortune. Still other authors may lack confidence in their work or may be too poor to afford the rental fee, which may cause them to withhold from the Xanadu system documents that would have been widely utilized if published there. Xanadu might benefit from a scheme by which authors can solicit sponsors willing to subsidize the inclusion of their works in Xanadu in exchange for some portion of the royalties.

Authors may not, in other words, behave in the way Xanadu's designers might expect them to behave. Authors may prefer the print world's system which does not require authors to pay directly for the privilege of being published. Authors may feel it is quite enough to have had to work hard to write the text in the first place. Some of the trick of authoring is writing something that publishers are willing to risk their capital to publish. A system that would make authors pay to get published may end up either deterring authorship or sending authors in search of another digital library/hypertext publishing system in which to place their work.

The Xanadu model may also have underestimated how reluctant many authors may be about giving other people unlimited rights to make derivatives of their work. Although authors seem likely to have no objection to letting Xanadu users link to their documents, they are likely to feel quite differently about allowing any Tom, Dick, or Susan make their own versions of the authors' works, or to combine portions of their documents with portions of others' documents. It will be little consolation to such authors that they too might get royalties when the revised version or compound document was accessed by Xanadu users. Authors often regard their writings as expressions of their personalities. They tend to regard any tampering with their text as a "mutilation" of the work, as objectionable as if someone had the effrontery to walk up to you and cut your hair without your permission. In many countries, authors are expressly granted "moral rights" in their intellectual products, one of which protects the integrity of the work. In the U.S., the derivative work right of copyright owners protects authors' economic interests in controlling adaptations of their works. Nelson, like many members of the computer community, may have a much more positive attitude about taking someone else's work

---

[2] These same college students would likely be more sensitive to issues of plagiarism and infringement applicable to printed works when they write term papers.

and building on it to create a better modified version. Nelson seems to have assumed this attitude is more widespread in the authorial community than may, in fact, be true.

THE XANADU MODEL OF USER BEHAVIOR

The Xanadu intellectual property system is also based on a model of user behavior. Nelson has proposed for Xanadu a set of incentives for people to make use of the system for a wide variety of purposes, from research to entertainment to hobby to full-time occupation. Probably his most creative idea is that by which he contemplates transforming the digital library part of Xanadu into a hypertext publishing system, incenting users to become system authors through linking and other derivative uses of documents in the system.

But the royalty mechanism in Xanadu may create some unfortunate, unintended incentives. The system would seem to give an especial premium to those who are first to mention a particular topic in the Xanadu system, even if the first treatment of the topic was shallow or wrong. This may create incentives to rush documents into the system rather than to craft them to be deeper and more accurate.[3] An example will illustrate one such problem.

Suppose a journalist attended the first conference of scientists concerning the just-formed Human Genome Initiative, that he was an avid Xanadu user, and that at the first break in the conference schedule, the journalist authored a document for Xanadu describing in a shallow but intelligible way what HGI was about. By virtue of being the first to mention HGI in Xanadu, this journalist's entry might be, for a time at least, the most frequently linked to source on HGI in Xanadu, which would make him the most compensated author on the topic.

A naive user of Xanadu, when faced with a decision to access the journalist's HGI description or a later much deeper one by a scientist who was a founder of the HGI, might see that the first had been linked to a thousand times, whereas the scientist's document had been linked to only five times in the time it was on the system. This might cause the user to choose the more frequently cited source over the better but less frequently cited source, again causing more royalties to flow into the journalist's account, and incenting rushed documents over considered documents in the system. In the print world, the shallow first treatment on a topic will tend to be ignored by later authors, but in Xanadu, the first document to mention a subject might always be called up on a user search, and not until the user reads the shallow document (and hence pays the author royalties on it) will the user know to ignore it. Even creating a derivative document advising users to ignore the underlying document will result in royalties to the author of the underlying document.

Suppose that the journalist's Xanadu document on HGI contained some errors. Other Xanadu users might well notice the errors, and make derivative documents containing the needed corrections. Although this would correct the error, an inadvertent result of the scenario would be that the journalist might make a lot of money from putting out an erroneous document, for every time someone people linked to his document or created a revised version of it, the journalist would share in the revenues. The more, and more noticeable, were the errors in the document, the larger the number of Xanadu users likely to notice the errors, to link to his document, and/or revise it, which once again would

---

[3] This phenomenon is well-known in conventional publication media, of course. A visit to a bookstore or grocery store uncovers scores of slipshod books that report on the latest fad, war, movie, or entertainment personality. But Xanadu seems likely to increase the odds that first-in authors are rewarded because it doesn't allow readers to scan the work while waiting in line at the cash register to discover how shallow it really is.

generate more revenues for the journalist. This would seem to over-reward the journalist for rushing to get his document on HGI into the Xanadu system and not deter entry of erroneous information.

Usage-based systems, such as Xanadu, may also have the disadvantage, at least for price-sensitive users, of making those with the most curiosity and tenacity in research to pay the highest cost. They are the ones who will presumably use Xanadu for longer periods of time. Now, one might argue that this is fair because those who use the system the most are those who pay most. But some may conceive the issue differently, and think it one of the great virtues of the library systems of the print world that scholars do not have to pay more than casual users for access to the library. We want to encourage deep scholarship; by not making scholars pay more for their use of the library, the print world encourages scholarship. Digital library and hypertext publishing systems may also need to find ways to encourage good scholarship and curiosity without making it prohibitively expensive.

But a more serious problem perhaps than this may be figuring out how to motivate users to be persistent and creative in their use of the Xanadu system. It is difficult enough for ordinary folk to use libraries with print materials in it which they can walk around and browse through until they find something to interest them. In Xanadu, the clock will be ticking and the price will be rising as one browses. Digital libraries, because of their invisibility to the user, may be, for ordinary folk, too abstract to be enjoyably browsable. Once again, Nelson may have mistakenly modeled the Xanadu user in terms of his own persistence and creativity which others may not share.

QUESTIONS ABOUT PRICING INCENTIVES

But perhaps the single most questionable element of the Xanadu intellectual property scheme, from the standpoint of economic incentives, may be limitations of its pricing scheme. If one looks at the universe of copyrighted works in the print-dominated world, one will immediately observe that copies are priced according, more or less, to what the publisher/distributor and author/creator think the market will bear for the number of copies of it that it is reasonable to think can be sold or licensed. Xanadu posits a flat fee for Xanadu connect time and a fixed royalty for authors based on per byte delivery for certain kinds of usage of the document. This is like mandating that all books must be priced according to the number of pages they contain and all pages must be priced at the same amount. The CompuServe example seems to suggest that differential pricing of information is necessary to encourage the development of specialized markets. Unless Xanadu were the world's only digital library and hypertext publishing system, which remains Nelson's vision but which is unlikely, Xanadu will lack the negotiating power to compel authors to accept fixed pricing per byte of their information.[4]

People who own copyrights in very valuable intellectual properties simply won't use a system that won't let them make market-based pricing decisions. The only options authors of very valuable intellectual properties would have in the world Nelson envisions is to put the work in Xanadu as an encrypted private document and contract with users for access to the document, or to withhold the document from Xanadu altogether. While encryption might allow market pricing to occur, Xanadu does not facilitate these transactions; they are to be dealt with between the parties, but if Xanadu does not facilitate the transactions, it is difficult to see how they can occur. The transaction costs of individual negotiations which must occur outside Xanadu in order to access the

[4] If Xanadu were the only means for authors to publish their works, it would enjoy what economists call a monopsony, a situation with only one buyer for many sellers, which generally leads to exploitation.

encrypted document in Xanadu would seem inordinately high.[5] Nelson contemplates that authors or publishers of some valuable copyrighted works will choose not to put their documents in Xanadu. Nelson has an answer to this problem that may end up getting him in trouble if it works. Nelson says that it will still be possible for Xanadu users to link to and create derivative documents of works not stored in the Xanadu system. It would not be surprising if a copyright lawsuit was brought to stop such derivative activity.

CONCLUSION

Whether digital library or hypertext publishing systems can be made commercially viable will depend on how they deal with intellectual property rights issues. The traditional copyright model will require adjustments in order to facilitate these new kinds of institutions. Ted Nelson offers one model of how such adjustments might be made. While Nelson's intellectual property scheme for the Xanadu system is bold and innovative, there are a number of respects in which his system can be questioned. Most uncertain are the accuracy of the Xanadu model of author and user behavior, and the adequacy of financial incentives for authors to put their most valuable copyrighted works in the Xanadu system.

A generation of exposure to tape recorders and VCRs, and a raft of new digital technologies for scanning, frame grabbing, and sampling are making it harder to predict how people understand and relate to intellectual property. What is legal, and what is merely technically possible to copy? What constitutes "fair use" of digitally-encoded copyrighted works? Laws that were suited for traditional kinds of copyrighted works no longer seem to fit.

More work is needed to develop new models of author and user behavior and the economics that will yield the right level of incentives for creation of digital library and hypertext publishing systems. The law can be made to conform to these new models, but only after we figure out what the right ones are.

ACKNOWLEDGMENTS

We thank Mark Bernstein, Joe Farrell, Anna Belle Lieserson, James Moore, and several anonymous reviewers for their helpful criticism.

REFERENCES

[Arms90] Arms, C. (Ed.). *Campus strategies for libraries and electronic information.* Bedford, MA: Digital Press.

[Benn91] Benn, N. Copyright Collectives and Reproduction Rights in Electronic Media. *New Media News*, 5(1), 21-23, Winter 1991.

[Bush45] Bush, V. As we may think. *Atlantic Monthly*, July 1945. (Reprinted in Lambert & Ropiequet (Eds.), 1986, *CD-ROM - The New Papyrus*, 101-108. Microsoft Press.)

---

[5] Similar complaints about transaction costs for licensing of rights for digital media are motivating the development of new copyright collectives for electronic works modeled after ASCAP and BMI for the music industry [Benn91].

[Enge90] Engelbart, D. Knowledge-domain interoperability and an open hyperdocument system. *CSCW'90: Proceedings of the Conference on Computer-Supported Cooperative Work.* New York: ACM, 1990, 143-156.

[Fraa87] Fraase, M. Hyper-Mania, *The MACazine*, 64 (Nov. 1987).

[Garb90] Garber, S., Kozak, D., Kruse, J., & Clare, M. *Intelligent optical navigator dynamic information presentation and navigation system.* U.S. Patent 4,905,163, issued February 27, 1990.

[Kahn88] Kahn, R., & Cerf, R. *The Digital Library Project.* Corporation for National Research Initiatives, 1988.

[Kahn89] Kahn, R., & Cerf, V. Knowbots in the real world. *Workshop on the protection of intellectual property rights in the digital library system.* Corporation for National Research Initiatives, 1989.

[Nels87] Nelson, T. *Literary Machines* (Ed. 87.1).

[Neuw90] Neuwirth, C., Kaufer, D., Chandhok, R., & Morris, J. Issues in the design of computer support for co-authoring and commenting. *CSCW'90: Proceedings of the Conference on Computer-Supported Cooperative Work.* New York: ACM, 1990, 183-195.

[Samu89] Samuelson, P., & Glushko, R. Comparing the views of lawyers and user interface designers on the software copyright "look and feel" lawsuits. *Jurimetrics*, 30(1), Fall 1989.

[Samu90a] Samuelson, P. *Benson* Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions. *Emory Law Journal*, 39(4), Fall 1990.

[Samu90b] Samuelson, P. Digital Media and the Changing Face of Intellectual Property Law, *Rutgers Computer & Technology Journal*, 16, 323, 1990.

[USC88a] 17 U.S.C. sec 102(a), 1988.

[USC88b] 17 U.S.C. sec 102(b), 1988.

[Zahr89] Zahry, P., and Sirbu, M. The Provision of Scholarly Journals by Libraries via Electronic Technologies: An Economic Analysis. Engineering and Public Policy Department Technical Report, Carnegie-Mellon University, 1/16/89.

Request for Comments: XXXX

David Raggett (Hewlett Packard)
12th July 1993

HTML+ (Hypertext markup language)

A proposed standard for a light weight presentation independent delivery format for browsing and querying information across the Internet

Status of this Memo

This document is a proposal for an Internet Draft, and specifies the HTML+ wide-area hypertext document format, with a view to requesting discussion[1] and suggestions for improvements. Distribution of this memo is unlimited.

Abstract

HTML+ is a simple SGML based format for wide-area hypertext documents, for use within the World Wide Web. Unlike desktop publishing formats, HTML+ captures the logical intent of authors. This simplifies the task of writing documents, and permits them to be effectively rendered on a wide range of display types as well as the printed page.

HTML+ represents a substantial improvement over the existing format: HTML, offering nested lists, figures, embedded data in foreign formats for equations etc, tables with support for titles and column headings, change bars, entry forms for querying and updating information sources and for use as questionaires for mailing. This document specifies the HTML+ format with guidelines on how it should be rendered by browsers.

Introduction

The World Wide Web is a wide area client-server architecture for retrieving hypermedia documents across the Internet. It also supports a means for searching remote information sources, for example bibliographies, phone directories and instruction manuals. There are three main ingredients:

a) Universal naming scheme for documents. The universal resource location syntax specifies documents in terms of the protocol to be used to retrieve them, their Internet host and path name. A format for location independent lifetime identifiers is currently being defined by working groups of the IETF. A network protocol will allow universal resource numbers (URNs) to be resolved to the URL for the nearest available copy.

b) Use of available protocols for retrieving documents over the network, including FTP, NNTP, WAIS, Gopher, and HTTP. The latter is designed specifically for use with the World Wide Web, and combines efficiency with an ability to flexibly exchange information between clients and servers.

c) A document format supporting hypertext links based on URLs and URNs which can specify documents anywhere in the Internet. HTML+ is designed for rendering on a wide variety of different display types and platforms.

Information browsers can display information in a wide variety of formats, e.g. plain text, rich text in the HTML+ format, images in the GIF and JPEG formats, MPEG movies, and MIME documents. The hypertext format has a special significance as it allows users to navigate from one document to the next at the click of a button. It provides the basis for menus, cross references, either within a document or to other documents,

---

[1]Please mail comments to the author dsr@hplb.hpl.hp.com, or to the WWW discussion group: www-talk@nxoc01.cern.ch

perhaps on the other side of the world. It also provides a means of building larger scale collections of documents that act as journals, books or encyclopedias. The format is also intended to act as a building block for creating wide area groupware applications.

HTML+ follows on from an earlier standard - HTML, see [Berners-Lee 93a], which has been widely used as the basis for hypertext documents in the World Wide Web. The new format grew out of experience with HTML, culminating in the desire to add new features, e.g. inline images, tables, and form fields for greater flexibility in querying remote information sources. This document specifies the HTML+ format and suggests ways in which browsers can choose to render it on a variety of different display types.

2. HTML+ and SGML

HTML+ itself is based on the Standardised General Markup Language (SGML), an international standard for document markup that is becoming increasingly important. The term markup derives from the way proof-readers have traditionally pencilled in marks that indicate how the document should be revised.

SGML grew out of a decade of work addressing the need for capturing the logical elements of documents as opposed to the processing functions to be performed on those elements. SGML is essentially an extensible document description language, based on a notation for embedding tags into the body of a document's text. It is defined by the international standard ISO 8879. The markup structure permitted for each class of documents is defined by an SGML Data Type Definition, usually abbreviated to DTD.

Working groups in ISO have recently produced a range of SGML DTDs for documents, e.g. ISO 12083 defines DTDs for books and ISO 10744, which defines the HyTime standard for hypermedia/time-based documents. These standards are large and complex, and perhaps best suited as interchange standards that facilitate conversion between proprietary document formats. By contrast, HTML+ provides a lightweight delivery format that can be rendered by relatively simple browsers, and which has grown out of two years practical experience with wide-area hypertext information systems in the Internet community.

HTML+ and HyTime

The HyTime standard provides a rich range of architectural forms, but is not aimed at run-time efficiency. Suggestions have been made as to how the HTML DTD could be adapted to comply with HyTime's clink architectural form [Kimber 93]. This would necessitate documents declaring links as external entities and the use of local names in link definitions, but in the absence of any immediate benefit, there has been little enthusiasm for this within the World Wide Web community. Instead, it is believed that a straightforward filter program should be used to map HTML and HTML+ documents into a format which is strictly compliant with HyTime, when this becomes appropriate.

A simple example of HTML+

The following is a simple example of an HTML+ document,which illustrates the basic ideas involved in SGML.

```
<title>A Simple HTML+ Document</title>

<h1 id="a1">This is a level one header</h1>

<p> This is some normal text which will wrap at the window margin. You
can emphasise <em>parts of the text</em> if you wish.

<p> This is a new paragraph. Notice that unlike title and header tags,
there is no matching end tag.
```

The text of the document includes tags which are enclosed in <angle brackets>. Many tags have matching end tags for which the tag name is preceded by the "/" character. The tags are used to markup the document's logical elements, for example, the title, headers and paragraphs. Tags may also be accompanied by parameters, e.g. the "id" attribute in the header tag, which is used to define potential destinations for hypertext jumps.

Unlike most document formats, SGML leaves out the processing instructions that determine the precise appearence of the document, for example the font name and point size, the margins, tab settings and how much white space to leave before and after different elements. The rendering software makes these choices for itself (perhaps guided by user preferences), and so can avoid problems with different page sizes or missing fonts.

Logical markup also preserves essential distinctions that are often lost by lower level procedural formats, making it easier to carry out operations like indexing, and conversion into other document formats.

Practical experience has shown that people often make mistakes when they have to type in the markup for themselves. As a result, most browsers are tolerant of bad markup. This problem is being minimised by keeping the format as simple as possible and encouraging the development of WYSIWYG editors.

The HTML+ Document Format

The following sections go through the various features of the format with suggestions as to how browsers should render them. The DTD for HTML+ is given in Appendix I.

Parsing HTML+ Documents

By default, HTML+ documents are made up of 8-bit characters in the ISO 8859 Latin-1 character set. In future, 16 bit character sets may be used to cover a wider range of languages. The HTTP network protocol uses the MIME standard (RFC 1341) to specify the document type and the character set. It is assumed that the chosen character set includes the printable 7 bit US ASCII characters as a subset.

The DTD specifies the syntax of the document structure, in particular, which tags are permitted in any given context. Certain tags are only permitted at the start of the document. Tags and attribute names are case insensitive, thus <TITLE> is equivalent to <title>. End tags may be minimised to </> instead of say </title>.

In general, SGML entity definitions are used to represent characters which would otherwise be confused with markup elements:

&	is represented by	`&`
<	is represented by	`<`
>	is represented by	`>`

Such entity definitions should be used in all places except within attribute values for tags (tag names and attribute names cannot contain these particular characters). Entity definitions can also be used for special characters, e.g. "é" for a small e with an accute accent. The full list is given in Appendix II. Additional entities may be defined within documents using the SGML entity declaration tag !ENTITY, e.g.

```
<!ENTITY % sgml "Standardised General Markup Language">
```

The browser will then insert the full form whenever it comes across "&sgml;".

Repeated white space characters such as space, tab, carriage return, line feed and form feed are ignored except within preformatted text, i.e. it doesn't matter which white space characters you use or how many of them you put between words, or before or after markup elements, the effect is the same as a single space character.

It is recommended that HTML+ documents start with the following external identifier, indicating that the document conforms to the HTML+ DTD. This will ensure that other SGML parsers can process HTML+ documents, without needing to include the DTD with each document.

```
<!DOCTYPE htmlplus PUBLIC "-//Internet/RFC xxxx//EN">
```

HTML+ departs slightly from pure presentation independence by allowing authors to specify rendering hints, e.g. to use a bold font for a given type of emphasis. This step was taken to give authors greater control over the final appearence, and is based upon practical experience with the earlier HTML format. In addition, attribute values are used to distinguish different subcategories of markup, rather than adding extra tags. New logical categories of emphasis etc. can be added in future without needing to change existing browsers. These decisions have made it practical to restrict HTML+ to a very small set of tags.

Backwards Compatibility with HTML

The format is designed to be largely compatible with the earlier format HTML, and HTML+ browsers will be able to display documents in the HTML format with little extra cost. Suggestions on how to map HTML elements to HTML+ are given in Appendix III.

Notes for Implementors

Please ensure that browsers can tolerate bad markup. In practice, this is quite straightforward to achieve, provided a naive top-down SGML parser is avoided. A forgiving parser should be able to cope with tags in unexpected positions, e.g. the <A> tag bracketing a header[2]. Unknown tags should be simply ignored.

Implementors should endevour to make sure that documents can be scrolled efficiently regardless of their length. Always parsing from the start of the document leads to jerky performance. Two strategies for efficiently scrolling through documents are:

a) Establish regular landmarks throughout the document for which the state of the parse is known. The browser can then work forward from the nearest landmark, when it needs to refresh the screen after a scroll operation. The landmarks need updating when users make changes, while using a WYSIWYG editor.

b) When scrolling up, parse backwards to work out the state at earlier points in the document. This can be done via a combination of skipping back, looking for markup which causes a line break etc. and then parsing forward until the current position, to find the change of state. This can be repeated until the parser reaches a point prior to the new top of the window.

Practical experience has shown the importance of providing cues to users on progress in retrieving documents over the network. These will depend on the protocol, but should show how much data has been received at any point. The network connections shouldn't block, and an abort button is essential[3]. It is generally better to avoid displaying the retrieved document in a new window, unless explicitly requested by the user, e.g. by holding down the shift key when clicking the hypertext link.

Normal Text

This is generally shown with a serif font and wraps on the right window margin. It can include:

- Entity references, e.g. "`>`" and "`&eacute;`"
- Significant Line breaks (the BR tag)
- Non-breaking spaces - the SP tag
- Hypertext links - the A tag
- Inlined graphics or icons - the ICON tag
- Various styles of logical emphasis - the EM tag
- Embedded data in an external format, e.g. TeX equations - the EMBED tag
- Input fields for forms - the INPUT tag.

Line breaks and

This tag causes the renderer to start a new line at the current left margin setting. There is no corresponding end tag. The BR tag is *empty*, that is to say, it doesn't act as a container around other text or markup.

---

[2]Headers typically cause a line break and leave a vertical gap. If the hypertext link definition is parsed prior to the beginning of the header, the starting position for the button will be in the wrong place - browsers should therefore adjust this position to the beginning of the text.

[3]For X11 on Unix systems, the *select* system call can be used with non-blocking I/O to poll the event queue at regular intervals. The *XtAddInput* call acts as a wrapper around select for this very purpose. Users can then continue to view the current document as well as being able to click an abort button (which sets a global variable, polled by the comms software). Be careful to disable unsafe actions, e.g. trying to get a second document while still waiting to get the first (a race hazard).

Non-breaking spaces and <SP>

This allows authors to be certain that browsers won't break a line at an inappropriate place. Authors may also use SP at the start of a line to indent text. This is deprecated.

Some authors like to have a slightly longer space after punctuation at the end of sentances. While this is a stylistic issue, browsers need to be able to distinguish periods which denote the end of sentances from those used in abbreviations. One way of doing this in HTML+ is to use the EM tag to delimit abbreviations.

Hypertext Links

When the user clicks on a hypertext link in the document, the current document is replaced by the one referenced by the link. Links can be made to a wide range of document types, based on the URL[4] and URN[5] notations. Some document types permit links to be made to specific sections within a document[6]. The syntax for links within the same document or to documents in the same directory is particularly simple:

```
Links are defined with the <a href="#z1">A tag</a>. HTML+ supports a
number of <a href="links.html">different link types</a>.
```

In a browser this might look like:

Links are defined with the *A tag*. HTML+ supports a number of *different link types*.

The first link is to an anchor named "z1" in the current document. The second is to a file named "links.html" in the same directory as the current document. The caption for the link is the text between the start and end tags. The value for the HREF attribute defines the destination point, and can be abbreviated in certain cases. If practical, word the caption in such a way that continues to make sense when the document is printed out. The link should be shown in a clearly recognisable way, e.g. as a raised button, or with underlined text in a particular color. For displays without pointing devices, it is suggested that a reference number is given in square brackets, which can then be typed by the user.

A more general discussion of hypertext links and their treatment in HTML+ is presented in a later section.

Inlined Graphics or Icons

These are treated like characters and inserted as part of the text, e.g.

```
This line has a egyptian hieroglyph at the end of the
line. <img src="ankh.tiff">
```

The URL notation is used to name the source of the graphics data. The *align* attribute can be used to control the vertical position of the image relative to the current text line in which the IMG element is placed. Use a value of "top", "middle" or "bottom" to align the top, middle or bottom of the image with the current text line. The *seethru* attribute allows authors to include a chromakey, i.e. a colour that designates portions of the image to be left unpainted so that the background shows through. The format for this attribute's value is dependent on the type of graphics data, and has yet to be defined.

Note that you can create simple iconic buttons, e.g.

```
<a href="bigpic.gif"><img src="littlepic.tiff"></a>
```

If the user clicks anywhere on the image, this will cause the browser to retrieve its bigger version. This approach allows users to preview images which may take significant time to download. Note that there is little additional penalty for displaying the same image at multiple points in the document. The *ismap* attribute is provided for backwards compatibility with HTML. When present the browser will send all mouse clicks and drags on the image, to the server.This mechanism is explained in more detail for the FIG tag.

---

[4]The notation for universal resource locators is defined in [Berners-Lee 93b].

[5]The notation for universal resource numbers and the protocol for resolving them to the nearest available copy is currently under study by the IETF URN working group.

[6]At the time this document was written, such links were restricted to named anchors within HTML and HTML+ documents

Sophistocated HTML+ editors should allow authors to modify images using an external editor. Larger images should be specified with the FIG tag.which provides support for flowing text around figures, along with captions, overlays and active areas.

Various Styles of Emphasis[7]

This allows you to emphasise a portion of the text. The simplest approach is:

```
<em>default emphasis, usually shown in an italic font</em>
```

The logical role of emphasis denotes the semantic significance, e.g. a citation, or text to be input by a user for a computer program.The physical style of emphasis controls its appearence. Note that EM elements can include inlined graphics.

Logical Role of Emphasis

It is strongly recommended that the logical role of the emphasis is given with the *role* attribute, e.g.

```
<em role="cite">a citation</em>
```

Providing a logical role allows browsers to apply differing rendering styles according to the role, but more importantly, it allows indexes to be constructed automatically, e.g. the list of bibliographic references in a technical report. These can be used for searching through collections of documents according to semantic keys giving better focussed searches compared with full text indexes .

The list of recommended roles are as follows:

For references to other works:

CITE	a reference to a related work
PUB	a publication containing a referenced work
AUTHOR	an author of a referenced work
EDITOR	an editor of a referenced work
CREDITS	e.g. the rights owner of a photograph
COPYRIGHT	the holder of the copyright
ISBN	for ISBN numbers
ACRONYM	for acronyms like "NATO" and "US"
ABBREV	for abbreviations

For annotations:

FOOTNOTE	shown as footnote or pop-up
MARGIN	shown as margin note or pop-up

For computer instruction manuals:

DFN	defining instance of a term
KBD	something a user would have to type
CMD	command name, e.g. "chmod"
ARG	command arguments, e.g. "-I"
VAR	named place holder, e.g. "filename"
INS	an instanced of a named printer, directory or file etc.
OPT	an option of some kind
CODE	an example of code (shown with a fixed pitch font)
SAMP	a sequence of literal characters

On dumb terminals annotations should be shown in round brackets. Margin notes should be right aligned, and may include graphics via the IMG tag. The set of recommended roles will be kept by the HTML+ registration authority.

---

[7]The name EM was chosen in preference to EMPH because it allows existing HTML browsers to show all HTML+ emphasis in italics. It also allows HTML+ browsers to correctly process the common case for emphasis in HTML documents.

Physical Styles

The appearence can be modified by adding optional rendering hints from the list:

```
<em b>          bold text
<em i>          italic text
<em u>          underlined text
<em sup>        superscript text
<em sub>        subscript text
<em tt>         type writer font (courier)
<em hv>         sans serif font (helvetica)
<em tr>         serif font (times roman)
```

These hints can be combined, e.g.

```
<em b i> for bold italic text </em>
```

Note that these are only hints and may be ignored by browsers. Indeed, arbitrary combinations will present difficulties for most browsers. If the display is limited to a single font, colour or underlining can be used, but should be clearly differentiated from hypertext links and headers. Dumb terminals can use email conventions, e.g. switching to all capitals, or delimiting with the * or _ characters. Subscript and superscript text should be shown in a smaller point size, vertically offset as appropriate.

Browsers may choose to simplify or ignore hints, but should aim to do so in a consistent manner. At the simplest level, browsers can ignore the attributes and render all emphasis in the same style.

Nested Emphasis

Emphasis can be nested as in:

```
<em b>bold text, and <em i>bold italic text</em></em>
```

Nested emphasis is better suited for grouping logical roles together, for instance, you could use the EM to separately tag author, title, and publication, and then wrap these up as a citation. Without this, indexing programs will have difficulty in grouping markup into the correct references.

Embedded data in an external format

The EMBED tag provides a simple form of object level embedding. This is very convenient for mathematical equations and simple drawings. It allows authors to continue to use familiar standards, such as *TeX* and *eqn*. Images and complex drawings are better specified using the FIG or IMG elements. The *type* attribute specifies a MIME content type and is used by the browser to identify the appropriate shared library or external filter to use to render the embedded data, e.g. by returning a pixmap. It should be possible to add support for new formats without having to change the browser's code, e.g. through using a common calling mechanism and name binding scheme. Sophistocated browsers can link to external editors for creating or revising embedded data. Arbitrary 8-bit data is allowed, but &, < and > must be replaced by their SGML entity definitions.

Input Fields for Forms

Input fields can be arranged with considerable freedom, as part of normal paragraphs, preformatted text, lists or tables. Examples of how to do this are given later on in the section describing the FORM tag. The INPUT tag has the following attributes:

name Used to name this input field, e.g. `name="phone number"` (required attribute).

type Defines the type of data the field accepts (the type name is insensitive to upper/lower case). If missing, the field is assumed to a a free text field.

size Specifies the size/precision of the input field according to its type (optional).

value The initial value for the field, or the value when checked for checkboxes and radio buttons (optional, except for radio buttons).

checked When present, this attribute indicates that a checkbox or radio button is selected.

disabled When present, this attribute indicates that this field is temporarily disabled. Browsers should show this by greying out or via a similar visual clue. Users are unable to set the focus to disabled fields, or change their values.

error When present, this attribute indicates that the current value for this field is in error in some way, e.g. because it violates some consistency constraints. Browsers should indicate this by a change to the shape and colour (red) of the field's border. This should be accompanied by an error message and a beep.

The following types of field should be supported:

TEXT	Single or multi-line text entry fields. Use the *size* attribute to specify the width and height in characters, e.g. size="24" or size="32x4".
URL/URN	For fields which expect document references.
INT	For entering integer numbers, the maximum number of digits may be given with the *size* attribute, e.g. size=3 for a 3 digit number[8].
FLOAT	For fields restricted to floating point numbers.
DATE	Restricted to a recognised date format.
CHECKBOX	Use these for simple boolean attributes, or for attributes which can take multiple values at the same time from some set of alternatives.
RADIO	Use these for attributes which can take a single value from a set of alternatives (groups input fields with the same *name*).

For the purposes of sending the contents of a form to a server, as part of a query, the input fields are mapped to a list of properties. In most cases the *name* and current *value* are used to define a property/value pair for each field. Radio buttons and check boxes are left out if they are unselected. This ensures that only the selected radio button yields a property/value pair. By missing out the *value* attribute for check boxes, these fields will map to a simple (value-less) property. The representation of property lists is defined as part of the HTTP protocol.

Browsers can choose to notify the server whenever a field is changed (i.e. when a field looses the focus and its contents have changed) or wait until the form is completed. This choice will depend on network latency.

Headers and Titles

The title tag is generally used to define the window banner when viewing a particular document, e.g.

```
<title>Reference Guide to HTML+</title>
```

This element should appear at the start of the document. There are six levels of headers, H1 to H6, with H1 the most important, and H6 the least. A common convention is to begin the body of the document with a level one header. e.g.

```
<h1>Introduction to HTML+</h1>
```

Header names should be appropriate to the following section of the document, while the document title should cover the document as a whole. There are no restrictions on the sequence of headers, e.g. you could use a level three header following a level one header. Browsers should render headers with a line break before and after the header text. A common convention for headers is to use a sans serif font, e.g. Helvetica, with a smaller point sizes for less significant headers, and a serif font, e.g. Times Roman, for normal text.

Headers can include an identifier, unique to the current document, for use as destinations of hypertext links, e.g.

```
<h1 id="intro">Introduction to HTML+</h1>
```

This allows authors to make links to particular sections of documents. It is a good idea to use something obvious when creating an identifier, to help jog your memory at a later date. WYSIWYG editors may automatically generate the identifiers. In this case, they should also provide a point and click mechanism for defining links, so that authors don't need to deal explicitly with the identifiers.

---

[8]Perhaps the syntax should permit integer ranges, e.g. size="1 to 6", in which case a more appropriate name for the attribute than *size* would be desirable.

The attribute "margin" when present acts as a hint to the browser to insert the header into the margin and causes the following text to be vertically aligned with the start of the margin header. By convention, margin headers are left justified, e.g.

```
<h4 margin> Deleting the Curve </h4>

The Delete command allows you to delete any selected symbol or text
block.
```

Note that headers don't act as containers for the subsequent text. You can group the header and text with the GROUP tag, see later for details.

Indexing

A good index plays an important role in helping users find their way to the material they need. It allows users to type in one or more keywords to see a meaningful list of matching topics. Alternatively they can browse through the index and take advantage of serendipity, and gain a feeling for the limits of what is covered in the associated document. The two approaches can be combined, when the characters typed act dynamically to control the viewing position within the index. Typically each keyword entry in the index is associated with one or more topics. This notion of guiding the user is absent from full text indexes like WAIS, where users are given very little help in choosing the keywords to search on.

Generating a conventional index for a document is a skilled task, and HTML+ allows authors to include annotations for creating an index. These directives can be included with document titles, headers and emphasis etc. using the *index* attribute. This allows each such element to be included in one or more entries in the index, under primary or secondary keys, e.g.

```
<h3 id="z23" index="Radiation damage/shielding from as difficult">Radiation shielding</h3>
```

This resulting index looks like:[9]

```
Radiation damage
    classical target theory
    dominance of
    in molecular mills
    shielding from as difficult
    simple lifetime model
    track-structure lifetime model
Radicals
  and so on.
```

Where each entry is a hypertext link to the associated anchor. The *index* attribute can specify multiple entries, each separated with the ";" character. The optional secondary key (*shielding from as difficult*) is introduced by the "/" character. Secondary keys are useful when the primary key occurs more than once. To allow for future extension, primary keys should not start with the "#" character. This prefix is being reserved to designate indirect index entries. Use "\/", "\;", "\#" and "\\" to escape "/", ";", "#" and "\" respectively.

Paragraphs and Preformatted Text

HTML+ includes support for paragraphs and preformattted or verbatim text.

Defining Paragraphs with <P>

The <P> tag splits normal text into paragraphs. Unlike headers, there is no corresponding end tag, so don't use </P>. The following optional attributes can be used:

id An identifier, unique to this document, which can be used as a destination in a hypertext link. Note that the paragraph tag acts as a container for the paragraph.

---

[9]Taken from K. Eric Drexlers's "Nanosystems, Molecular Macherinery, Manufacturing and Computation".

role The role of the paragraph, see the following list for supported types.

align A rendering hint to the browser to justify lines. The supported values should be: `align="left"`, `align="center"` and `align="right"`. This is useful for single line paragraphs or when the lines are made explicit with the <BR> tag.

indent When present, this hint suggests that the left and right margins are indented by an amount dependent on the browser, e.g. about 4 character widths.

The *role* attribute is used to indicate the logical role of the paragraph, e.g. a stanza in a poem or a cautionary note in a computer manual. Browsers may apply particular rendering styles to certain roles. The role name is case insensitive. The following roles are recommended:

quote A paragraph quoted directly from some other work. Browsers could indent the paragraph and maybe use a different font.

byline Information about the author of the document, e.g contact details. This could be displayed in a different font, and perhaps right aligned.

note Advisory note in an instruction manual. The browser could display a hand icon in the margin.

caution Cautionary note. The browser could display an warning road sign in the margin.

error A note describing error conditions. The browser could indicate the importance of the note by displaying a stop sign in the margin.

An example of a paragraph element:

```
<p role="note"> If you accidentally delete a symbol other than the red
circle, immediately press ALT+BKSP to choose the undo command, and
then select the red circle and delete it again.
```

Paragraphs can be rendered by indenting the first line, or by leaving a vertical gap equal to half the current line spacing. When using the latter style, browsers should take care to avoid this vertical gap when the paragraph element immediately follows a header. This rule ensures that authors can tag paragraphs directly following a header without causing unwanted extra space before the start of the text.

Ordered, Unordered and Definition Lists

There are three kinds of lists: ordered or numbered lists, unordered lists and definition lists. Ordered and unordered lists can be nested arbitrarily, and browsers should progressively inset the left margin for each level of nesting.

Ordered Lists with <OL>

The list items are automatically numbered, e.g.

```
<OL>
  <LI>Wake up
  <LI>Get dressed
  <LI>Have breakfast
  <LI>Drive to work
</OL>
```

Is displayed as:

1) Wake up
2) Get dressed
3) Have breakfast
4) Drive to work

The *compact* attribute when present has the effect of reducing interitem spacing, e.g. <ol compact>. Authors can also make both the OL tag and the LI tag potential destinations for hypertext links with the *id* attribute. List item text can include normal text and paragraph elements, but not headers.

Unordered Lists with <UL>

These are bulleted lists, e.g.

```
<UL>
  <LI>Wake up
  <LI>Get dressed
  <LI>Have breakfast
  <LI>Drive to work
</UL>
```

Is displayed as a bulleted list:

- Wakeup
- Get Dressed
- Have breakfast
- Drive to work

The *compact* attribute when present has the effect of suppressing bullets and reducing interitem spacing, e.g. <ol compact>. Multicolumn lists can be requested with the *narrow* attribute, e.g. <ul narrow>. This causes the browser to try to lay out the list as a number of columns, depending on the window width. This attribute should only be used when all the items are less than 20 characters long. Authors can also make both the UL tag and the LI tag potential destinations for hypertext links with the *id* attribute. List item text can include normal text and paragraph elements, but not headers. For nested unordered lists, browsers may use different bullet symbols for different levels, in addition to progressively insetting the left margin. The *src* attribute on the LI tag can be used to specify an icon for use in place of the standard bullet symbols.

Definition Lists with <DL>

These consists of pairs of terms <DT> and definitions <DD>. The following example is part of a french dictionary:

```
<DL>
  <DT>endetter
  <DD>Engager dans des dettes

  <DT>endeuiller
  <DD>Plonger dans le deuil, remplir de tristesse

  <DT>endiablé, ée
  <DD>D'une vivacité extrême
</DL>
```

Is commonly displayed as:

endetter Engager dans des dettes

endeuiller Plonger dans le deuil, remplir de tristesse

endiablé, ée D'une vivacité extrême

With the *compact* attribute, e.g. <dl compact>, this is altered to:

endetter Engager dans des dettes

endeuiller Plonger dans le deuil, remplir de tristesse

endiablé, ée D'une vivacité extrême

In this style, the term and definition appear in the same paragraph, with the term text emphasised in a bold font. The definition text follows on, and wraps to a left margin a little further inset than the term text. This style is common place in dictionaries.

Term text following the <DT> is restricted to normal text. The definition text after the <DD> tag can additionally include paragraph elements and ordered/unordered lists. Headers are not allowed in either case. Authors can make the DL, DT and DD tags potential destinations for hypertext links with the *id* attribute.

Authors are reminded to check that DT and DD are paired up. Common misunderstandings lead to people repeating DD tags to separate paragraphs (use <P> instead), or leaving out the DT tag altogether to indent text (use <p indent> or <group indent>). The ability of browsers to cope with bad markup seems to encourage such problems, which will hopefull fade away as wysiwyg editors become commonplace

Figures

Figures provide great flexibility:

- linked or embedded graphics
- control of picture alignment and text flow
- Figure description for when the image can't be shown
- caption placement
- scaled or pixel-based coordinates
- hypertext links with active areas
- text and image overlays

The following simple example will set the scene for the description of the various features:

```
<fig align="right" src="map.gif">How to get to my house </fig>
```

Here, the image is defined by a link to an external document. The caption "How to get to my house" will appear at the bottom of the image. The *align* attribute directs the browser to display the figure at the right of the widow, and to flow subsequent text around the left of the image.

Using embedded graphics data

Instead of the *src* attribute, you can include an EMBED element immediately following the <fig> tag. This is useful for graphs etc. defined in an external format.

Figure Description

The FIGD tag allows you to give a textual description which can be shown when the figure itself can't be shown, e.g. for browsers working on dumb terminals, e.g.

```
<FIGD> This is an aerial photograph of central London, showing
Buckingham Palace and the Houses of Parliament. On the left you can see
Hyde Park and in front the Albert Hall and the Natural History
Museum.</FIGD>
```

Alignment and Text Flow

The *align* attribute controls the horizontal position of the figure: "left", "right", or "center". The default is "left". Browsers may flow text when there is sufficient room, unless the figure is center aligned or the *noflow* attribute is present.

Caption Placement

The *cap* attribute allows you to ask the browser to position the caption text to the "left", "right", "top" or "bottom". The default is to place the caption at the bottom of the figure. Text flow will occur around the figure and caption, leaving a suitable gully. The browser will ignore this attribute if there is insufficient room for the requested placement.

Pixel-base or Scaled Coordinates

The upper left of the figure is designated as x,y = (0, 0), with x increasing across the page, and y down the page. If points are given in real numbers, the lower right is taken as being (1.0, 1.0), otherwise with integer values, the coordinates are assumed to be in pixels[10]. Note that using scaled coordinates is much safer, especially for graphics! The extent of the image in pixels may change, e.g. as a result of format negotiation with the server, and by retrieving images with lower resolution when network performance is poor.

Active areas

The *ismap* attribute causes the browser to send mouse clicks on the figure, back to the server using the selected coordinate scheme. The mouse button-up event is sent with the URL formed by adding "?x,y" as a suffix to the URL for the current document. You can also designate rectangular regions of interest in the picture by holding the mouse button down while dragging the mouse. The browser should show a rubber band outline for the rectangle defined by the current location of the mouse pointer and the point at which the mouse button was pressed. The region is named by taking the current URL and adding the suffix: "?x1,y1;x2,y2", where (x1, y1) and (x2, y2) define the points at which the mouse button went down and came up, respectively. The *ismap* mechanism is relatively slow, but makes sense when the active regions change their boundaries over time, e.g.

```
<fig ismap src="weather.gif">Click on your area for todays weather</fig>
```

You can also designate arbitrary areas of the figure as hypertext links. Mouse clicks are handled locally, and the browser can provide visual clues that the pointer is over an active area, for example, by changing the pointer from an arrow to a hand symbol, or highlighting the area in some way.

Active areas are defined with the FIGA tag. This has two attributes:

href	A URL specifying the link to traverse when clicked (required)
area	Defines a polygonal[11] area as a list of points: "x1, y1; x2, y2; ..." (optional)

The *area* attribute lists a sequence of points defining a polygon. Closure is ensured by joining the last point in the list to the first (i.e. a triangular area is defined with a list of 3 points).When the *area* attribute is missing, the whole of the picture is assumed. Polygons may be non-convex or even intersect themselves, thereby complicating the definition of what is enclosed by the polygon. Holes should be excluded. Note that active areas defined with FIGA take precedence over the *map* mechanism.

Overlays

The FIGT tag allows you to position text and image overlays on top of the figure, e.g.

```
<fig src="map.giff">
  <figt at="0.2, 0.3" framed>A text overlay</figt>
  The figure caption
</fig>
```

The overlay can contain a wide variety of elements including text, images (IMG), lists and tables. Figures shouldn't be nested. Any hypertext links in the overlay text will take precedence over the *href* attribute in FIGT. The following attributes are permitted:

at	The upper left of the overlay, relative to the figure.
width	As a fraction of the figure, e.g. width="0.3". This allows you to limit the lengths of wrapped text lines. The vertical extent is then determined automatically.
framed	Directs the browser to draw a frame around the overlay and to colour in the background in some way.
href	Allows you to make the overlay into a hypertext button.

---

[10]This mechanism was designed to be backwards compatible with the *ismap* feature as used with IMG in HTML, and as a consequence forces the choice of y increasing down rather than up the page. A simple test to distinguish the two schemes is to check if the "," character occurs anywhere in the list of points.

[11]The code for hit testing polygons is tricky, but quite fast. A public domain version of the code would be helpful.

Tables

Tables are defined with the TBL tag. Cells are designated as being headers or data. You can join adjacent cells, e.g. to define a header spanning two columns.

An Example of a Table

	average		other
	height	weight	category
males	1.9	0.003	yyy
females	1.7	0.002	xxx

This is defined by the markup:

```
<tbl border>
   <tt top> An Example of a Table
   <th rowspan=2> <th colspan="2"> average <th> other <tr>
   <th> height <th> weight <th> category <tr>
   <th align=left> males <td> 1.9 <td> .003 <td> yyy <tr>
   <th align=left> females <td> 1.7 <td> .002 <td> xxx
</tbl>
```

The *border* attribute for TBL directs the browser to draw borders. The *compact* attribute is used when you want the table to appear in a smaller size.

The optional <tt> tag defines a title. By default (i.e. when *top* is missing) this should be positioned below the table. The <th> and <td> tags define header or data cells respectively. The <tr> tag acts as a separator between rows. In the example, you can see that the first header in each of the first two rows is void.

TH, and TD all have the same permitted attributes:

colspan	Columns spanned by this cell, see example
rowpan[12]	Rows spanned by this cell, see example
align=left	Left justify the cell's content
align=center	Center justify the cell's content
align=right	Right justify the cell's content

By default, headers are centered, while other cells are left justified. If practical, browsers should be smarter than this, e.g. if all the cells in a column are shorter than the column header, then indent the cells to make them appear under the middle of the header.

Browsers need to carry out a pre-parse (e.g. when sizing the vertical scroll bar) in order to determine the number of columns and their widths. The following guidelines may be useful:

- There is no need to declare empty cells at the end of a row, so the number of columns for the table is given by the row with the most columns.
- Restricting text to a fixed pitch font may simplify matters.
- If a column only contains numbers or empty cells then align on units and set width to maxium precision needed (before and after decimal point, allowing for an exponent). This rule also applies when currency symbols are used.
- Otherwise set column width to the minimum of a threshold width and the maximum text length for all cells in the column. Text is left aligned and wrapped if it exceeds the chosen column width.

---

[12]This is tricky to handle. The parser should carry a spanned cell over to the next row, the definition of which should miss out the spanned cell, i.e. the next row will have one fewer explicit cell definitions.

The threshold column width can be set according to the number of columns and the width of the display window. It is also necessary to take the column headers into account in this process. Header text wraps to the next line if the column is too narrow. Browsers will by default center the header in the column.

A complication occurs when a header or data cell spans more than one column, as specified by the *s* attribute. This can be used to give complex headers which share a header between columns followed by individual headers on the next line.

Vertical gaps can be introduced with the <tb> element - this inserts 1/2 line space into the next row. Header and Data rows can be intermixed. Authors can use alternate header and data rows when the rows alternate between text and numbers. The vertical alignment of numbers only applies to data fields.

Tables which don't fit into this model should be defined as figures using an external format, e.g. Postscript, Tex or Computer Graphics Metafile.

Forms

A document can include one or more forms. Each form is defined by a FORM element, which contains a number of input fields laid out with normal and preformatted text, lists and tables. The browser should manage the input focus, e.g. with the tab key and mouse clicks. The Return key can be used to mean that the user has filled in the form and wants the appropriate action to be taken. Browsers may also display "Accept" and "Cancel" buttons as part of the document (or perhaps on another part of the browser). Note that forms shouldn't be nested.

The action to be taken is specified by the *action* attribute of the FORM tag. If missing the URL for the current document is assumed. This attribute uses a URL to specify a server to query, or an email address to send the form to. When sending the form to a server as a query, the form's contents are encoded as a property list (see definition of the INPUT tag). The precise encoding is dependent on the HTTP protocol and defined in [Berners-Lee 93c][13]. When the form is to be mailed, it is first converted into plain text, closely resembling the appearence on the screen. You can include multiple RFC 822 mail headers with the MH tag. The *hidden* attribute may be used to hide the headers when browsing the document. The following is an example of a simple questionaire:

```
<form action="mailto:www_admin@info.cern.ch">

<mh hidden>
  Subject: WWW questionaire
</mh>

Please help us to improve the World Wide Web by filling in the
following questionaire:

<p>
Your organisation? <input name="org" size="48">

<p> commercial? <input name="commerce" type="checkbox">
How many users? <input name="users" type="int">

<p> Which browsers do you use?

<ol compact>

<li> X Mosaic <input name="browsers" type="checkbox" value="xmosaic">

<li> Cello <input name="browsers" type="checkbox" value="cello">

<li> Viola <input name="browsers" type="checkbox" value="viola">

<li> Others? <input name="other browsers" size="48x4">

</ol>

A contact point for your site: <input name="contact" size="48">

<p>Many thanks on behalf of the WWW central support team.

</form>
```

[13]This and the *ismap* feature rely on the forthcoming definition of HTTP as an official Internet standard.

Floating Panels

The PANEL tag can be used to define panels or boxes which are free to float with respect to the standard flow of text. These are often used in magazine articles for asides on background material. The panel is typically shown with a distinctive background colour and border. The lay out software positions the panel to coincide with the page boundaries in printed media. For on-line use, panels can be rendered as pop-up windows. The body of the panel can be defined by a link to a separate document or included in the current document.

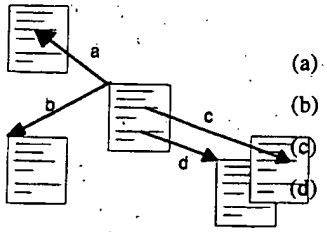
The following optional attributes are permitted with the <panel> tag:

id	An identifier, unique to this document, which can be used as a destination in a hypertext link.
at	An identifier elsewhere in this document. The panel mustn't be placed before this point. (Defaults to the current position if the *at* attribute is missing).
href	This attribute allows authors to fill the panel from a separate document, as specified by a URL. Note that the matching end tag: </panel> is always needed.

The text contained by the panel element can include any of the markup elements and looks like a separate document (panels themselves can't be nested). If the *href* attribute is used the text delimited by <panel> ... </panel> may be used as the caption for a pop-up. The *at* attribute allows you to include the panel definition at a convenient point in the HTML+ document, rather than interrupting the main flow of the document.

More on Links

Before describing the details of how links are represented in HTML+ it is worth looking more generally at the nature of hypertext links. First a terminological point: a *node* is the atomic unit for information retrieval, while *documents* may consist of one or more nodes, perhaps arranged as a hierarchy. A node may even be shared between several documents. Hypertext links start and end on nodes or anchor points within nodes.



The diagram illustrates the basic possibilities:

(a) Link from a node to an anchor

(b) Link from a node to a node

(c) Link from an anchor to another anchor

(d) Link from an anchor to a node

Links in HTML+ are represented with the LINK and A tags. The LINK tag is used for cases (a) and (b), while the A tag is used for cases (c) and (d). These links are held in the source node only, so there is a risk that the destination may have disappeared. Organisations can manage this risk by continued support for a few well published nodes (servers can use redirection to hide name changes). Links to other subsidiary nodes are at higher risk. This structured approach allows people to become familiar with the major routes through the web, without needing to worry about the minor routes.

In most cases URLs and URNs explicitly specify a node/anchor. The nodes may be explicit files or generated as the result of some process invoked by the server, e.g. a hypertext listing of a directory or a list of matches for a given search string. The search string can be explicitly encoded as part of a link, or dynamically defined by the user (see the ISINDEX tag, as described later on).

Links may be held separately from the source and destination nodes[14]. This is particularly appropriate for annotations and discussion groups. For example, consider making an annotation on a document held by a server located far away in another organisation. You could take a local copy and directly annotate it, but this is only appropriate for private use. The remote server might even support a protocol to add your annotations in place. More likely though, you will have to use an annotation server. This mechanism can be used to obtain a

[14]These correspond to HyTime's *ilink* architectural form.

copy of the document with the annotations inserted as hypertext links and shown as pop-ups or separate documents.

Context Dependent Links

For discussion groups, responses are made asynchronously, and include one or more references to other articles. In this situation, context dependent links are appropriate. The resolution to an explicit node can be carried out by either the client or server. The former approach is often appropriate, but requires special support, e.g. for network news and nntp.

Context dependent links are also useful for links to the table of contents for documents consisting of multiple nodes, when some of the nodes also appear in other documents. The appropriate table of contents for a given node will depend on which document is currently being viewed. In this case, the context will depend on how the current node was reached. This is quite simple to track if the links from the table of contents are differentiated from cross reference links.[15]

Hypertext paths are recommended routes through a set of nodes, and generally shown by next and previous buttons on a toolbar. Paths can be defined using explicit links in a node, or held separately in another node. The latter case once again, depends on the context. Paths and tables of contents all fall under the general category of navigating around a hierarchy of nodes forming a document too large or unwieldy to be held in a single node.

Types of Links

There are several motivations for differentiating between types of links:

how it is viewed	The potential to show different cues depending on the type and size of the node to be retrieved. If this information is explicitly stated as part of the link, there is the risk that it will become out of step with the linked node.
what happens	Whether the linked document replaces the current one, or appears in a new window, or as a pop-up overlay on top of the current one.
printed appearance	Whether links are treated as references, footnotes or as separate sections
effect on context	After traversing the link, will there be implicit values for the table of contents, and hypertext path etc?

Link Attributes

The A tag has the following attributes:

id	An identifier unique to this document which can act as a hypertext anchor
name	The same as *id* and included for backwards compatibility with HTML. New documents should use the *id* attribute for consistency with the other tags.
href	The URL or URN identifying the destination of the link.
role	A string giving the role of the link, e.g. `role="partof" or "annotation"`
effect	A string defining how the linked node is shown: "replace", "new", "overlay", with the default effect of replacing the current document.
print	How should the link be printed: "reference", "footnote" and "section", defaulting to "reference" (i.e. a footnote stating the link's URL).
title	The title to show when otherwise undefined for the node.
type	The MIME content type for the linked node for use with presentation cues.
size	The size in bytes for the linked node. This allows the browser to show a gauge indicating progress in retrieving long documents or images etc.

The LINK tag has only the *href* and *role* attributes.

---

[15]This is more general than deriving the role of the link from that of the node alone.

The *role* attribute is appropriate when context dependent properties such as table of contents (toc) are implied for the linked node, e.g. if the current node is a toc (as defined by the html or group tags) and the link has the role "partof", then the current node should act as the *toc* for the linked node. This property propagates down "partof" links, but not normal links. The *next* and *prev* properties are given by the sequence of "partof" links in the parent node. The *parent* property is only defined if the current node was reached via a "partof" link.

The LINK tag is used to express these properties in an explicit form, e.g.

```
<LINK href="toc.html" role="toc">
```

The recommended property names are:

toc	Table of contents for current node.
next	The next node in a hypertext path.
prev	The previous node in a hypertext path.
parent	The next level up in the hierarchy.
style	The style sheet appropriate to this node.

Style sheets provide a way for authors to express their detailed preferences for fonts, and layout, whether for the screen or when the node is printed out. A possible format is given in [Raisch 93].

The *effect* attribute is a hint and may be disregarded by browsers. It allows you to click on an image and to see a linked movie as an overlay at the same position. The browser tries to position the overlay at the same origin as the link. In some cases, the linked node is a description of the current node. By including `effect="new"`, the linked node will appear in a new window so that users can see both nodes at the same time. This hint should be used sparingly!

The *print* attribute makes it practical to print nodes along with relevant linked nodes. By default each link appears as a footnote stating the link's URL. Short nodes can be included in their entirety as footnotes, and longer ones as sections in their own right. This approach could be extended in future, to reorder the sequence of nodes from that defined by the position of the links in the source node, and to control the level that nodes appear as, e.g. chapter, section or subsection.

The *title* attribute is useful for nodes without titles of their own, e.g. Gopher menus. The *type* attribute can be used to show cues for the node type, e.g. iconic decorations[16]. The size attribute allows browsers to show a gauge on how much of a document has been retrieved at a any time. These attributes are liable to get out of step with the target node, and should be treated as hints only.

Groups

The GROUP tag allows you to define arbitrary groups, e.g. books, chapters, and sections. The *role* attribute is used to name the logical role of the group. You can use most markup elements inside a group element, including group itself. The *inset* attribute is a rendering hint to inset the left margin. Using the A tag with role="partof" allows you to designate a node as being included within the group, allowing hierarchies of groups which cross multiple nodes. See previous discussion of how properties are propagated.

Groups offer opportunities for presenting and searching documents at different levels of abstraction. For example, you might first describe a book by its title, author, publisher and ISBN number. The next level down could add a cover illustration together with a summary of the book's contents, some comments by reviewers and a short biography of the author. A number of books to be presented in an iconic form using a miniature version of the "cover page". Publishers could include copyright and other details in a standard place.

---

[16]The appropriate cue might also depend on the role of the link, e.g. for annotations browsers could show an icon of a drawing pin (as in attaching a note to a pin board). The colour of the pin could then vary according to the media type of the annotation.

Change Bars

Authors can indicate a part of a document has been changed using the CHANGED tag. This may appear anywhere that normal text is allowed (as designated by the entity reference `%text;` in the DTD). This tag signals the beginning or end of changes, which should be rendered by a vertical bar in the left margin. The tag can have one (but not both) of the following attributes:

id An identifier unique to the current document, which can also be used as a a destination for hypertext links. This signals the beginning of changes.

idref This must be an identifier matching the preceding changed element. It signals the end of changes. Note that you mustn't have both *id* and *idref* together.

Miscellaneous Tags

The remaining tags must appear at the start of the node like TITLE and LINK. They describe properties which apply to the node as a whole.

The HTML tag.

This is intended to provide short informal classifications for use in cataloging documents held by HTTP servers. The *role* attribute identifies the purpose of the node, for example `<html role="home page">`. Another common role is "toc" for table of contents. See previous discussion of link attributes.

The ISINDEX tag

This specifies that the URL designated with the *href* attribute is searchable (defaults to this document's URL). Browsers should allow users to enter a search string of one or more keywords. When the Return key is pressed the search string is appended to the designated URL, after a "?" character and sent to the server specified by the URL. Certain characters should be escaped as specified by the standard URL syntax, for example, the space character is mapped to "+". The newer HTTP protocol offers an alternative means for specifying that documents are searchable. In this case, the search string is sent as part of an RFC 822 style header. See [Berners-Lee 93c] for details.

The NEXTID tag

This is used by browsers that automatically generate identifiers for anchor points. It specifies the next identifier to use, to avoid confusionwith old (deleted) values, e.g. `<nextid n="id56">`. The identifier should take the form of zero or more letters followed by one or more digits. The numeric suffix should be incremented to generate successive identifiers.

The BASE tag

The *href* attribute gives the full URL of the document, and is added by the browser when the user makes a local copy. Keeping the original URL in a local copy is essential when subsequently viewing the copy as it allows relative URLs in the document to be resolved to their original references.

Note that one motivation for using relative URLs is to allow a group of documents to be copied without the need to alter any links between them. In this case, the BASE tag is inappropriate, since it would cause links to be intepreted as being to the original documents rather than their copies.

The HEAD and BODY tags

The HEAD tag can be used to delimit properties which apply to the document as a whole, and if used, must be present at the start of the document, followed by the BODY tag which then delimits the rest of the document.

Acknowledgements

I would like to thank the many people on the *www-talk* mailing list who have contributed to the design of HTML+ and to the management of HP Labs for their support during this work.

David Raggett, Hewlett Packard Laboratories, July 1993.

Email: `dsr@hplb.hpl.hp.com`, *Phone*: `+44 272 228046`

Appendix I - The HTML+ DTD

```
<!DOCTYPE HTMLPLUS [

<!-- DTD for HTML+ It assumes the default <!SGML> declaration

Markup minimisation should be avoided with the exception of </>
for the endtag. Browsers should be forgiving of markup errors.

Common Attributes:

   id       the id attribute allows authors to name elements such as
            headers and paragraphs as potential destinations for links.
            Note that links don't specify points, but rather extended
            objects.

            index allows authors to specify how given headers etc should
            be indexed as primary or secondary keys, where "/" separates
            primary from secondary keys, ";" separates multiple entries

-->

<!-- ENTITY DECLARATIONS

  <!ENTITY % foo "X | Y | Z"> is a macro definition for parameters and in
  subsequent statements, the string "%foo;" is expanded to "X | Y | Z"

  Various classes of SGML text types:

    #CDATA   text which doesn't include markup or entity references

    #RCDATA  text with entity references but no markup

    #PCDATA  text occurring in a context in which markup and entity
             references may occur.
-->

<!ENTITY % URL "CDATA" -- a URL or URN designating a hypertext node -->

<!ENTITY % text "#PCDATA|A|IMG|EM|EMBED|INPUT|SP|BR|CHANGED">

<!ENTITY % paras "P|PRE|FIG">

<!ENTITY % lists "UL|OL|DL">

<!ENTITY % misc "TBL|FORM|PANEL|GROUP">

<!ENTITY % heading "H1|H2|H3|H4|H5|H6">

<!ENTITY % table "%text;|P|%heading;|%lists;">

<!ENTITY % main "%heading;|%misc;|%lists;|%paras;|%text;">

<!ENTITY % setup "(TITLE? & HTML? &ISINDEX? & NEXTID? & LINK* & BASE?)">

<!--

<!ELEMENT tagname - - CONTENT> elements needing closing tags

<!ELEMENT tagname - O CONTENT> elements without closing tags

<!ELEMENT tagname - O EMPTY> elements without content or closing tags

   The content definition is:

      a)    an entity definition as defined above

      b)    a tagname

      c)    (brackets enclosing the above)

   These may be combined with the operators:

      A*    A occurs zero or more times

      A+    A occurs one or more times
```

```
      A|B   implies either A or B
      A?    A occurs zero or one times
      A,B   implies first A then B
-->

<!ELEMENT HTMLPLUS O O ((HEAD, BODY) | ((%setup;), (%main;)*))>

<!ELEMENT HEAD - - (%setup;)>

<!ELEMENT BODY - - (%main;)*>

<!-- Document title -->

<!ELEMENT TITLE - - (#PCDATA | EM)+>

<!ATTLIST TITLE
      id       ID      #IMPLIED -- link destination --
      index    CDATA   #IMPLIED -- entries for index compilation -->

<!-- Document role for cataloging documents held by servers -->

<!ELEMENT HTML - O (EMPTY)>

<!ATTLIST HTML role CDATA #IMPLIED -- home page, index, ... -->

<!-- Floating panel which can be moved around relative to the normal text
flow. Often rendered with a different background and possibly framed. The
panel can be anchored to a named point in the document as specified by
the AT attribute. The panel may be placed at that point or after, but not
before.
-->

<!ELEMENT PANEL - - (TITLE?, (%main;)*)>

<!ATTLIST PANEL
      id       ID      #IMPLIED -- defines link destination --
      at       IDREF   #IMPLIED -- anchor point --
      index    CDATA   #IMPLIED -- entries for index compilation -->

<!-- Document headers -->

<!ELEMENT (%heading;) - - (#PCDATA | EM)+>

<!ATTLIST (%heading;)
      id       ID      #IMPLIED -- defines link destination --
      index    CDATA   #IMPLIED -- entries for index compilation -->

<!-- logical emphasis with optional style hints -->

<!ELEMENT EM - - (%text;)*>

<!ATTLIST EM
   role   CDATA  #IMPLIED -- semantic category e.g. CITE --
   b      (b)    #IMPLIED -- render in bold font --
   i      (i)    #IMPLIED -- render in italic font --
   u      (u)    #IMPLIED -- underline text --
   tt     (tt)   #IMPLIED -- render in typewriter font --
   tr     (tr)   #IMPLIED -- render in serif (Times Roman) font --
   hv     (hv)   #IMPLIED -- render in sans serif (Helvetica) font --
   sup    (sup)  #IMPLIED -- superscript --
   sub    (sub)  #IMPLIED -- subscript --
   index  CDATA  #IMPLIED -- entries for index compilation -->

<!-- Paragraphs with different roles and optional style hints -->

<!ELEMENT P - O (%text;)+>

<!ATTLIST P
   id       ID          #IMPLIED -- link destination --
   role     CDATA       #IMPLIED -- semantic role --
   align    CDATA       #IMPLIED -- left, center or right --
   indent   (indent)    #IMPLIED -- indented margins --
   index    CDATA       #IMPLIED -- entries for index compilation -->

<!ELEMENT BR - O EMPTY -- line break -->
```

```
<!ELEMENT SP - O EMPTY -- unbreakable space -->

<!-- Preformatted text with fixed pitch font, respecting original spacing
and newlines. Authors can also request proportional fonts. Further
control is possible with EM. -->

<!ELEMENT PRE - - (%text;)+>

<!ATTLIST PRE
   id       ID       #IMPLIED -- link destination --
   style    CDATA    #IMPLIED -- various styles --
   tr       (tr)     #IMPLIED -- serif (Times Roman) font --
   hv       (hv)     #IMPLIED -- sans serif (Helvetica) font --
   width    NUMBER   #IMPLIED -- e.g. 40, 80, 132 --
   index    CDATA    #IMPLIED -- entries for index compilation -->

<!-- Lists which can be nested -->

<!ELEMENT OL - - (LI | UL | OL)+ -- ordered list -->

<!ATTLIST OL
   id       ID         #IMPLIED
   compact  (compact)  #IMPLIED
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!ELEMENT UL - - (LI | UL | OL)+ -- unordered list -->

<!ATTLIST UL
   id       ID         #IMPLIED      -- link destination --
   compact  (compact)  #IMPLIED      -- reduced interitem spacing --
   narrow   (narrow)   #IMPLIED      -- narrow perhaps multi columns --
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!-- List items for UL and OL lists -->

<!ELEMENT LI - O (P|%text;)+>

<!ATTLIST LI
   id       ID         #IMPLIED
   src      %URL;      #IMPLIED -- icon for use in place of bullet --
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!-- Definition Lists (terms + definitions) -->

<!ELEMENT DL - - (DT,DD)+ -- DT and DD *MUST* be paired -- > <!ATTLIST DL
   id       ID         #IMPLIED
   compact  (compact)  #IMPLIED
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!ELEMENT DT - O (%text;)+ -- term text -- >

<!ELEMENT DD - O (P|QUOTE|UL|OL|%text;)+ -- definition text -- >

<!ATTLIST (DT|DD)
   id       ID         #IMPLIED
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!-- Tables with titles and column headers, e.g.

   <tbl border>
      <tt> An Example of a Table
      <th> <th s="2"> average <th> other <tr>
      <th> <th> height <th> weight <th> category <tr>
      <td> males <td> 1.9 <td> .003 <td> yyy <tr>
      <td> females <td> 1.7 <td> .002 <td> xxx
   </tbl>

-->

<!ELEMENT TBL - - (TT?, (TH|TD|TR|TB)*) -- mixed headers and data -->

<!ATTLIST TBL
   id       ID         #IMPLIED
   compact  (compact)  #IMPLIED      -- if present use compact style --
   border   (border)   #IMPLIED      -- if present draw borders --
   index    CDATA      #IMPLIED       -- entries for index compilation -->

<!ELEMENT TT - O (%text;)+ -- table title -->
```

```
<!ATTLIST TT top (top) #IMPLIED -- place title above table -->

<!ELEMENT TH - O (%table;)* -- a header cell -->

<!ATTLIST TH
   colspan  NUMBER     1          -- columns spanned --
   rowspan  NUMBER     1          --. rows spanned --
   align    CDATA    #IMPLIED     -- left, center or right -->

<!ELEMENT TD - O (%table;)* -- a data cell -->

<!ATTLIST TD
   colspan  NUMBER     1          -- columns spanned --
   rowspan  NUMBER     1          --. rows spanned --
   align    CDATA    #IMPLIED     -- left, center or right -->

<!ELEMENT TR - O EMPTY -- row separator -->

<!ELEMENT TB - O EMPTY -- vertical break of 1/2 line spacing -->


<!-- Forms composed from input fields and selection menus

These elements define fields which users can type into or select with
mouse clicks. The browser should manage the input focus e.g. with the
tab/shift tab keys and mouse clicks.

The enter/return key is then taken to mean the use has filled in the form
and wants the apppropriate action taken:
   -  send as query/update to WWW server
   -  email/fax to designated person

The action is specified as a URL, e.g. "mailto:dsr@hplb.hpl.hp.com You
can specify additional mail headers with the MH tag:

<MH>Subject: Please add me to tennis tournament</MH>

Each FORM should include one or more INPUT elements which can be layed
out with normal and preformatted text, lists and tables.
 -->

<!ELEMENT FORM - - (MH, (%main;)*)>

<!ATTLIST FORM
   id       ID         #IMPLIED
   action   %URL;      #IMPLIED
   index    CDATA      #IMPLIED -- entries for index compilation -->

<!ELEMENT MH - - CDATA -- one or more RFC 822 header fields -->

<!ATTLIST MH hidden (hidden) #IMPLIED -- hide the mail headers from view -->

<!-- INPUT elements should be defined within a FORM element.

Users can alter the value of the INPUT element by typing or clicking with
the mouse. Use radio buttons for selecting one attribute value from a set
of alternatives. In this case there will be several INPUT elements with
the same name. Attributes which can take multiple values at the same time
should be defined with checkboxes: define each allowed value in a
separate INPUT element but with the same attribute name. For checkboxes
and radio buttons, the value doesn't change, instead the state of the
button shown by the presence or absence of the checked attribute in each
element.

The size attribute specifies the size of the input field as appropriate
to each type. For text this gives the width in characters and height in
lines (separated by an "x"). For numbers this gives the maximum
precision.

-->
```

```
<!ELEMENT INPUT - O EMPTY>

<!ATTLIST INPUT
   name     CDATA     #IMPLIED -- attribute name (may not be unique) --
   type     CDATA     #IMPLIED --TEXT,URL,INT,FLOAT,DATE,CHECKBOX,RADIO--
   size     CDATA     #IMPLIED -- e.g."32x4" for multiline text --
   value    CDATA     #IMPLIED -- attribute value (altered by user) --
   checked  (checked) #IMPLIED -- for check boxes and radio buttons --
   disabled(disabled)   #IMPLIED    -- if grayed out --
   error    (error)   #IMPLIED -- if in error -->

<!-- Embedded Data

You can embed information in a foreign format into the HTML+ document.
This is very convenient for mathematical equations and simple drawings.
Images and complex drawings are better specified as linked documents
using the FIG or IMG elements.

Arbitrary 8 bit data is allowed but any occurrences of the following
chars must be escaped as shown:

      "&"      by      "&amp;"
      "<"      by      "&lt;"
      ">"      by      "&gt;"

The browser can pipe such data thru filters to generate the corresponding
pixmap The data format is specified as a MIME content type, e.g.
"text/eqn"
-->

<!ELEMENT EMBED - - (RCDATA)>

<!ATTLIST EMBED
   id       ID        #IMPLIED
   type     CDATA     #IMPLIED  -- mime content type --
   index    CDATA     #IMPLIED -- entries for index compilation -->

<!-- Figures

The image/drawing is specified by a URL or as embedded data for simple
drawings. The element's text serves as the caption. Use the emphasis with
style = "credits" to record photo credits etc.

-->

<!ELEMENT FIG - - (EMBED?, FIGD?, (FIGA|FIGT)*, (%text;)*)>

<!ATTLIST FIG
   id       ID        #IMPLIED
   align    CDATA     #IMPLIED -- position: left, right or center --
   cap      CDATA     #IMPLIED -- caption at left, right, top, bottom --
   noflow   (noflow)  #IMPLIED -- disables text flow --
   ismap    (ismap)   #IMPLIED -- server can handle mouse clicks/drags --
   src      %URL;     #IMPLIED -- link to image data --
   index    CDATA     #IMPLIED -- entries for index compilation -->

<!ELEMENT FIGD - - (%table;) -- figure description -->

<!-- Figure anchors designate polygonal areas on the figure which can be
clicked with the mouse. The default area is the whole of the figure. This
mechanism interprets mouse clicks locally, and browsers can choose to
highlight the designated area (or change the mouse sprite) when the mouse
is moved over the area.

Note that polygons may be non-convex or even intersect themselves,
thereby complicating the definition of what is enclosed by the polygon.
Holes are excluded.
-->

<!ELEMENT FIGA - O EMPTY>
```

```
<!ATTLIST FIGA
   href    %URL;     #REQUIRED -- link to traverse when clicked --
   area    NUMBERS   #IMPLIED -- x1,y1,x2,y2,x3,y3,... -->

<!-- FIGT Text on top of an figure background, or in a colored background
box which sits arbitrarily on top of an figure background. The text can
include headers, lists and tables etc. The width attribute allows you to
limit the width of the text box. The height is then determined
automatically by the browser.

FIGT can also be used to position a graphic on top of a picture using an
IMG element within FIGT. In this case the chromakey attribute may allow
parts of the underlying image to show through.

You can make the whole of the box into a hypertext link. This will act as
if it is underneath any hypertext links specified by the overlay markup
itself.
-->

<!ELEMENT FIGT - - (%main;)>

<!ATTLIST FIGT
   at      NUMBERS #IMPLIED -- upper left origin for text --
   width   NUMBER  #IMPLIED -- given as fraction of picture --
   framed  (framed)#IMPLIED -- framed with coloured background --
   href    %URL;   #IMPLIED -- link to traverse when clicked -->

<!-- inline icons/small graphics
The align attribute defines whether the top middle or bottom of the
graphic and current text line should be aligned vertically

The SEETHRU attribute is intended as a chromakey to allow a given colour
to be designated as "transparent". Pixels with this value should not be
painted. The exact format of this attribute's value has yet to be
defined.


Use the FIG tag for captioned figures with active areas etc.
-->

<!ELEMENT IMG - O EMPTY>

<!ATTLIST IMG
   src     %URL;   #REQUIRED -- where to get image data --
   align   CDATA   #IMPLIED  -- top, middle or bottom --
   seethru CDATA   #IMPLIED  -- for transparency --
   ismap   (ismap) #IMPLIED  -- send mouse clicks/drags to server -->

<!-- Hierarchical groups for books, chapters, sections etc. -->

<!ELEMENT GROUP - - ((TITLE|LINK*), (%main;)*)>

<!ATTLIST GROUP
   id    ID        #IMPLIED
   role  CDATA     #IMPLIED -- book, chapter, section etc. --
   inset (inset)   #IMPLIED -- rendering hint: indent margins -->

<!-- change bars defined by a matched pair of CHANGED elements:

     <changed id=z34> changed text <changed idref=z34>

This tag can't act as a container, since changes don't respect
the nesting implied by paragraphs, headers, lists etc.
-->

<!ELEMENT CHANGED - O EMPTY>

<!ATTLIST CHANGED -- one of id and idref is always required --
   id       ID      #IMPLIED -- signals start of changes --
   idref    IDREF   #IMPLIED -- signals end of changes -->
```

```
<!-- Hypertext Links from points within document nodes -->

<!ELEMENT A - - (#PCDATA | IMG | EM | EMBED)*>

<!ATTLIST A
   id       ID       #IMPLIED -- as target of link --
   name     ID       #IMPLIED -- backwards compatibility --
   href     %URL;    #IMPLIED -- destination node --
   role     CDATA    #IMPLIED -- role of link, e.g. "partof" --
   effect   CDATA    #IMPLIED -- replace/new/overlay --
   print    CDATA    #IMPLIED -- reference/footnote/section --
   title    CDATA    #IMPLIED -- when otherwise unavailable --
   type     CDATA    #IMPLIED -- for presentation cues --
   size     NAMES    #IMPLIED -- for progress cues -->

<!-- Other kinds of relationships between documents -->

<!ELEMENT LINK - O EMPTY>

<!ATTLIST LINK
   href     %URL;    #IMPLIED -- destination node --
   role     CDATA    #IMPLIED -- role played, e.g. "toc" -->

<!-- Original document URL for resolving relative URLs  -->

<!ELEMENT BASE - O EMPTY>

<!ATTLIST BASE HREF %URL; #IMPLIED>

<!-- Signifies the document's URL accepts queries -->

<!ELEMENT ISINDEX - O (EMPTY)>

<!ATTLIST ISINDEX href %URL; #IMPLIED -- defaults to document's URL -->

<!-- For use with autonumbering editors - don't reuse ids, allocate next
one starting from this one -->

<!ELEMENT NEXTID - O (EMPTY)>

<!ATTLIST NEXTID N NAME #REQUIRED>

<!-- Mnemonic character entities. -->

<!ENTITY AElig "&#198;" -- capital AE diphthong (ligature) -->
<!ENTITY Aacute "&#193;" -- capital A, acute accent -->
<!ENTITY Acirc "&#194;" -- capital A, circumflex accent -->
<!ENTITY Agrave "&#192;" -- capital A, grave accent -->
<!ENTITY Aring "&#197;" -- capital A, ring -->
<!ENTITY Atilde "&#195;" -- capital A, tilde -->
<!ENTITY Auml "&#196;" -- capital A, dieresis or umlaut mark -->
<!ENTITY Ccedil "&#199;" -- capital C, cedilla -->
<!ENTITY ETH "&#208;" -- capital Eth, Icelandic -->
<!ENTITY Eacute "&#201;" -- capital E, acute accent -->
<!ENTITY Ecirc "&#202;" -- capital E, circumflex accent -->
<!ENTITY Egrave "&#200;" -- capital E, grave accent -->
<!ENTITY Euml "&#203;" -- capital E, dieresis or umlaut mark -->
<!ENTITY Iacute "&#205;" -- capital I, acute accent -->
<!ENTITY Icirc "&#206;" -- capital I, circumflex accent -->
<!ENTITY Igrave "&#204;" -- capital I, grave accent -->
<!ENTITY Iuml "&#207;" -- capital I, dieresis or umlaut mark -->
<!ENTITY Ntilde "&#209;" -- capital N, tilde -->
<!ENTITY Oacute "&#211;" -- capital O, acute accent -->
<!ENTITY Ocirc "&#212;" -- capital O, circumflex accent -->
<!ENTITY Ograve "&#210;" -- capital O, grave accent -->
<!ENTITY Oslash "&#216;" -- capital O, slash -->
<!ENTITY Otilde "&#213;" -- capital O, tilde -->
<!ENTITY Ouml "&#214;" -- capital O, dieresis or umlaut mark -->
<!ENTITY THORN "&#222;" -- capital THORN, Icelandic -->
<!ENTITY Uacute "&#218;" -- capital U, acute accent -->
```

```
<!ENTITY Ucirc "&#219;" -- capital U, circumflex accent -->
<!ENTITY Ugrave "&#217;" -- capital U, grave accent -->
<!ENTITY Uuml "&#220;" -- capital U, dieresis or umlaut mark -->
<!ENTITY Yacute "&#221;" -- capital Y, acute accent -->
<!ENTITY aacute "&#225;" -- small a, acute accent -->
<!ENTITY acirc "&#226;" -- small a, circumflex accent -->
<!ENTITY aelig "&#230;" -- small ae diphthong (ligature) -->
<!ENTITY agrave "&#224;" -- small a, grave accent -->
<!ENTITY amp "&amp;" -- ampersand -->
<!ENTITY aring "&#229;" -- small a, ring -->
<!ENTITY atilde "&#227;" -- small a, tilde -->
<!ENTITY auml "&#228;" -- small a, dieresis or umlaut mark -->
<!ENTITY ccedil "&#231;" -- small c, cedilla -->
<!ENTITY eacute "&#233;" -- small e, acute accent -->
<!ENTITY ecirc "&#234;" -- small e, circumflex accent -->
<!ENTITY egrave "&#232;" -- small e, grave accent -->
<!ENTITY eth "&#240;" -- small eth, Icelandic -->
<!ENTITY euml "&#235;" -- small e, dieresis or umlaut mark -->
<!ENTITY gt "&#62;" -- greater than -->
<!ENTITY iacute "&#237;" -- small i, acute accent -->
<!ENTITY icirc "&#238;" -- small i, circumflex accent -->
<!ENTITY igrave "&#236;" -- small i, grave accent -->
<!ENTITY iuml "&#239;" -- small i, dieresis or umlaut mark -->
<!ENTITY lt "&lt;" -- less than -->
<!ENTITY ntilde "&#241;" -- small n, tilde -->
<!ENTITY oacute "&#243;" -- small o, acute accent -->
<!ENTITY ocirc "&#244;" -- small o, circumflex accent -->
<!ENTITY ograve "&#242;" -- small o, grave accent -->
<!ENTITY oslash "&#248;" -- small o, slash -->
<!ENTITY otilde "&#245;" -- small o, tilde -->
<!ENTITY ouml "&#246;" -- small o, dieresis or umlaut mark -->
<!ENTITY szlig "&#223;" -- small sharp s, German (sz ligature) -->
<!ENTITY thorn "&#254;" -- small thorn, Icelandic -->
<!ENTITY uacute "&#250;" -- small u, acute accent -->
<!ENTITY ucirc "&#251;" -- small u, circumflex accent -->
<!ENTITY ugrave "&#249;" -- small u, grave accent -->
<!ENTITY uuml "&#252;" -- small u, dieresis or umlaut mark -->
<!ENTITY yacute "&#253;" -- small y, acute accent -->
<!ENTITY yuml "&#255;" -- small y, dieresis or umlaut mark -->

<!-- dash entities -->

<!ENTITY endash "--"  -- En dash -->
<!ENTITY emdash "---" -- Em dash -->

<!-- The END -->

]>
```

Appendix II - Entity Definitions

ISO Latin 1 character entities in HTML+ derived from "ISO 8879:1986//ENTITIES Added Latin 1//EN". The corresponding 8-bit character codes are given in the DTD.

Æ	capital AE diphthong (ligature)
Á	capital A, acute accent
Â	capital A, circumflex accent
À	capital A, grave accent
Å	capital A, ring
Ã	capital A, tilde
Ä	capital A, dieresis or umlaut mark
Ç	capital C, cedilla
Ð	capital Eth, Icelandic
É	capital E, acute accent
Ê	capital E, circumflex accent
È	capital E, grave accent
Ë	capital E, dieresis or umlaut mark
Í	capital I, acute accent
Î	capital I, circumflex accent
Ì	capital I, grave accent
Ï	capital I, dieresis or umlaut mark
Ñ	capital N, tilde
Ó	capital O, acute accent
Ô	capital O, circumflex accent
Ò	capital O, grave accent
Ø	capital O, slash
Õ	capital O, tilde
Ö	capital O, dieresis or umlaut mark
Þ	capital THORN, Icelandic
Ú	capital U, acute accent
Û	capital U, circumflex accent
Ù	capital U, grave accent
Ü	capital U, dieresis or umlaut mark
Ý	capital Y, acute accent
á	small a, acute accent
â	small a, circumflex accent
æ	small ae diphthong (ligature)
à	small a, grave accent
å	small a, ring
ã	small a, tilde
ä	small a, dieresis or umlaut mark
ç	small c, cedilla
é	small e, acute accent
ê	small e, circumflex accent
è	small e, grave accent
ð	small eth, Icelandic
ë	small e, dieresis or umlaut mark

í	small i, acute accent
î	small i, circumflex accent
ì	small i, grave accent
ï	small i, dieresis or umlaut mark
ñ	small n, tilde
ó	small o, acute accent
ô	small o, circumflex accent
ò	small o, grave accent
ø	small o, slash
õ	small o, tilde
ö	small o, dieresis or umlaut mark
ß	small sharp s, German (sz ligature)
þ	small thorn, Icelandic
ú	small u, acute accent
û	small u, circumflex accent
ù	small u, grave accent
ü	small u, dieresis or umlaut mark
ý	small y, acute accent
ÿ	small y, dieresis or umlaut mark

In addition, there are two entity definitions for horizontal dashes longer than the "-" character.

&endash;	En sized horizontal dash (--)
&emdash;	Em sized horizontal dash (---)

Appendex III - Compatibility with HTML

HTML documents can be easily converted into the HTML+ format, and only a few changes are needed. Most documents won't need any changes at all. HTML+ browsers should be able to view HTML documents with very little effort. Older browsers will be able to view HTML+ documents which don't contain, tables or forms.

Lists

<menu>	becomes	<ul compact>
<dir>	becomes	<ul narrow>

Emphasis

HTML+ replaces the various tags used by HTML with a single tag. It may be worth changing the name for the emphasis tag in HTML+ from EM to EM, to gain compatibility with this common form. However, using EM might be confused with the typographical term *em* as in em dash (you also get en dash). EM has the merit of being unambigous. **I would like to get peoples views on this.**

	becomes	
<tt>	becomes	<em tt>
	becomes	<em b>
<b>	becomes	<em b>
<i>	becomes	<em i>
<u>	becomes	<em u>
<code>	becomes	
<samp>	becomes	
<kbd>	becomes	
<var>	becomes	
<dfn>	becomes	
<cite>	becomes	

Miscellaneous

Some tags which are deprecated in HTML are now obsolete, and should be mapped to preformatted text:

<plaintext>	becomes	<pre>
<xmp>	becomes	<pre>
<listing>	becomes	<pre>

The following two tags have been absorbed into the standard mechanism for paragraphs:

<address>	becomes	<p role="byline" align="right">
<blockquote>	becomes	<p role="quote">

References

This is missing the appropriate references to work on the syntax and name service for URNs. The HTTP definition needs updating to cover the encoding of form data (and *ismap* ?).

[Berners-Lee 93a] *"Hypertext Markup Language (HTML)"*, Tim Berners-Lee, March 1993.
URL=`ftp://info.cern.ch/pub/www/doc/http-spec.ps`

[Berners-Lee 93b] *"Uniform Resource Locators"*, Tim Berners-Lee, January 1992.
URL=`ftp://info.cern.ch/pub/ietf/url4.ps`

[Berners-Lee 93c] *"Protocol for the Retrieval and Manipulation of Textual and Hypermedia Information"*, Tim Berners-Lee, 1993.
URL=`ftp://info.cern.ch/pub/www/doc/html-spec.ps`

[Raisch 93] *"Style sheets for HTML"*, Robert Raisch, June 1993, O'Reilly & Associates
email: `raisch.ora.com`

[Kimber 93] Article in comp.text.sgml newsgroup, 24th May 1993 by Elliot Kimber (drmacro@vnet.almaden.ibm.com),
URL=`news:19930524.152345.29@almaden.ibm.com`

A1

perh2

Knowbots, Permissions Headers and Contract Law

paper for the conference on

Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment

April 2-3, 1993 with revisions of 4/30/93

Henry H. Perritt, Jr.
Professor of Law
Villanova Law School
Villanova, PA 19085
(215) 645-7078
FAX (215) 645-7033, (215) 896-1723
Internet: perritt@ucis.vill.edu

Introduction

One of the ways to protect intellectual property on the NREN is through a digital library concept. Under this concept, a work would have attached to it a "permissions header," defining the terms under which the copyright owner makes the work available. The digital library infrastructure, implemented on the NREN, would match request messages from users with the permissions headers. If the request message and the permissions header match, the user would obtain access to the work. This concept encompasses major aspects of electronic contracting, which is already in wide use employing Electronic Data Interchange ("EDI") standards developed by ANSI Committee X12.1

This paper explains the relationship between the digital library concept and EDI practice, synthesizing appropriate solutions for contract law, evidence, and agency issues that arise in electronic contracting. The question of how electronic signatures should work to be legally effective is an important part of this inquiry. The paper also defines particular types of service identifiers, header descriptors, and other forms of labeling and tagging appropriate to allow copyright owners to give different levels of permission, including outright transfer of the copyright interest, use permission, copying permission, distribution permission, display permission, and permission to prepare derivative works.The paper considers how payment authorization procedures should work in conjunction with a permissions header and digital library concept in order to integrate the proposed copyright licensing procedures with existing and anticipated electronic payment authorization systems. The paper necessarily considers whether existing standards approaches related to SGML and X12 are sufficient or whether some new standards development efforts will be necessary for implementation of the concepts. The paper considers the relationship between technology and law in enforcing intellectual property, and emphasizes that the traditional adaptation of legal requirements to levels of risk is appropriate as the law is applied to new technologies.

There are certain common issues between the intellectual property question and other applications of wide area

digital network technology. The question of signatures and writings to reflect the establishment of duties and permissions and the transfer of rights is common to the intellectual property inquiry and to electronic commerce using EDI techniques. There also are common questions involving rights to use certain information channels: First Amendment privileges, and tort liability. These are common not only to technological means of protecting intellectual property but to all forms of wide area networking.

The problem

The law recognizes intellectual property because information technology permits one person to get a free ride on another person's investment in creating information value. Creative activity involving information usually is addressed by copyright, although patent has a role to play in protecting innovative means of processing information.2

Intellectual property arose in the context of letterpress printing technology. Newer technologies like xerography and more recently small computer technology and associated word processing and networking have increased the potential for free rides and accordingly increased the pressure on intellectual property.

The concern about free ride potential is especially great when people envision putting creative works on electronic publishing servers connected to wide area networks intending to permit consumers of information products to access these objects, frequently combining them and generally facilitating "publishing on demand" rather than the well known publishing just in case, typified by guessing how many copies of a work will sell, printing those in advance, and then putting them in inventory until someone wants them.

The concern is that it will be too easy to copy an entire work without detection and without paying for it. Worse, it will be easy to copy an entire work and resell it either by itself or as a part of a new derivative work or collection.

But technology is capable of protecting investment in new ways as well as gaining a free ride. Computer networks make it possible to restrict access and to determine when access occurs. Depending on how new networks are designed, they may actually reduce the potential for a free ride. The digital library is one way of realizing that potential. Professor Pamela Samuelson has observed that the digital library model replaces intellectual property with a system of technological controls.3

Digital Library Concepts

Basic Concepts

A digital library is a set of information resources ("information objects") distributed throughout an electronic network. The objects reside on servers (computers with associated disk drives connected to the network). They can be retrieved remotely by users using "client" workstations.

Origin of Concepts

The phrase "digital library" and the basic concept was first articulated in a 1989 report growing out of a workshop sponsored by the Corporation for National Research Initiatives.4 From its inception, the digital library concept envisioned retrieval of complete information

resources and not merely bibliographic information.5

The technologies of remote retrieval of complete information objects using electronic technologies is in wide use through the WESTLAW, Dialog, LEXIS, NEXIS, and National Library of Medicine databases. These remotely accessible databases, however, unlike the digital library involved a single host on which most of the data resides. The digital library concept envisions a multiplicity of hosts (servers).

Recent Developments

The remotely accessible database host concept is converging with the digital library concept as more of the electronic database vendors provide gateways to information objects actually residing on other computers. This now is commonplace with WESTLAW access to Dialog, and Dialog's gateways to other information providers.

The most explicit implementation of the digital library concept is the Wide Area Information Service ("WAIS"), which implements ANSI standard Z.39.6 WAIS permits a remote user to formulate a query that is applied to a multiplicity of WAIS servers each of which may contain information responsive to the query. The WAIS architecture permits search engines of varying degrees of sophistication, resident on WAIS information servers to apply the query against their own information objects, reporting matches back to the user.7 Future implementations of WAIS permit automatic refinement of searches according to statistical matching techniques.

The Corporation for National Research Initiatives has proposed a test bed for an electronic copyright management system.8 The proposed system would include four major elements: automated copyright recording and registration, automated, on line clearance of rights, private electronic mail and digital signatures to provide security. It would include three subsystems: a Registration and Recording System (RRS), a Digital Library System (DLS), and a Rights Management System (RMS). The RRS would provide the functions enumerated above and would be operated by the Library of Congress. It would provide "change of title" information.9 The RMS would be an interactive distributed system capable of granting rights on line and permitting the use of copyrighted material in the Digital Library System. The test bed architecture would involve computers connected to the Internet performing the RRS and RMS functions.

Digital signatures would link an electronic bibliographic record with the contents of the work, ensuring against alteration after deposit.10 Multiple RMS servers would be attached to the Internet. A user wishing to obtain rights to an electronically published work would interact electronically with the appropriate RMS. When copyright ownership is transferred, a message could be sent from the RMS to the RRS11 - creating an electronic marketplace for copyrighted material.

The EBR submitted with a new work would "identify the rights holder and any terms and conditions on the use of the document or a pointer to a designated contact for rights and permissions."12 The EBR, thus, is apparently equivalent to the permissions header discussed in this paper. Security in the transfer of rights would be provided by digital signatures using public key encryption, discussed further, infra in the section on encryption.

Basic Architectural Concepts

The digital library concept in general contemplates three basic architectural elements: a query, also called a "knowbot" in some descriptions; a permissions header attached to each information object; and a procedure for matching the query with the permissions header.

Two kinds of information are involved in all three architectural elements: information about the content of information objects desired and existing, and information about the economic terms on which an information object is made available. For example, a query desiring court opinions involving the enforcement of foreign judgments evidencing a desire to download the full text of such judicial opinions and to pay up to $1.00 per minute of search and downloading time would require that the knowbot appropriately represent the subject matter "enforcement of foreign judgments." It also requires that the knowbot appropriately represent the terms on which the user is willing to deal: downloading and the maximum price. The permissions header similarly must express the same two kinds of information. If the information object to which the permissions header is attached is a short story rather than a judicial opinion, the permissions header must so indicate. Or, if the information object is a judicial opinion and it is about enforcement of foreign judgments, the permission header may indicate that only a summary is available for downloading at a price of $10.00 per minute. The searching, matching, and retrieval procedure in the digital library system must be capable of determining whether there is a match on both subject matter and economic terms, also copying and transmitting the information object if there is a match.

Comparison to EDI

Electronic Data Interchange ("EDI") is a practice involving computer-to-computer commercial dealing without human intervention. In the most widespread implementations, computers are programmed to issue purchase orders to trading partners, and the receiving computer is programmed to evaluate the terms of the purchase order and to take appropriate action, either accepting it and causing goods to be manufactured or shipped or rejecting it and sending an appropriate message. EDI is in wide use in American and foreign commerce, using industry-specific standards for discrete commercial documents like purchase orders, invoices, and payment orders, developed through the American National Standards Institute.

There obviously are similarities between the three architectural elements of the digital library concept and EDI. There is a structured way of expressing an offer or instruction, and a process for determining whether there is a match between what the recipient is willing to do and what the sender requests.

There is also, however, an important difference. In the digital library concept, a match results in actual delivery of the desired goods and services in electronic form. In EDI practice, the performance of the contractual arrangement usually involves physical goods or performance of nonelectronic services.

Nevertheless, the digital library and EDI architectures are sufficiently similar and, it turns out the legal issues associated with both are sufficiently similar to make analogies appropriate.

Elements of Data Structure

For purposes of this paper, the interesting parts of the data structure are those elements that pertain to permission, more than those elements that pertain to content of the information object to which the header is attached. Accordingly, this section will focus on only permissions-related elements, after noting in passing that the content part of the header well might be a pointer to an inverted file to permit full text searching and matching.

The starting point conceptually for identifying the elements of the permissions header are the rights exclusively reserved to the copyright owner by 106 of the copyright statute. But these exclusive rights need not be tracked directly because the owner of an information object free to impose contractual restrictions as well as to enjoy rights granted by the Copyright Act. Accordingly, it seems that the following kinds of privileges in the requester should be addressed in the permissions header:

outright transfer of all rights

use privilege, either unrestricted or subject to restrictions

copying, either unlimited or subject to restrictions like quantitative limits

distribution, either unlimited or subject to restrictions, like geographic ones or limits on the markets to which distribution can occur

preparation of derivative works

Display and presentation rights, separately identified in 106 would be subsumed into the use element, because they are particular uses.

The simplest implementation would allow only binary values for each of these elements. But a binary approach does not permit the permissions header to express restrictions, like those suggested in the enumerated list. Elements could be defined to accept the most common kinds of restrictions on use, and quantitative limits on copying, but it would be much more difficult to define in advance the kinds of geographic or market-definition restrictions that an owner might wish to impose with respect to distribution.

In addition to these discrete privileges, the permissions header must express pricing information. The most sensible way of doing this is to have a price associated with each type of privilege. In the event that different levels of use, copying, or distribution privilege are identified, the data structure should allow a price to be associated with each level.

A complicating factor in defining elements for price is the likelihood that different suppliers would want to price differently. For example, some would prefer to impose a flat fee for the grant of a particular privilege. Others might wish to impose a volume-based fee, and still others might wish to impose a usage or connect-time based fee. The data structure for pricing terms must be flexible enough to accommodate at least these three different approaches to pricing.

Finally, the data structure must allow for a specification of acceptable payment terms and have some kind of trigger for a payment approval procedure. For example,

the permissions header might require presentation of a credit card number and then trigger a process that would communicate with the appropriate credit card database to obtain authorization. Only if the authorization was obtained would the knowbot and the permissions header "match."

There is a relationship between the data structures and legal concepts. The knowbot is a solicitation of offers. The permissions header is an offer. The matching of the two constitutes an acceptance. Mr. Linn's "envelope" could be the "contract."

There are certain aspects of the data structure design that are not obvious. One is how to link price with specific levels of permission. Another is how to describe particular levels of permission. This representation problem may benefit from the use of some deontic logic, possibly in the form of a grammar developed for intellectual property permissions. Finally, it is not clear what the acceptance should look like. Conceptually, the acceptance occurs when the knowbot matches with a permissions header, but it is unclear how this legally significant event should be represented.

Role of Encryption

The CNRI test bed proposal envisions the use of public key encryption to ensure the integrity of digital signatures and to ensure the authenticity of information objects. Public key encryption permits a person to encrypt a message - like a signature using a secret key, one known only to the sender, while permitting anyone with access to a public key to decrypt it. Use of public key cryptography in this fashion permits any user to authenticate a message, ensuring that it came from the purported sender.13 A related technology called "hashing" permits an encrypted digital signature to be linked to the content of a message. The message can be sent in plain text (unencrypted) form, but if any part of it is changed, it will not match the digital signature. The digital signature and hashing technologies thus permit not only the origin but also the content integrity of a message of arbitrary length to be authenticated without necessitating encryption of the content of the message. This technology has the advantage, among others, that it is usable by someone lacking technological access to public key encryption. An unsophisticated user not wishing to incur the costs of signature verification nevertheless can use the content of the signed information object.

It is well recognized that encryption provides higher levels of security than other approaches. But security through encryption comes at a price. Private key encryption systems require preestablished relationships and exchange of private keys in advance of any encrypted communication. The burdens of this approach have led most proponents of electronic commerce to explore public key encryption instead. But public key systems require the establishment and policing of a new set of institutions. An important infrastructure requirement for practicable public key cryptography is the establishment and maintenance of certifying entities that maintain the public keys and ensure that they are genuine ones rather than bogus ones inserted by forgers. A rough analogy can be drawn between the public key certifying entities and notaries public. Both kinds of institutions verify the authenticity of signature. Both kinds require some level of licensing by governmental entities. Otherwise the word of the "electronic notary"

(certifying entity) is no better than an uncertified, unencrypted signature. In a political and legal environment in which the limitations of regulatory programs have been recognized and have led to deregulation of major industries, it is not clear that a major new regulatory arrangement for public key encryption is practicable. Nevertheless, experimentation with the concept in support of digital library demonstration programs can help generate more empirical data as to the cost and benefits of public key encryption to reinforce electronic signatures.

On the other hand, it is not desirable to pursue approaches requiring encryption of content. No need to encrypt the contents is apparent in a network environment. Database access controls are sufficient to prevent access to the content if the permissions header terms are not matched by the knowbot. On the other hand, if the electronic publishing is effected through CDROMs or other physical media possessed by a user, then encryption might be appropriate to prevent the user from avoiding the permissions header and going directly to the content.

While encrypted content affords greater security to the owner of copyrighted material. Someone who has not paid the price to the copyright owner must incur much higher cost to steal the material. But the problem is everyone must pay a higher price to use the material. One of the dramatic lessons of the desktop computer revolution was the clear rejection of copyright protection in personal computer software. The reasons that copy protection did not survive in the market place militate against embracing encryption for content. Encryption interferes with realization of electronic markets, because producer and consumer must have the same encryption and description protocols. Encryption burdens processing of electronic information objects because it adds another layer. Some specific implementations have encryption require additional hardware at appreciable costs.

Digital libraries cannot become a reality until consumers perceive that the benefits of electronic formats outweigh the costs, compared to paper formats. Encryption interferes with electronic formats' traditional advantages of density, reusability, editability, and computer search ability and also, by impairing open architectures may perpetuate some of papers' advantages with respect with browsibility.14

The need for encryption of any kind depends upon whether security is available without it. That depends, in turn, on the kinds of free rides that may be obtainable and the legal status of various kinds of electronics transactions in the digital library system.

Legal Issues

Copyright: What legal effect is intended?

The design of the permissions header and the values in the elements of the header must be unambiguous as to whether an outright transfer of a copyright interest is intended or whether only a license is intended. If an outright transfer15 is intended, then the present copyright statute requires a writing signed by the owner of the rights conveyed.16 Recordation of the transfer with the Copyright office is not required, but provides advantages in enforcing transferee rights.17 On the other hand, non exclusive licenses need not be in writing nor registered. If the electronic transaction transfers the copyright in its entirety, then the rights of the transferor are extinguished, and the rights of the

transferee are determined by the copyright statute. The only significant legal question is whether the conveyance was effective.

On the other hand, when the copyright is not transferred outright but only certain permissions are granted or certain rights conveyed, the legal questions become more varied. Then, the rights of the transferor and the obligations of the transferee are matters of contract law. It is important to understand the degree to which the contract is enforceable and how it is to be interpreted in the event of subsequent disputes. The following sections consider briefly the first sale doctrine as a potential public policy obstacle to enforcing contractual restrictions different from those imposed by the copyright statute and then explore in greater depth whether electronic techniques satisfy the formalities traditionally required for making a contract, whether they adequately ensure against repudiation, and whether they provide sufficient information to permit predictable interpretation of contractual obligations and privileges.

First Sale Doctrine

The first sale doctrine may invalidate restrictions on use. It is impermissible for the holder of a patent to impose restrictions on the use of a patented product after the product has been sold. Restrictions may be imposed, however, on persons who merely license the product.18 The rationale for this limit on the power of the owner of the intellectual property interest is that to allow limitations on use of the product would interfere with competition beyond what the Congress - and arguably the drafters of the Constitution - intended in setting up the patent system.

The first sale doctrine applies to copyright owners.19 Indeed, because of the First Amendment's protection of informational activity, the argument against restrictions after the first sale may be even stronger in the copyright arena then in the patent arena.

The first sale doctrine is potentially important because it may invalidate restrictions imposed on the use of information beyond what is authorized by the Copyright Act and by common law trade secret. Thus, there may be serious questions about the legal efficacy of use restrictions suggested in ___, although such restrictions are common in remote database service agreements. The vendors could argue that the limitations pertain to the contractual terms for delivery of a service rather than use of information as such. The characterization avoids the overlap with copyright and thus may also avoid the conflict between federal policy and contract enforcement.20

Contract Formation Issues

The law does not enforce every promise. Instead, it focuses its power only on promises surrounded with certain formalities to make it likely that the person making the promise (the "promisor") and the person receiving the promise (the "promisee") understood that their communication had legal consequences. A threshold question for the digital library system is whether the traditional formalities for making a contract are present when the contract is made through electronic means. The digital library system considered in this paper clearly contemplates that a contract is formed when the knowbot and the permissions header achieve a match. In this respect, the digital library concept converges with EDI where trading parties contemplate that a contract to perform services or deliver goods is

formed when a match occurs either upon the receipt of a purchase order or upon the transmission of a purchase order acknowledgment.

It is not altogether clear, however, whether the match between values and computer data structures meets contract formation requirements, particularly those expressed in various statutes of frauds. Statutes of frauds require "writings" and "signatures" for certain kinds of contracts - basically those contemplating performance extending beyond a period of one year.21

In many instances, the digital library contract will be fully performed almost instantaneously upon delivery of the information object after the knowbot and the permissions header match. In such a case, the statute of frauds is not a problem and its requirements need not be satisfied. In other cases, however, as when the intent of the owner of the information object is to grant a license to do things that will extend beyond one year, the statute of frauds writing and signature requirements must be met.

Historical application of Statutes Of Frauds by the courts clearly indicates that there is flexibility in the meaning of "writing" and "signature." A signature is any mark made with the intent that it be a signature.22 Thus an illiterate person signs by making an "X," and the signature is legally effective. Another person may sign a document by using a signature stamp. Someone else may authorize an agent to sign his name or to use the signature stamp. In all three cases the signature is legally effective. There may of course be arguments about who made the X, or whether the person applying the signature stamp was the signer or his authorized agent, but these are evidentiary and agency questions, not arguments about hard and fast contract-law requirements.

Under the generally accepted legal definition of a signature, there is no legal reason why the "mark" may not be made by a computer printer, or for that matter by the write head on a computer disk drive or the data bus in a computer random access memory. The authorization to the computer agent to make the mark may be given by entering a PIN ("Personal Identification Number") on a keyboard. To extend the logic, there is no conceptual reason to doubt the legal efficacy of authority to make a mark if the signer writes a computer program authorizing the application of a PIN upon the existence of certain conditions that can be tested by the program. The resulting authority is analogous to a signature pen that can be operated only with a mechanical key attached to somebody's key ring, coupled with instructions to the possessor of the key.

Which of these various methods should be selected for particular types of transactions must depend, not on what the law requires, because the law permits any of these methods. Rather, it must depend on the underlying purposes of the legal requirement and which method best serves those purposes.

The real issue is how to prove that a particular party made the mark. In other words, the contingency to be concerned about is repudiation, not absence of formalities. Repudiation should be dealt with through usual evidentiary and fact finding processes rather than artificial distinctions between signed and unsigned documents.

Authority is skimpier on how flexible the "writing"

requirement is. The best approach is to borrow the fixation idea from the copyright statute and conclude that a writing is "embodiment in a copy . . . sufficiently permanent or stable to permit it to be perceived, reproduced, or otherwise communicated for more a period of more than transitory duration."23

The most important thing conceptually is to understand the purpose of the writing and signature requirements. They have two purposes: awareness or formality and reliability of evidence. Signature requirements, like requirements for writings and for original documents have an essentially evidentiary purpose. If there is a dispute later, they specify what kind of evidence is probative of certain disputed issues, like "who made this statement and for what purpose?" The legal requirements set a threshold of probativeness. Surely the values in a knowbot as well as the values in a permissions header constitute and "mark," and someone who knowingly sets up potential transactions in a digital library scheme can have the intent that the mark be a signature.

When a contract is made through a signed writing, it is more likely that the parties to the contract understand what they are doing. They are aware of the legal affect of their conduct because the writing in the signature involve a greater degree of formality than a simple conversation.

The awareness/formality purpose can be served by computerized contracting systems. This is so not so much because the computers are "aware" of the affect of their "conduct." Rather, it is true because the computers are agents of human principals. The programming of the computer to accept certain contract terms is the granting of authority to the computer agent to enter into a contract. The fact that a principal acts through an agent engaging in conduct at a later point and time never has been thought to defeat contract formation in the traditional evolution of agency and contract law. Nor should it when the agent is a computer.

Fulfillment of the evidentiary purpose depends on the reliability of the information retained by the computer systems making up the digital library. Such systems must be designed to permit the proponent of contract formation to establish the following propositions if the other party to the purported contract attempts to repudiate it.

1. It came from computer X

2. It accurately represents what is in computer X24 now25

3. What is in computer X now is what was in computer X at the time of the transaction

4. What was in computer X at the time of the transaction is what was received from the telecommunications channel26

5. What was received from the telecommunications channel is what was (a) sent, (b) by computer Y.

Two other questions relate to matters other than the authenticity of the message:

6 Computer Y was the agent of B

7 The message content expresses the content of the

contract (or more narrowly, the offer or the acceptance).27

Factual propositions 1-4 can be established by testimony as to how information is written to and from telecommunications channel processors, primary storage, and secondary storage. Factual proposition 5 requires testimony as to the accuracy of the telecommunications channel and characteristics of the message that associate it with computer Y. Only the last proposition (number 5) relates to signatures, because signature requirements associate the message with its source.28 The other propositions necessitate testimony as to how the basic message and database management system works. It is instructive to compare these propositions with the kinds of propositions that must be established under the business records exception to the hearsay rule when it is applied to computer information.

Those propositions may be supported with non technical evidence, presented by non programmers. A witness can lay a foundation for admission of computer records simply by testifying that the records are generated automatically and routinely in the ordinary course of business. The more inflexible the routine, and the less human intervention in the details of the computer's management of the database the better the evidence.29

The ultimate question is trustworthiness, and if the computer methods are apparently reliable, the information should be admitted unless the opponent of admissibility can raise some reasonable factual question undercutting trustworthiness.30

Contract Interpretation Issues

Assuming that the permissions header and knowbot constitute sufficient writings to permit a contract to be formed and that the signature requirement also is met, through digital signature technology or otherwise, there still are difficult contract interpretation questions. Contract interpretation questions arise not only after contractual relationships are formed, but also in connection with deciding whether there has been offer and acceptance, the prerequisites to contract formation.31 Contract interpretation always seeks to draw inferences about what the parties intended. When contract interpretation issues arise at the contract formation stage, the questions are what the offeror intended the content of the offer to be and what the offeree intended the content of the purported acceptance to be. The proposed Digital Library System envisions extremely cryptic expressions of offer and acceptance - by means of codes. The codes have no intrinsic meaning. Rather, extrinsic reference must be made to some kind of table, standard, or convention associating particular codes with the concepts they represent. Extrinsic evidence is available to resolve contract interpretation questions when the language of the contract itself is ambiguous, and perhaps at other times as well.32 The codes in the permissions header and knowbots certainly are ambiguous and become unambiguous only when extrinsic evidence is considered. So there is no problem in getting a standard or cable into evidence. The problem is whether the parties meant to assent to this standard.

In current EDI practice, this question is resolved by having parties who expect to have EDI transactions with each other to sign a paper trading partner agreement, in which the meaning of values or codes in the transaction sets is established.33 But requiring each pair of suppliers and users

of information in a digital library to have written contracts with each other in advance would defeat much of the utility of the digital library. Thus the challenge is to establish some ground rules for the meaning of permissions header and knowbot values that all participants are bound by. There are analogous situations. One is a standard credit card agreement that establishes contractual terms among credit card issuer, credit card subscriber, and merchant who accepts the credit card. The intermediary - the credit card company - unilaterally establishes contract terms to which the trading partners assent by using and accepting the credit card.34 Also, it is widely recognized that members of a private association can, through their constitution and bylaws establish contractual relationships that bind all of the members in dealing with each other.35 In the Digital Library System, similar legal arrangements can establish the standards by which electronic transactions between permissions header and knowbots will bind transferor and transferee of information.

Third Party Liability

It is not enough merely to ensure that the licensee is contractually bound. Trading partners also must ensure that the participants in funds transfers have enforceable obligations. For example, if the digital library system envisions that the information object would not be released to the purchaser without simultaneous release of a payment order, the supplier may be interested in enforcing the obligations of financial intermediaries who handle the payment order. This implicates the federal Electronic Funds Transfer Act, and Article 4A of the Uniform Commercial Code, regulating wire transfers.

Solutions

Satisfy the Business Records Exception to the Hearsay Rule

The discussion of contract formalities earlier in this paper concluded that legally enforceable contracts can be formed through electronic means and that the significant legal questions relate to reliability of proof and intent of the parties to be bound by using the electronic techniques. This section considers the reliability of proof further. Traditional evidence law permits computer records to be introduced in evidence when they satisfy the requirements of the business records exception: basically that they are made in the ordinary course of business, that they are relied on for the performance of regular business activities, and that there is no independent reason for questioning their reliability.36

The business records exception shares with the authentication concept statute of frauds and the parol evidence rule a common concern with reliability.37 The same procedural guarantees and established practices that ensure reliability for hearsay purposes also ensure reliability for the other purposes. Under the business records exception, the proponent must identify the source of a record, through testimony by one familiar with a signature on the record, or circumstantially.38 The steps in qualifying a business record under the common law, which since have been relaxed,39 were:

Proving that the record is an original entry made in the routine course of business

Proving that the entries were made upon the personal knowledge of the proponent/witness or someone reporting to him

Proving that the entries were made at or near the time of the transaction

Proving that the recorder and his informant are unavailable.40

These specific requirements are easier to understand and to adapt to electronic permissions and obligations formed in a digital library system by understanding the rationale for the business records exception. The hearsay rule excludes out of court statements because they are inherently unreliable, primarily because the maker of the statement's demeanor cannot be observed by the jury and because the maker of the statement is not subject to cross examine. On the other hand, there are some out of court statements that have other guarantees of reliability. Business records are one example. If a continuing enterprise finds the records sufficiently reliable to use them in the ordinary course of business, they should be reliable enough for a court. The criteria for the business records exception all aim at ensuring that the records really are relied upon the business to conduct its ordinary affairs.

The Manual for Multidistrict Litigation suggests steps for qualifying computer information under the business records exception:

1.The document is a business record

2.The document has probative value

3.The computer equipment used is reliable

4.Reliable data processing techniques were used41

The key in adapting the business records exception to electronic permissions in a digital library system are points 3 and 4. Establishing these propositions and the propositions set forth in section ___ of this paper requires expert testimony. Any designer of a digital library system must consult with counsel and understand what testimony an expert would give to establish these propositions. Going through that exercise will influence system design.

Reinforce the Evidentiary Reliability by Using Trusted Third Parties

The evidentiary purpose of contract formation requirements can be satisfied by using a trusted third party as an intermediary, when the third party maintains archival records of the transactions. The third party lacks any incentive for tampering with the records and when the third parties archiving system is properly designed, it can provide evidence sufficient to establish all of the propositions identified in ___.

This third party intermediary concept is somewhat different from the concept for a certifying agent in digital signature systems. To be sure, the custodian of transaction records envisioned by this section could be the same as the certifying entity for public and key encryption, but the custodian role can be played in the absence of any encryption. Indeed, the digital library itself is a good candidate for the custodian role. The library has no incentive to manipulate its records in favor of either of the producers of information value or the consumers. In order to carry out its affairs, it must use these transactional records in the ordinary course of business,

thereby making it likely that digital library records would qualify under the business records exception.

Standardization

Obviously, the digital library concept depends upon the possibility of an automated comparison between the knowbot and the permissions header. This means that potential requesters of information and suppliers of information must know in advance the data structures for representing the elements of the permissions header and the knowbot. This requires compatibility. Compatibility requires standardization. Standardization does not, however, necessarily require "Standard" in the sense that they are developed by some bureaucratic body like ANSI. It may simply imply market acceptance of a particular vendor's approach. Indeed, each digital library might use different data structures. All that is necessary is that the structure of the knowbot and the structure of the permissions header be compatible within any one digital library system. Also, as demands emerge for separate digital libraries to communicate with each other, there can be proprietary translation to assure compatibility between systems much as common word processing programs translate to and from other common formats and much as printers and word processing software communicate with each other through appropriate printer drivers. In neither of these cases has any independent standards organization developed a standard that is at all relevant in the marketplace.

Standardizing the elements of Knowbot and permissions headers involves content standardization, which generally is more challenging than format standardization.42 A permissions header/Knowbot standard is a system for representing legal concepts and for defining legal relations. As such, the standard is basically a grammar for a rule based substantive system in a very narrow domain.43 The data elements must correspond to legally meaningful relational attributes. The allowable values must correspond to legally allowable rights, obligations, privileges and powers. In other words, the standard setter must meet many of the challenges that a legal expert system designer working with Hohfeldian frameworks must meet.44 This adds a constraint to the standards setting process. Unlike setting format standards, where the participants are free to agree on an arbitrary way of expressing format attributes, participants in setting a content standard must remain within the universe of permissible content. The set of permissible values is determined by the law rather than being determined only by the imagination of format creators.

Enforcement and Bottlenecks

One of the many profound observations by Ithiel de Sola Pool was that copyright always has depended upon technological bottlenecks for its enforceability. The printing press was the original enforcement bottleneck. Now, a combination of the printing press and the practical need to inventory physical artifacts representing the work constitute the enforcement bottlenecks. As technologies change, old bottlenecks disappear and enforceability requires a search for new bottlenecks. When there are single hosts, like Westlaw, Dialog, Lexis, and CompuServe, access to that host is the bottleneck. The problem with distributed publishing on an open architecture internet is that there is no bottleneck in the middle of the distribution chain corresponding to the printer, the warehouse or the single host.

If new bottlenecks are to be found, they almost surely will be found at the origin and at the point of consumption. Encryption and decryption techniques discussed elsewhere in this volume concentrate on those bottlenecks as points of control. It also is possible that rendering software could become the new bottleneck as Mr. Linn suggests.

Even with those approaches, however, a serious problem remains in that the new technologies make it difficult or impossible to distinguish between mere use and copying. Thus the seller cannot distinguish between an end user45 and a potential competitor. On the other hand, the new technologies permit a much better audit trail, potentially producing better evidence for enforcement adjudication.

If network architectures for electronic publishing evolve in the way that Ted Nelson suggests with his Xanadu concept, the real value will be in the network and the pointers, not in the raw content. Thus, the creative and productive effort that the law should reward is the creation and productive effort that the law should reward is the creation and production and delivery of pointers, presentation, distribution, and duplication value. If this is so, then technological means will be particularly important, foreclosing access by those lacking passwords and other keys and limiting through contract what a consumer may do with the information.

In such an architecture, the law either will be relatively unimportant because technology can be counted on to prevent free riding or, the law will need to focus not on prohibiting copying or use without permission, but on preventing circumvention of the technological protections. Thus, legal approaches like that used to prevent the sale of decryption devices for television broadcasts and legal issues associated with contract enforcement may be more important than traditional intellectual property categories.

Weighing Risks and Costs

The law generally imposes sensible levels of transaction costs. Usually, transaction costs are proportional to the risk. Figure 1 shows a continuum of risk and transaction cost in traditional and new technologies. A real estate closing involves significant risks if there is some dispute later about the transaction. Therefore, the law affords much protection, including a constitutional officer called a registrar of deeds who is the custodian of records associated with the transaction. The risk level analogous to this in electronic publishing might be access to an entire library including access software as well as contents. Next, is a transaction involving a will or power of attorney. There, the risk is substantial because the maker of the instrument is not around to help interpret it. The law requires relatively high levels of assurance here, though not as great as those for real estate transactions. The law requires witnesses and attestation by a commissioned minor official called a notary public. The electronic publishing analogy of this level of risk might be the contents of an entire CDROM.

Next, in level of risk is the purchase of a large consumer durable like an automobile. The law requires somewhat less, but still significant protections for this kind of transaction: providing for the filing and enforcement of financing statements under the Uniform Commercial Code. The electronic publishing analogy might be the transfer of copyright to a complete work. Next, down

the risk continuum, is the purchase of a smaller consumer durable like a television set. Here, the law typically is reflected in written agreements of sale, but no special third party custodial mechanisms. The electronic publishing analogy might be use permission for a complete work.

Finally, is the purchase of a relatively small consumer item, say a box of diskettes. Neither the law or commercial practice involves much more than the exchange of the product for payment, with no written agreement or anything else to perform channeling, cautionary, evidentiary, or protective functions [make sure these function and the citation appears earlier]. The electronic publishing analogy might be use permission for part of a work.

Cost effectiveness = risk-proportional security

traditional transaction equivalent	institutions	electronic
real estate closing software and	registrar of deeds	entire library - contents
will/power of attorney CDROM	witnesses, notary public	contents of entire
auto purchase transfer of	UCC financing statement	complete work - copyright
television set purchase permission	written sale agreement	complete work - use
box of diskettes	-	part of a work - use permission

An encrypted object combined with rendering software is probably inconsistent with an open architecture. Because of the difficulty of setting standards for such technologies, this approach to intellectual property protection probably would be effectuated by proprietary approaches thus frustrating the vision of an open market for electronic publishing.

Conclusion

Realization of the digital library vision requires a method for collecting money and granting permission to use works protected by intellectual property. The concept of a knowbot and a permissions header attached to the work is the right way to think about such a billing and collection system. Standards for the data structures involved must be agreed to, and systems must be designed to satisfy legal formalities aimed at ensuring awareness of the legal significance of transactions and reliable proof of the terms of the transactions.

In the long run, not only must these technological issues be resolved, with appropriate attention to levels of risk and protections available under traditional legal doctrines, but also further conceptual development must be undertaken. Proponents of electronic publishing over wide area networks need to think about the appropriate metaphors: whether it is a library or a bookstore, if a library whether with or without xerox machines, if a bookstore whether it is a retail bookstore, or a mail order operation. Then,

thought must be given to how standards will be set. Finally, and most important, much more needs to be understood about the need for third party institutions. There is a good deal of enthusiasm for public key encryption. Yet the vulnerability of public key encryption systems is in the integrity of the key authority. In traditional legal protections, the third party custodians or authenticating agents like notary public and registrars of deeds receive state sanction and approval, and in the case of registrars of deeds, public funding. We must be clearer as to whether a similar infrastructure must be developed to protect against substantial risks and the use of EDI and electronic publishing technologies.

Finally, and perhaps most importantly, we must be thoughtful about what legal obligations, imposed on whom, are appropriate? The suggested 102(e) and (f) in the High Performance Computing Act looks very much like King James I's licensing of printing presses. It also looks like the FBI's proposal to prohibit the introduction of new technologies until certain conformity with past legal concepts is assured. Such approaches make the law a hurdle to new technology -- an uncomfortable position for both law and technology.

---

1 The use of EDI techniques to meter usage and determine charges for use of intellectual property is an example of billing and collection value in a typology of different types of value that can be produced in electronic marketplaces for information. See Henry H. Perritt, Jr., Market Structures for Electronic Publishing and Electronic Contracting in Brian Kahin, ed., Building Information Infrastructure: Issues in the Development of the National Research and Education Network (Harvard University and McGraw-Hill 1992) (developing typology for different types of value and explaining how market structures differ for the different types); Henry H. Perritt, Jr., Tort Liability, the First Amendment, and Equal Access to Electronic Networks, 5 Harv.J.Law & Tech. 65 (1992) (using typology of ten types of value to analyze access by competing producers of value).

2 See, e.g.U.S.Pat. No. 5,016,009, Data compression apparatus and method (May 14, 1991); U.S. Pat. No. 4,996,690, Write operator with gating capability (Feb. 26, 1991);U.S. Pat. No. 4,701,745, Data compression system (Oct. 20, 1987); Multi Tech Systems, Inc. v. Hayes Microcomputer Products, Inc., 800 F. Supp. 825 (D. Minn. 1992) (denying summary judgment on claim that patent for modem escape sequence is invalid)..

3 Comments on the 8\21 draft of "Knowbots in the Real World" from the intellectual property workshop participants at page 6 (author unknown, source unknown). Professor Samuelson also observed that the workshop, despite its title, actually did not focus much on intellectual property issues.

4 Corporation for National Research Initiatives, Workshop On The Protection Of Intellectual Property Rights In A Digital Library System: Knowbots in the Real World-May 18-19, 1989 (describing digital library system).

5 See generally Clifford A. Lynch, Visions of Electronic Libraries (libraries of future can follow acquisition-on-demand model rather than acquiring an advance of use; Z39.50 protocol will facilitate realization of that possibility, citing Robert E. Kahn & Vinton G. Serf, An Open Architecture

for a Digital Library System and a Plan for Its Development. The Digital Library Project, volume 1: The World of Knowbots (draft) (Washington D.C.: Corporation for National Research Initiatives; 1988)).

6 Clifford A. Lynch, The Z39.50 Information Retrieval Protocol: An Overview and Status Report, ACM Sigcomm Computer Communication Review at 58 (describing Z39.50 as an OSI application layer protocol that relieves clients from having to known the structure of data objects to be queried, and specifies a framework for transmitting and managing queries and results and syntax for formulating queries).

7 Brewster Kahle, Wide Area Information Server Concepts (Nov. 3, 1989 working copy; updates available from Brewster @THINK. (describing WAIS as "open protocol for connecting user interfaces on workstations and server computers") (describing information servers as including bulletin board services, shared databases, text searching and automatic indexing and computers containing current newspapers and periodicals, movie and television schedules with reviews, bulletin boards and chat lines, library catalogues, Usenet articles).

8 Robert E. Kahn, Deposit, Registration, Recordation in an Electronic Copyright Management System (August 1992) (Corporation for National Research Initiatives, Reston, Virginia).

9 Kahn 1992 at 4.

10 Kahn 1992 at 6.

11 Kahn 1992 at 10.

12 Kahn 1992 at 12.

13 Kahn 1992 at 15.

14 Browsability through techniques like the collapsible outliner function in Microsoft Word for Windows and competing products require more chunking and tagging value in the form of style and text element codes. Handling this additional formatting information through encryption and description processes is problematic.

15 " A 'transfer of copyright ownership' is an assignment, mortgage, exclusive license, or any other conveyance, alienation, or hypothecation of a copyright or of any of the exclusive rights comprised in a copyright, whether or not it is limited in time or place of effect, but not including a non-exclusive license " 17 U.S.C. 101 (1988).

16 17 U.S.C. 204(a) (1988); Valente-Kritzer Video v. Pinckney, 881 F.2d 772, 774 (9th Cir. 1989) (affirming summary judgment for author; oral agreement unenforceable under Copyright Act); Library Publications, Inc. v. Medical Economics Co., 548 F. Supp. 1231, 1233 (E.D. Pa. 1982) (granting summary judgment against trade book publisher who sought enforcement of oral exclusive distribution agreement; transfer of exclusive rights, no matter how narrow, must be in writing), aff'd mem., 714 F.2d 123 (3d Cir. 1983).

17 17 U.S.C. 205 (1988) provides constructive notice of the contents of the recorded document, determining priority as between conflicting transfers, and determines priority as between recorded transfer and non-exclusive license. The

former requirement for transfers to be recorded in order for the transferee to maintain an infringement, 17 U.S.C. 205(d), was repealed by the Berne Act Amendments 5.

18 under Adams v. Burke, 84 U.S. (17 Wall.) 453 (1873), a patentee must not attempt to exert control past the first sale. In general, use restrictions may be placed only on licensees, consistent with General Talking Pictures v. Western Elec., 304 U.S. 175 (1938). See generally Baldwin-Lima-Hamilton Corp. v. Tatnall, 169 F. Supp. 1 (E.D. Pa.1958) (applying no control after purchase rule).

19 See Red-Baron-Franklin Park, Inc. v. Taito Corp., 883 F.2d 275, 278 (4th Cir. 1989) (purchase of video game circuit boards did not create privilege to perform video game under first sale doctrine); United States v. Moore, 604 F.2d 1228, 1232 (9th Cir. 1979) (pirated sound recording not within first sale doctrine in criminal copyright infringement prosecution). But see Mirage Editions, Inc. v. Albuquerque A.R.T. Co., 856 F.2d 1341, 1344 (9th Cir. 1988) (first sale doctrine did not create privilege to prepare derivative work by transferring art in book to ceramic tiles).

20 The way in which the first sale doctrine would impact the electronically imposed use restrictions is by frustrating a breach-of-contract lawsuit by the licensor against a licensee who exceeds the use restrictions. The licensee exceeding the use restrictions would argue that it violates public policy to enforce the restrictions and therefore that state contract law may not impose liability for their violation. See generally Restatement (second) of Contracts 178 (1981) (stating general rule for determining when contract term in unenforceable on grounds of public policy).

21 In addition, as ___ of this paper notes, the Copyright Act itself requires signed writings for transfers of copyright interests. 17 U.S.C. 204(a). (1988).

22 Michael S. Baum & Henry H. Perritt, Jr., Electronic Contracting, Publishing and EDI Law ch. 6 (1991) (contract, evidence and agency issues) [hereinafter "Baum & Perritt"]. Accord, Signature Requirements Under EDGAR, Memorandum from D. Goelzer, Office of the General Counsel, SEC to Kenneth A. Fogash, Deputy Executive Director, SEC (Jan. 13, 1986) (statutory and non-statutory requirements for "signatures" may be satisfied by means other than manual writing on paper in the hand of the signatory . . . "In fact, the electronic transmission of an individual's name may legally serve as that person's signature, providing it is transmitted with the present intention to authenticate.").

23 17 U.S.C. 101 (1988). For copyright purposes, a work is created, and therefore capable of protection, when it is fixed for the first time. 17 U.S.C. 101 (1988). "[I]t makes no difference what the form, manner, or medium of fixation may be - whether it is in words, numbers, notes, sounds, pictures, or any other graphic or symbolic indicia, whether embodied in a physical object in written, printed, photographic, sculptural, punched, magnetic, or any other stable form, and whether it is capable of perception directly or by means of any machine or device 'now known or later developed.'" 1976 U.S. Code Cong. & Admin. News 5659, 5665. The legislative history further says that, "the definition of 'fixation' would exclude from the concepts purely of an evanescent or transitory nature -- reproductions such as those projected briefly on a screen

shown electronically on a television or other video display or captured momentarily in the 'memory' of a computer." 17 U.S.C. 102 note (excerpting from House Report 94-1476).

24 Or, more likely, what is on computer medium read by computer x, such as a magnetic cartridge used for archival records. Further references in the textual discussion to "what is in computer x now" should be understood to include such computer readable media.

25 Cf. Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 NW.U.L.Rev. 956, 980 (1986) (proof that a printout accurately reflects what is in the computer is too limited a basis for authentication of computer records).

26 In some cases, the electronic transaction will be accomplished by means of a physical transfer of computer readable media. In such a case, this step in the proof would involve proving what was received physically.

27 See generally Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 NW.U.L.Rev. 956, 979 (1986) (citing as examples of authentication Ford Motor Credit Co. v. Swarens, 447 S.W.2d 53 (Ky. 1969) (authentication by establishing relationship between computer-generated monthly summary of account activity and the customer reported on); Ed Guth Realty, Inc. v. Gingold, 34 N.Y.2d 440, 315 N.E.2d 441, 358 N.Y.S.2d 367 (1974) (authentication of summary of taxpayer liability and the taxpayer)).

28 Of course, a paper document signed at the end also is probative of the fact that no alternations have been made. In this sense, a signature requirement telescopes several steps in the inquiry outlined in the text.

29 United States v. Linn, 880 F.2d 209, 216 (9th Cir. 1989) (computer printout showing time of hotel room telephone call admissible in narcotics prosecution). See also United States v. Miller, 771 F.2d 1219, 1237 (9th Cir. 1985) (computer generated toll and billing records in price-fixing prosecution based on testimony by billing supervisor although he had no technical knowledge of system which operated from another office; no need for programmer to testify; sufficient because witness testified that he was familiar with the methods by which the computer system records information).

30 See United States v. Hutson, 821 F.2d 1015, 1020 (5th Cir. 1987) (remanding embezzlement conviction, although computer records were admissible under business records exception, despite trustworthiness challenged based on fact that defendant embezzled by altering computer files; access to files offered in evidence was restricted by special code).

31 Restatement (Second) of Contracts ___ (1981).

32 Cite for when extrinsic evidence is admissible.

33 See Baum & Perritt 2.6; The Electronic Messaging Services Task Force, The Commercial Use of Electronic Data Interchange--A Report and Model Trading Partner Agreement, 45 Bus.Law. 1645 (1990); Jeffrey B. Ritter, Scope of the Uniform Commercial Code: Computer Contracting Cases and Electronic Commercial Practices, 45 Bus.Law. 2533 (1990); Note, Legal Responses to Commercial Transactions Employing

Novel Communications Media, 90 Mich.L.Rev. 1145 (1992)

34 Garber v. Harris Trust & Savings Bank, 432 N.E.2d 1309, 1311-1312 (Ill. App. 1982) ("each use of the credit card constitutes a separate contract between the parties;" citing cases).

It is not quite this simple, because both merchant and credit card customer have separate written contracts with the credit card issuer. But there is no reason that a supplier of information to a Digital Library System and all customers of that system might not have their own contracts with the Digital Library System in the same fashion.

35 Rowland v. Union Hills Country Club, 757 P.2d 105 (Ariz. 1988) (reversing summary judgment for country club officers because of factual question whether club followed bylaws in expelling members); Straub v. American Bowling Congress, 353 N.W.2d 11 (Neb. 1984) (rule of judicial deference to private associations, and compliance with association requirements, counseled affirmance of summary judgment against member of bowling league who complained his achievements were not recognized). But see Wells v. Mobile County Board of Realtors, Inc., 387 So.2d 140 (Ala. 1980) (claim of expulsion of realtor from private association was justiciable and bylaws, rules and regulations requiring arbitration were void as against public policy; reversing declaratory judgment for defendant association).

36 F.R.E. 803(6) (excluding business records from inadmissibility as hearsay); 28 U.S.C. 1732 ("Business Records Act" permitting destruction of paper copies of government information reliably recorded by any means and allowing admission of remaining reliable record).

37 See Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 Nw.U.L.Rev. 956, 978-80, 984-85 (1986) (noting body of commentator opinion saying that business records exception and authentication are parallel ways of establishing reliability).

38 See F.R.E. 901(b)(4) (appearance, contents, substance, internal patterns, as examples of allowable authentication techniques).

39 Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 Nw.U.L.Rev. 956, 963-64 (1986) (identifying steps and trend resulting in F.R.E.).

40 Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 Nw.U.L.Rev. 956, 963 (1986).

41 Peritz, Computer Data and Reliability: A Call for Authentication of Business Records Under the Federal Rules of Evidence, 80 Nw.U.L.Rev. 956, 974 (1986) (reporting four requirements of Manual, and endorsing their use generally).

42 See Henry H. Perritt, Jr., ___, ___ Jurimetrics ___ (1993) (distinguishing between format and content standardization).

43 See Marc Lauritsen, ___ (explaining relationship between substantive legal systems and the field of artificial intelligence).

44 See Thorne, McCarty; Kevin Ashley; and Gardner.

45 It may not be particularly important to limit competition by consumers, because the consumers will never have the pointers and the rest of the network infrastructure.

The Artech HouseUniversal Personal Communications Series

Ramjee Prasad, Series Editor

CDMA for Wireless Personal Communications, Ramjee Prasad

Universal Wireless Personal Communications, Ramjee Prasad

Wideband for Third Generation Mobile Communications, Tero Ojanperä and Ramjee Prasad

For further information on these and other Artech House titles, including previously considered out-of-print books now available through our In-Print-Forever® (IPF®) program, contact:

Artech House
685 Canton Street
Norwood, MA 02062
Phone: 781-769-9750
Fax: 781-769-6334
e-mail: artech@artechhouse.com

Artech House
46 Gillingham Street
London SW1V 1AH UK
Phone: +44 (0)20 7596-8750
Fax: +44 (0)20 7630-0166
e-mail: artech-uk@artechhouse.com

Find us on the World Wide Web at:
www.artechhouse.com

For a recent listing of titles in the *Artech House Mobile Communications Library*, turn to the back of the book.

02

Wideband CDMA for Third Generation Mobile Communications

Tero Ojanperä
Ramjee Prasad
editors

Artech House
Boston • London

Library of Congress Cataloging-in-Publication Data
Ojanperä, Tero
Wideband CDMA for third generation mobile communications / Tero Ojanperä, Ramjee Prasad, editors.
p. cm.
Includes bibliographical references and index.
ISBN 0-89006-735-X (alk. paper)
1. Code division multiple access. 2. Mobile communication systems.
3. Broadband communication systems. I. Prasad, Ramjee. II. Title.
TK5103.45.034 1998
621.3845—dc21 98-33857
CIP

British Library Cataloguing in Publication Data
Wideband CDMA for third generation mobile communications
1. Code division multiple access 2. Broadband communication systems
3. Global system for mobile communications
I. Ojanperä, Tero II. Prasad, Ramjee
6213'84'56

ISBN 0-89006-735-X

Cover design by Lynda Fishbourne

Books are the joy of life, and learning is a never-ending experience.

To my wife Tiina, and to our son Eerik
To my brother Aki, to my mother Maiju and to the memories of my brother Juha and father Iisakki
—Tero Ojanperä

To my wife Jyoti, to our daughter Neeli, and to our sons Anand and Rajeev
—Ramjee Prasad

might degrade the performance. Therefore, in some instances it might be better to implement higher bit rates with a multicode scheme.

5.9.1.3 *Forward Link Orthogonality*

Since the forward link is synchronous, it is possible to maintain orthogonality between the codes. For multicode transmission we can select a set of orthogonal codes. With VSF, it is also possible to maintain orthogonality between different spreading factors if we impose the following constraint between different rates:

$$R = R_c/2^n \qquad n = 0, 1, 2, \ldots \tag{5.3}$$

This can be achieved with the variable length Walsh sequences or with the three structured orthogonal codes [16] discussed in Section 5.6.3.

5.9.1.4 *Power Control Requirements*

In order for the receiver to be able to estimate the path loss for power control purposes, transmission power should not vary. Otherwise, the receiver has to first know the rate of a traffic channel. Another possibility is to use a fixed rate control channel for the power measurements. With multicode transmission, each of the parallel channels has a fixed power. For VSF, the power varies according to transmission rate, and thus, explicit transmission power needs to be signaled to the receiver. One further advantage of multicode transmission is that in case of several simultaneous services, the QoS can be adjusted using power control. Parallel service with similar power requirements can be time multiplexed, while code multiplexing is used for parallel services with different power requirements.

5.9.1.5 *Complexity*

Power amplifier linearity requirements should be as low as possible to allow the use of power-efficient power amplifiers. The multicode scheme results in larger envelope variations than the single code transmission. An additional drawback of the multicode scheme is that it requires as many RAKE receivers as there are codes. However, each RAKE receiver may be less complex, since a higher spreading factor might facilitate the use of fewer bits for quantization [20].

5.9.2 Granularity of Data Rates

Granularity means the minimum possible bit rate change. In order to maintain high flexibility, it is desirable to have as fine of granularity as possible. For multicode, granularity is *R/M* Kbps. If one code has the capability of smaller quantization of data rates, then better granularity can be achieved. For the VSF scheme, granularity varies according to the spreading factor and is finest for the low bit rates [20]:

$$SF = 1 \quad SF = \frac{R_c}{R} - 1 \qquad R_c = R$$

where SF is the spreading factor, R_c is the chip rate, and R is the user bit rate.

For low bit rate services, this would most likely be sufficiently fine granularity. However, if we restrict the spreading factors according to (5.2), then the granularity is too coarse. Furthermore, if we assume for multicode transmission that one code carries 10 Kbps of traffic, the basic granularity is 10 Kbps. This is too large for some services such as low rate speech codecs, which typically require better granularity. The problem of granularity can also be considered from the viewpoint that, given the coded user bit rate, it is possible to match it with the given chip rate. For example, if one code carries 32 ksymbol/s and the source data rate is 9.6 Kbps coded with a 1/3 convolutional code, it results in a gross bit rate of 28.8 Kbps. Thus we need to match the 28.8 Kbps to 32 Kbps.

One alternative to matching the bit and chip rate is the use of rate compatible punctured codes (RCPC) [33]. The basic idea of RCPC coding is to divide the bit stream of the mother code into blocks whose bits are either punctured or repeated according to a perforation/repetition matrix. The drawback of this approach is that a fixed number of rates have to be selected, since not all service rates can be directly matched to the available chip rate. Furthermore, for RCPC, good codes are only known up to the constraint length of 7.

Repetition coding, or puncturing, is another possibility for rate matching. Even unequal repetition coding (i.e., only some symbols in the frame are repeated to implement the desired symbol rate within the frame) can be used [17].

5.9.3 Transmission of Control Information

In order to vary the data rate or other service parameters, the receiver needs to know the structure of the received signal. This can be achieved either by transmitting explicit control information or by blind rate detection, for example, from the CRC information [34]. In case explicit control information is transmitted, the following issues need to be solved:

- Coding of the control information to achieve desired quality of service;
- Multiplexing of the control information;
- Position of the control information.

The control bits have to have a considerably lower error rate than the information bits, since, if a control word has an error, the whole frame is lost. Control information can either be coded together with the user data or independently from the user data. Furthermore, control information can be transmitted either code multiplexed or time multiplexed.

There are two possibilities for the position of control information: in the previous frame or in the same frame as the user data, as illustrated in Figure 5.14. Since the receiver is informed in advance of the transmission parameters, the processing of

the data can be done "on-line." However, for services with a short delay requirement the additional delay of one frame might be too long. A further drawback is that erroneous control information results in loss of the previous and the next frame, except if the next frame is transmitted with the same parameters as the previous frame.

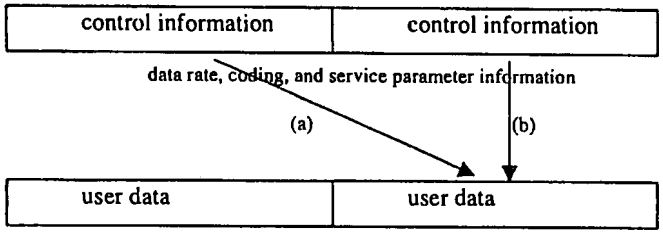


Figure 5.14 Control information transmitted (a) in the previous frame, (b) in the same frame as user data.

In case the control information is transmitted within the same frame as the user data, then the receiver needs to decode the control information first. Thus, the data need to be buffered. Memory requirements for the buffer depend on the spreading code solution. In case the spreading ratios are multiples of each other, it is possible to despread with the lowest spreading ratio and to buffer the sub-symbols after despreading, instead of the samples before the despreading, which need to be used if arbitrary spreading ratios are used. However, for high data rates buffering might be still a problem.

5.10 PACKET DATA

Since non real-time packet data services are not delay sensitive, they use the retransmission principle implemented with ARQ protocol to improve the error rate. The retransmission protocol can be either implemented in layer 2 as part of MAC and RLP or in the physical layer (layer 1). If packet data retransmission is implemented as part of the layer 2, the transmission of packet data in the physical layer does not differ from the transmission of circuit switched data. So the multirate aspects discussed above also apply to the transmission of packet data. If the physical layer ARQ is used, then the physical layer is modified depending on the ARQ scheme used (see Section 5.8 about ARQ schemes). In both cases, the access procedure and handover for packet data services have certain special implications, which are discussed in the next subsections.

5.10.1 Packet Access Procedure

The packet access procedure in CDMA should minimize the interference [illegible]. Since there is no connection between the base station and the mobile station [illegible] access procedure, initial access is not power controlled and thus the [illegible] transmitted during this period should be minimized. There are three scenarios for packet access:

- Infrequent transmission of short packets containing little information;
- Transmission of long packets;
- Frequent transmission of short packets.

Since the establishment of a traffic channel itself requires signaling and thus consumes radio resources, it is better to transmit small packets within the random access message without power control. For long and frequent short packets, a dedicated traffic channel should be allocated.

If a dedicated channel has been reserved and there is nothing to send, the mobile station either cuts off the transmission or keeps the physical connection by transmitting power control and reference symbols only. In the former case, a virtual connection (higher layer protocols) is retained in order to rapidly re-establish the link in case of a new transmission. Selection between these two alternatives is a trade-off between resources spent for synchronization and power control information, and resources spent for random access.

5.10.2 MAC Protocol

The task of the medium access protocol is to share the transmission medium with different users in a fair and efficient way. Sometimes, multiple access protocols such as FDMA, CDMA, and TDMA are also classified as medium access protocols. However, as already discussed in the beginning of this chapter, the medium access protocol is part of the link layer, while the multiple access scheme is part of the physical layer. The medium access protocol has to resolve contention between users accessing the same physical resource. Thus, it also manages the packet access procedure described in the previous section. Since the third generation systems offer a multitude of services to customers at widely varying quality of service requirements, the MAC needs to offer capabilities to manage the access demands of different users and different service classes. This can be performed using reservation and priority schemes. Services with delay constraints can use a reservation scheme to reserve capacity to guarantee the quality of service. Priority schemes can be used to prioritize the requests from different services.

5.10.3 Packet Data Handover

Since CDMA operates with a reuse factor of one, it needs efficient and fast handover in order to avoid excessive interference with the other cells. This has been realized with soft handover in the case of circuit switched connections. Soft handover also improves [illegible] performance through increased diversity. For packet connections, and especially for [illegible] there may be no need to establish soft handover even if the user is at the [illegible] However, there is still the need to route packets via the base station that [illegible] best connection. This is more important with frequent packet transmissions

A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms

TAHER ELGAMAL, MEMBER, IEEE

Abstract—A new signature scheme is proposed, together with an implementation of the Diffie–Hellman key distribution scheme that achieves a public key cryptosystem. The security of both systems relies on the difficulty of computing discrete logarithms over finite fields.

I. Introduction

In 1976, Diffie and Hellman [3] introduced the concept of public key cryptography. Since then, several attempts have been made to find practical public key systems (see, for example, [6], [7], [9]) depending on the difficulty of solving some problems. For example, the Rives–Shamir–Adleman (RSA) system [9] depends on the difficulty of factoring large integers. This paper presents systems that rely on the difficulty of computing logarithms over finite fields.

Section II shows a way to implement the public key distribution scheme introduced by Diffie and Hellman [3] to encrypt and decrypt messages. The security of this system is equivalent to that of the distribution scheme. Section III introduces a new digital signature scheme that depends on the difficulty of computing discrete logarithms over finite fields. It is not yet proved that breaking the system is equivalent to computing discrete logarithms. Section IV develops some attacks on the signature scheme, none of which seems to break it. Section V gives some properties of the system. Section VI contains a conclusion and some remarks.

II. The Public Key System

First, the Diffie–Hellman key distribution scheme is reviewed. Suppose that A and B want to share a secret K_{AB}, where A has a secret x_A and B has a secret x_B. Let p be a large prime and α be a primitive element mod p, both known. A computes $y_A \equiv \alpha^{x_A} \bmod p$, and sends y_A. Similarly, B computes $y_B \equiv \alpha^{x_B} \bmod p$ and sends y_B. Then the secret K_{AB} is computed as

$$\begin{aligned} K_{AB} &\equiv \alpha^{x_A x_B} \bmod p \\ &\equiv y_A^{x_B} \bmod p \\ &\equiv y_B^{x_A} \bmod p. \end{aligned}$$

Hence both A and B are able to compute K_{AB}. But, for an intruder, computing K_{AB} appears to be difficult. It is not yet proved that breaking the system is equivalent to computing discrete logarithms. For more details refer to [3].

In any of the cryptographic systems based on discrete logarithms, p must be chosen such that $p-1$ has at least one large prime factor. If $p-1$ has only small prime factors, then computing discrete logarithms is easy (see [8]).

Now suppose that A wants to send B a message m, where $0 \le m \le p-1$. First A chooses a number k uniformly between 0 and $p-1$. Note that k will serve as the secret x_A in the key distribution scheme. Then A computes the "key"

$$K \equiv y_B^k \bmod p, \tag{1}$$

where $y_B \equiv \alpha^{x_B} \bmod p$ is either in a public file or is sent by B. The encrypted message (or ciphertext) is then the pair (c_1, c_2), where

$$c_1 \equiv \alpha^k \bmod p \qquad c_2 \equiv Km \bmod p \tag{2}$$

and K is computed in (1).

Note that the size of the ciphertext is double the size of the message. Also note that the multiplication operation in (2) can be replaced by any other invertible operation such as addition mod p.

The decryption operation splits into two parts. The first step is recovering K, which is easy for B since $K \equiv (\alpha^k)^{x_B} \equiv c_1^{x_B} \bmod p$, and x_B is known to B only. The second step is to divide c_2 by K and recover the message m.

The public file consists of one entry for each user, namely y_i for user i (since α and p are known for all users). It is possible that each user chooses his own α and p, which is preferable from the security point of view although that will triple the size of the public file.

It is not advisable to use the same value k for enciphering more than one block of the message, since if k is used more than once, knowledge of one block m_1 of the message enables an intruder to compute other blocks as follows. Let

$$c_{1,1} \equiv \alpha^k \bmod p \qquad c_{2,1} \equiv m_1 K \bmod p,$$

$$c_{1,2} \equiv \alpha^k \bmod p \qquad c_{2,2} \equiv m_2 K \bmod p.$$

Then $m_1/m_2 \equiv c_{2,1}/c_{2,2} \bmod p$, and m_2 is easily computed if m_1 is known.

Breaking the system is equivalent to breaking the Diffie–Hellman distribution scheme. First, if m can be computed from c_1, c_2, and y, then K can also be computed from y, c_1, and c_2 (which appears like a random

Manuscript received February 6, 1984; revised November 12, 1984. This work was supported by the National Science Foundation under Grant ECS83 07741. The material in this paper was presented at the IEEE Crypto '84 Conference, August 1984, Santa Barbara, CA.

The author was with the Information Systems Laboratory, Stanford University, Stanford, CA. He is now with Hewlett-Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304, USA.

0018-9448/85/0700-0469$01.00 ©1985 IEEE

number since k and m are unknown). That is equivalent to breaking the distribution scheme. Second, (even if m is known) computing k or x from c_1, c_2, and y is equivalent to computing discrete logarithms. The reason is that both x and k appear in the exponent in y and c_1.

III. A Digital Signature Scheme

A new signature scheme is described in this section. The public file contains the same public keys for encrypting messages as well as verifying signatures.

Let m be a document to be signed, where $0 \le m \le p-1$. The public file still consists of the public key $y \equiv \alpha^x \bmod p$ for each user. To sign a document, a user A should be able to use the secret key x_A to find a signature for m in such a way that all users can verify the authenticity of the signature by using the public key y_A (together with α and p), and no one can forge a signature without knowing the secret x_A.

The signature for m is the pair (r, s), $0 \le r, s < p-1$, chosen such that the equation

$$\alpha^m \equiv y^r r^s \bmod p \tag{3}$$

is satisfied.

A. The Signing Procedure

The signing procedure consists of the following three steps.

1) Choose a random number k, uniformly between 0 and $p-1$, such that $\gcd(k, p-1) = 1$.

2) Compute

$$r \equiv \alpha^k \bmod p. \tag{4}$$

3) Now (3) can be written as

$$\alpha^m \equiv \alpha^{xr}\alpha^{ks} \bmod p, \tag{5}$$

which can be solved for s by using

$$m \equiv xr + ks \bmod (p-1). \tag{6}$$

Equation (6) has a solution for s if k is chosen such that $\gcd(k, p-1) = 1$.

B. The Verification Procedure

Given m, r, and s, it is easy to verify the authenticity of the signature by computing both sides of (3) and checking that they are equal.

Note 1: As will be shown in Section IV, the value of k chosen in step 1) should never be used more than once. This can be guaranteed, for example, by using as a "k generator" a DES chip used in the counter mode as a stream cipher.

IV. Some Attacks on the Signature Scheme

This section introduces some of the possible attacks on the signature scheme. Some of these attacks are easily shown to be equivalent to computing discrete logarithms over $GF(p)$. It has not yet been proved that breaking the signature scheme is equivalent to computing discrete logarithms, or equivalent to breaking the distribution scheme. However, none of the attacks shown in this section appear to break the system. The reader is encouraged to develop new attacks, or find fast algorithms to perform one of the attacks described in this section. The attacks will be divided into two groups. The first group includes some attacks for recovering the secret key x, and in the second group we show some attacks for forging signatures without recovering x.

A. Attacks Aiming to Recover x

Attack 1: Given $\{m_i: i = 1, 2, \cdots, l\}$ documents, together with the corresponding signatures $\{(r_i, s_i): i = 1, 2, \cdots, l\}$, an intruder may try to solve l equations of the form (6). Since there are $l+1$ unknowns (since each signature uses a different k), the system of equations is underdetermined and the number of solutions is large. The reason is that each value for x yields a solution for the k_i since a system of linear equations with a diagonal matrix of coefficients will result. Since $p-1$ is chosen to have at least one large prime factor q, recovering $x \bmod q$ requires an exponential number of message-signature pairs.

Note 2: If any k is used twice in the signing, then the system of equations is uniquely determined and x can be recovered. So for the system to be secure, any value of k should never be used twice.

Attack 2: Trying to solve equations of the form (3) is always equivalent to computing discrete logarithms over $GF(p)$, since both unknowns x and k appear in the exponent.

Attack 3: An intruder might try to develop some linear dependencies among the unknowns $\{k_i, i = 1, 2, \cdots, l\}$. This is also equivalent to computing discrete logarithms since if $k_i \equiv ck_j \bmod (p-1)$, then $r_i \equiv r_j^c \bmod p$, and if c can be computed then computing discrete logarithms is easy.

B. Attacks for Forging Signatures

Attack 4: Given a document m, a forger may try to find r, s such that (3) is satisfied. If $r \equiv \alpha^j \bmod p$ is fixed for some j chosen at random, then computing s is equivalent to solving a discrete logarithm problem over $GF(p)$.

If the forger fixes s first, then r could be computed from the equation

$$r^s y^r \equiv A \bmod p. \tag{7}$$

Solving equation (7) for r is not yet proved to be at least as hard as computing discrete logarithms, but we believe that it is not feasible to solve (7) in polynomial time. The reader is encouraged to find a polynomial time algorithm for solving (7).

Attack 5: It seems possible that (3) can be solved for both r and s simultaneously, but we have not been able to find an efficient algorithm to do that.

Attack 6: The signature scheme allows the following attack, whereby the intruder, knowing one legitimate signature for one message, can generate other legitimate sig-

natures and messages. This attack does not allow the intruder to sign an arbitrary message and therefore does not break the system. This property exists in all the existing digital signature schemes and can be avoided by either requiring that m has to have a certain structure or by applying a one-way function to the message m before signing it.

Given a signature (r, s) for the message (m), then

$$\alpha^m = y^r r^s \bmod p.$$

Select integers A, B, and C arbitrarily such that $(Ar - Cs)$ is relatively prime to $p - 1$. Set

$$r' \equiv r^A \alpha^B y^C \bmod p,$$
$$s' \equiv sr'/(Ar - Cs) \bmod (p - 1),$$
$$m' \equiv r'(Am + Bs)/(Ar - Cs) \bmod (p - 1).$$

Then it is claimed that (r', s') signs the message (m'). Calculate

$$\begin{aligned} y^{r'} r'^{s'} &\equiv y^{r'} \left(r^A \alpha^B y^C\right)^{sr'/(Ar-Cs)} \\ &\equiv \left(y^{r'Ar - r'Cs + r'Cs} r^{Asr'} \alpha^{Bsr'}\right)^{1/(Ar-Cs)} \\ &\equiv \left(\left(y^r r^s\right)^{Ar'} \alpha^{Bsr'}\right)^{1/(Ar-Cs)} \\ &\equiv \alpha^{(mAr' + Bsr')/(Ar-Cs)} \\ &\equiv \alpha^{m'} \end{aligned}$$

(all calculations mod p).

As a special case, setting $A = 0$, legitimate signatures can be generated with corresponding messages without ever seeing any signatures:

$$r' \equiv \alpha^B y^C \bmod p,$$
$$s' \equiv -r'/C \bmod (p - 1),$$
$$m' \equiv -r'B/C \bmod (p - 1).$$

It can be shown that (r', s') signs (m').

V. Properties of Our System and Comparison to Other Signature Schemes and Public Key Systems

Let m be the number of bits in either p for the discrete logarithm problem or n for the integer factoring problem. Then the best known algorithm for both computing discrete logarithms and factoring integers (which is the function used in some of the existing systems such as the RSA system [9]) is given by (see [1], [5], [10])

$$O\left(\exp \sqrt{cm \ln m}\right), \tag{8}$$

where the best estimate for c is $c = 0.69$ for factoring integers (due to Schnorr and Lenstra [10]), as well as for discrete logarithms over GF(p) (see [5]). These estimates imply that we have to use numbers that are about the size of the numbers used in the RSA system in order to obtain the same level of security (assuming the current value for c for both the discrete logarithms problem and the integer factorization problem). So the size of the public file is larger than that for the RSA system. (For the RSA system, each user has one entry n as his public key, together with the encryption key in the public file.)

A. Properties of the Public Key System

As shown above, our system differs from the other known systems. First, due to the randomization in the enciphering operation, the cipher text for a given message m is not repeated, i.e., if we encipher the same message twice, we will not get the same cipher text $\{c_1, c_2\}$. This prevents attacks like a probable text attack where if the intruder suspects that the plain text is, for example, m, then he tries to encipher m and finds out if it was really m. This attack, and similar ones, will not succeed since the original sender chose a random number k for enciphering, and different values of k will yield different values of $\{c_1, c_2\}$. Also, due to the structure of our system, there is no obvious relation between the enciphering of m_1, m_2, and $m_1 m_2$, or any other simple function of m_1 and m_2. This is not the case for the known systems, such as the RSA system.

Suppose that p is of about the same size as that required for n in the case of the RSA system. Then the size of the cipher text is double the size of the corresponding RSA cipher text.

For the enciphering operation, two exponentiations are required. That is equivalent to about $2 \log p$ multiplications in GF(p). For the deciphering operation only one exponentiation (plus one division) is needed.

B. Properties of the Signature Scheme

For the signature scheme using the above arguments for the sizes of the numbers in our system and the RSA system, the signature is double the size of the document. Then the size of the signature is the same size as that needed for the RSA scheme, and half the size of the signature for the new signature scheme that depends on quadratic forms published by Ong and Schnorr [6], and also Ong, Schnorr, and Shamir [7] (since both systems are based on the integer factoring problem). The Ong–Schnorr–Shamir system has been broken by Pollard and new variations are being suggested. Thus, it is not clear at the present time whether a secure system based on modular equations can be found, and hence no further remarks will be made regarding these schemes.

Note that, since the number of signatures is p^2, while the number of documents is only p, each document m has a lot of signatures but any signature signs only one document.

For the signing procedure, one exponentiation (plus a few multiplications) is needed. To verify a signature, it seems that three exponentiations are needed, but it was pointed to the author by Shamir that only 1.875 exponentiations are needed. This is done by representing the three exponents m, r, s in their binary expansion. At each step square the number $\alpha^{-1} y r$ and divide by the necessary

factor to account for the different expansions of m, r, and s. The different multiples of α^{-1}, y, and r can be stored in a table consisting of eight entries. We expect that 0.875 of the time a multiplication is needed. That accounts for the 1.875 exponentiations needed.

VI. Conclusions and Remarks

The paper described a public key cryptosystem and a signature scheme based on the difficulty of computing discrete logarithms over finite fields. The systems are only described in $GF(p)$. The public key system can be easily extended to any $GF(p^m)$, but recent progress in computing discrete logarithms over $GF(p^m)$ where m is large (see [2, 5]) makes the key size required very large for the system to be secure. The subexponential time algorithm has been extended to $GF(p^2)$ [4] and it appears that it can be extended to all finite fields, but the estimates for the running time for the fields $GF(p^m)$ with a small m seem better at the present time. Hence, it seems that it is better to use $GF(p^m)$ with $m = 3$ or 4 for implementing a cryptographic system. The estimates for the running time of computing discrete logarithms and for factoring integers are the best known so far, and if the estimates remain the same, then, for the same security level, the size of the public key file and the size of the cipher text will be double the size of those for the RSA system.

Acknowledgment

The author would like to thank a referee for including Attack 6 described in Section IV.

References

[1] L. Adleman, "A subexponential algorithm for the discrete logarithm problem with applications to cryptography," in *Proc. 20th IEEE Symp. Foundations of Computer Science* 1979, pp. 55–60.

[2] D. Coppersmith, "Fast evaluation of logarithms in fields of characteristic two," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 587–594, 1984.

[3] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 472–492, 1976.

[4] T. ElGamal, "A subexponential-time algorithm for computing discrete logarithms over $GF(p^2)$," *IEEE Trans. Inform. Theory*, this issue.

[5] A. Odlyzko, "Discrete logarithms in finite fields and their cryptographic significance," *Proc. Eurocrypt 84*, to appear.

[6] H. Ong and C. Schnorr, "Signatures through approximate representations by quadratic forms," to appear.

[7] H. Ong, C. Schnorr, and A. Shamir, "An efficient signature scheme based on quadratic forms," in *Proc. 16th ACM Symp. Theoretical Computer Science*, 1984, pp. 208–216.

[8] S. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 106–110, 1978.

[9] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[10] C. Schnorr and H. W. Lenstra Jr., "A Monte Carlo factoring algorithm with finite storage," *Math. Comput.*, vol. 43, pp. 289–311, 1984.

Authorization for Metacomputing Applications

G. Gheorghiu, T. Ryutov and B.C. Neuman
Information Sciences Institute
University of Southern California
4676 Admiralty Way suite 1001
Marina del Rey, CA 90292
grig, tryutov, bcn@isi.edu
(310)822-1511 (voice) (310)823-6714 (fax)

Abstract

One of the most difficult problems to be solved by metacomputing systems is to ensure strong authentication and authorization. The problem is complicated since the hosts involved in a metacomputing environment often span multiple administrative domains, each with its own security policy. This paper presents a distributed authorization model used by our resource allocation system, the Prospero Resource Manager [8]. The main components of our design are Extended Access Control Lists, EACLs, and a General Authorization and Access API, GAA API. EACLs extend conventional ACLs to allow conditional restrictions on access rights. In the case of the Prospero Resource Manager, specific restrictions include limits on the computational resources to be consumed and on the characteristics of the applications to be executed by the system, such as name, version or endorser. The GAA API provides a general framework for applications to access the EACLs. We have built a prototype of the system.

1. Introduction

Metacomputing is sometimes defined as the abstraction of geographically dispersed computing and communication resources (e.g. supercomputers and high-speed networks) into a single metacomputer [2]. Ideally, the user of the system is presented with a consistent and familiar interface that hides the geographic scale, the complexity and the heterogeneity.

A metacomputing system usually crosses administrative domains and involves a very large number of computing resources. Such systems have particularly sensitive requirements for security. This is one of the most difficult requirements to satisfy, due to the large scale and heterogeneity of the resources involved. The problem is complicated by the variety of representations and by the application of access control policies across multiple administrative domains.

This paper describes the authentication and authorization mechanisms and policies used by the Prospero Resource Manager (PRM [8]), a scalable resource allocation system that manages processing resources in metacomputing environments. PRM uses Kerberos [9] to achieve strong authentication and integrates a new distributed authorization model. Because different administrative domains might use different security services for authentication of principals (e.g. DCE, X.509), we designed the system to be extensible, allowing a variety of security services to be used instead of or in addition to Kerberos. The model is based on two ideas:

1. Extended Access Control Lists (EACL): conventional Access Control Lists (ACL) are extended with an optional field added to each ACL entry specifying restrictions on authorized rights. In the case of PRM, the attributes include strength of authentication, limits on the physical resources managed by the system (e.g. CPU load, memory usage) and characteristics of applications that the users are willing to run on their processors (e.g. name, version, endorser).

2. General Authorization and Access API (GAA API): we defined a common API to facilitate authorization decisions for applications. PRM invokes the GAA API functions to determine if a requested operation or set of operations is authorized or if additional checks are necessary.

Ease of use and configurability are important issues to be considered for any resource management system. For this reason, we developed a scalable mechanism based on the Prospero Directory Service to facilitate the management of the extended access control lists.

The paper is organized as follows. Section 2 describes the Prospero Resource Manager. Section 3 presents the motivation for the authorization model applied to metacomputing applications. Section 4 discusses the two components of the distributed authorization model: the EACL framework and the GAA API. Section 5 shows how the model is adapted and integrated within PRM. Section 6 describes the management of the EACL using the Prospero Directory Service. Section 7 discusses related work.

2. The Prospero Resource Manager

The design of the Prospero Resource Manager was guided by the concept of the Virtual System Model, in which resources of interest are readily accessible and those of less interest are hidden from view [8]. PRM applies this concept to the problem of allocating resources in large scale systems by dividing the functions of resource management between three types of managers: the system manager, the job manager and the node manager. Each manager makes scheduling decisions at a different level of abstraction and this separation of management enables PRM to scale as the number of managed resources increases.

Throughout the paper, we will use the term *node* to denote a processing element, be it a processor in a multiprocessor environment or a workstation whose resources are made available for running jobs. A *job* consists of a set of communicating tasks, running on the nodes allocated to the job. A *task* consists of one or more threads of control of an application, together with the address space in which they run.

2.1. The system manager

In PRM, the total collection of processing resources is divided into subsets which correspond usually to administrative domains. Each subset is managed by a *system manager* which is responsible for allocating its resources to jobs as needed. The system managers themselves can be organized in a hierarchical manner in order to avoid bottlenecks and ensure scalability.

The system manager maintains information about the characteristics of each resource it manages, together with the mapping from resources to jobs. The system manager receives status updates from node managers (e.g. availability, load information) and uses them to make allocation decisions. The system manager also responds to resource requests from job managers.

2.2. The job manager

The *job manager* offers a single point of contact for applications to request necessary resources. It hides from the application the complexity of managing the resources that have been allocated by the system manager to a particular job.

When a job is initiated, the job manager locates system managers (by using the Prospero Directory Service if available or from a configuration file) and sends resource requests. If the response from the system manager is affirmative, then the job manager allocates the resources to the tasks in the job and contacts the node manager for each resource in order execute the tasks on the appropriate processors. If a system manager refuses the request, for example when the job manager is not authorized to make the request or when there are no resources available, then the job manager contacts other system managers which can satisfy the request.

2.3. The node manager

Each resource in the system is managed by a *node manager* which is informed by the system manager about job managers that are authorized to use the resource. When the node manager receives a request from an authorized job manager, it responds by loading and executing the requested program. The node manager sends messages to the job manager upon termination or failure of tasks and to the system manager about the availability of the node for future assignments.

3. Motivation for a New Authorization Model

Metacomputing systems cover large networks connecting mutually suspicious domains, which are independently administered.

Consider the following scenario. A user logs onto a machine and wants to perform a computation on a remote machine residing in a different security domain. Let us identify the security issues to be considered:

- Establishing a trust relationship of the users between different security domains. The domain security manager must maintain an authorization database listing principals authorized to request resources belonging to this domain.
- Access control and authorization policies to protect server resources. In a wide area network, it is unlikely that sites would make their resources available to others if there are no means of protection. There should be a flexible mechanism to represent user-defined security policies, such as:
  - type and amount of resources that the node is willing to allocate, e.g. memory, processors, terminal access, access to the local files and directories, network connections

- applications that can be run on the node, e.g. name of application, version, platform, endorser
- requirement of payment or accounting for the resources consumed

- Enforcement of the security policies. There should be a mechanism for monitoring execution of the program on a particular node to ensure that the program keeps strictly to the limits imposed by the local administrators

Specification of security policies for principals from multiple administrative domains poses additional problems:

- There are multiple mechanisms for authentication of users in different domains. Therefore, there may be no single syntax for specification of principal names
- Similarly there may be no standard security policy representation. Administrators of each domain might use domain-specific policy syntax and heterogeneous implementations of the policies

Therefore our goal was to design a flexible and expressive mechanism for representing and evaluating authorization policies. It should be general enough to support a variety of mechanisms based on public or secret key cryptosystems and provide integration of local and distributed security policies.

4. Overview of the Model

Our model is designed for a system that spans multiple administrative domains where each domain can impose its own security policies. It is still necessary that a common authentication mechanism be supported between two communicating systems. The model we present enables the syntactic specification of multiple authentication policies, but it does not translate between heterogeneous authentication mechanisms.

4.1. The Extended Access Control List (EACL) framework

EACLs extend the conventional ACL concept by using conditional authorization as an extension to authorization policies, implemented as restrictions (or conditions, we use these words interchangeably) on authentication and authorization credentials. An EACL is associated with an object and lists principals allowed to access this object and the type of access granted.

The objects to be protected in PRM are hosts, but our model is suitable for applications in which the objects are files, physical devices like printers or faxes etc.

4.1.1. Notation

We will use the Backus-Naur Form to denote the elements of our EACL language. Square brackets, [ ], denote optional items and curly brackets, {}, surround items that can repeat zero or more times. A vertical line, |, separates alternatives. Items inside single quotes are the terminal symbols. The wild-card symbol `"*"` is used in an EACL just as in the UNIX environment.

4.1.2. EACL : Specification Format

An EACL consists of a set of EACL entries. Each EACL entry represents access control policies directly associated with a particular principal entity. An EACL entry specifies a principal or a list of principals, a set of granted and/or denied access rights, and optionally, any associated conditions.

```
eacl_entry::= principal {principal}
  access_rights {condition}
  {access_rights {condition}} ';'
```

4.1.3. Specification of Principals

The principal is specified according to the following format:

```
principal::=
 principal_type sec_mech  principal_ID |
 'ANYBODY'
principal_type::='HOST' | 'USER' |
 'GROUP' | 'APPLICATION'
```

where `sec_mech` and `principal_ID` are alphanumerical strings.

Different administrative domains might use different authentication mechanisms, each having a particular syntax for specification of principals. For example, an application may use Kerberos V5 [9] as an authentication service. Kerberos V5 provides secret-key based authentication and the format of the Kerberos V5 principal name is `user_name/instance@realm`. Other domains may use DCE to obtain the user's identity credentials, usually identified by a User ID and Group ID. Another domain might use client authentication in SSL, based on public-key cryptography, where principals are identified by a global name, syntactically tied to the X.500 directory. In our model, the syntax of `principal_ID` is defined according to the underlying `sec_mech`, but is tagged to identify the name space.

Principals can be aggregated into a single entry when the same set of access rights and conditions applies to all of them. `ANYBODY` is used to represent all principals regardless of authentication. Examples of principal entities are:

```
ANYBODY
USER KERBEROS.V5 kot@ISI.EDU
HOST IPaddress 164.67.21.82
GROUP DCE 8
APPLICATION CHECKSUM 0x75AA31
```

4.1.4 Specification of Access rights

Access rights are specified using the format:

```
access_rights::='<' tag ':' ['-'] value
 { tag ':' ['-']value } | '*' '>'
```

where `tag` and `value` are alphanumerical strings.

Access rights are names for types of access to the protected object. All operations defined on the object are grouped by type of access to the object they represent, and named using a tag. The right is granted when there is no "-" preceding the right specification, otherwise the right is denied. The meaning of access control rights is application specific.

4.1.5. Specification of Conditions

Conditions specify the type-specific policies under which an operation can be performed on an object. The format used for specifying access rights conditions is as follows:

```
condition::= type ':' value
```

where `type` and `value` are alphanumerical strings.

A condition is interpreted according to its type. Conditions can be categorized as generic or specific. A condition is generic if it is interpreted by the GAA API. For example: time of day, authentication mechanism, required endorsement. Specific conditions are interpreted by the application: CPU load , memory usage, applications that are to be loaded on the node.

4.1.6. EACL evaluation

The authorization language we presented supports authorization models based on the closed world model, when all rights are implicitly denied. Authorizations are granted by an explicit listing of positive access rights. The open world model, which is based on implicit granting of all rights and listing of only negative authorizations, can be represented in our model by including `ANYBODY *` as the final entry in an EACL. This will grant everybody all rights regardless of authentication. Denial of rights is then specified using negative rights in entries earlier in the ACL.

If one allows both negative and positive authorizations in individual or group entries, inconsistencies must be resolved according to different resolution rules. The design approach we adopted allows the ordered interpretation of EACLs. An ordered evaluation approach is easier to implement, it allows only partial evaluation of EACL and resolves the authorization conflicts. Evaluation of ordered EACL starts from the first to the last in the list of EACL entries. The resolution of inconsistent authorization is based on ordering: the authorizations or denials that have already been examined take precedence over later ones. Other interpretations are possible, but we found that for many such policies, resolution of inconsistencies was either NP-Complete or undecidable.

4.2. GAA API

The GAA API is used by applications to decide whether the subject is authorized for access. In this subsection we provide a brief description of the GAA API routines.

4.2.1. GAA API functions

1) `gaa_get_object_eacl`

This function is called before other GAA API routines which require a handle to the object EACL on which to operate. It returns a handle to the object EACL.

2) `gaa_check_authorization`

This function tells the application server whether the requested operation is authorized, or if additional application-specific checks are required. It returns the code `YES` (indicating authorization) if all requested operations are authorized, `NO` (indicating denial of authorization) if at least one operation is not authorized, `MAYBE` (indicating need for application-specific checks) if there are some unevaluated conditions and additional application-specific checks are required. A list of conditions is also returned, each condition being marked as evaluated or not evaluated, and if evaluated, marked as met or not met. The time period during which the authorization is granted is returned as a condition that may be used by the application.

If no operation was specified as an input, a list of authorized rights is returned as a condition that must be checked by the application. This allows the application to discover access control policies associated with the target object. The application must understand the conditions that are returned unevaluated, otherwise it rejects the request. If the application understands the conditions, it checks them against the information about the request, the target object, or other environmental conditions to determine whether the conditions have been met.

4.2.2 GAA API Security Context

The security context is an argument passed to the GAA API. Some of its constituents follow:

Identity Verified authentication information, such as principal ID for a particular security mechanism. To determine which entries apply, the GAA API checks if the specified principal ID appears in an EACL entry that is paired with a privilege for the type of access requested.

Authorization Attributes Verified authorization credentials, such as group membership, group non-membership and proxies.

Delegated Credentials Delegation is supported through inclusion of delegated credentials, such as those supported by *restricted proxies* [6].

Evaluation and Retrieval Functions for Upcalls These functions are called to evaluate application-specific conditions; request additional credentials and verify them.

5. Applying the Distributed Authorization Model to PRM

We will first discuss the integration of the EACL framework into PRM and then we will show how PRM makes use of the GAA API to enforce the policies expressed through the EACL.

5.1. EACL conditions specific to PRM

Our experience with deploying PRM on a wide scale has shown that administrators are more willing to grant access to their workstations if they can restrict access to users or organizations they trust. Administrators must also be able to specify restrictions on the specific applications that will run on their systems. These restrictions are important in the context of movement of executable or interpreted content between different systems and platforms, i.e. what is usually known as "mobile code". We have therefore introduced EACL conditions specific to this type of policy:
- name of application:

```
application_name : matlab
```

- name of interpreter, in case the application is written in an interpreted language:

```
interpreter_name : Tcl
```

- platform the application runs on:

```
application_platform : Solaris
```

- version number for the application:

```
application_version : 1.0
```

- endorser or certifying authority for the application:

```
application_endorser : Globus
```

Authorizing a user to run an application on the specific resources is often not detailed enough for system administrators. What is needed is a way to impose and enforce limits on the physical resources consumed by the applications. To specify these limits, PRM uses specific EACL conditions:
- CPU load, expressed as maximum percentage of the CPU time that an application is allowed to use:

```
cpu_load : 20%
```

- memory usage, expressed as maximum size in Kbytes that a process can occupy in main memory:

```
mem_usage : 1024
```

- machine idle time, expressed as minimum interval in minutes that the machine has to be idle before any application managed by PRM is allowed to run:

```
idle_time : 30
```

5.2. Using the GAA API in PRM

5.2.1. Creation of the GAA API security context

For communications, PRM uses calls to the Asynchronous Reliable Delivery Protocol (ARDP), which handles several security services including authentication, integrity and payment. ARDP calls the Kerberos library through a security API, requesting the principal's identity, which is placed into the security context and is passed to the GAA API. Figure 1 shows the flow of control: the system manager calls ARDP requesting the principal's identity (1); the request and the verification of the principal's identity credentials take place (2, 3, 4, 5); ARDP places the principal's authentication credentials in the security context (6a) and returns it to the system manager (6); the system manager calls the GAA API (7); the security context, containing the verified principal's identity is being passed into the GAA API (7a).

When additional security attributes are required for the requested operation, the list of required attributes is returned and obtained by the application. The application or transport may add an upcall function to the security context which is passed to the GAA API and used to request required additional credentials. Such additional credentials are requested, verified, and added to the security context by this upcall function.

5.2.2. Authorization Walk-through

Here we present two authorization scenarios. First, let's consider a request from user Joe to run `matlab` on the host `kot.isi.edu` on Monday at 7:30 PM. Assume that this host has the following ordered EACL stored in the Prospero directory service:

```
USER  kerberos.v5  joe@ISI.EDU
    <HOST : load > time_window : 6AM-8PM,
                   cpu_load : 20% ;
GROUP kerberos.v5  operator@ISI.EDU
```

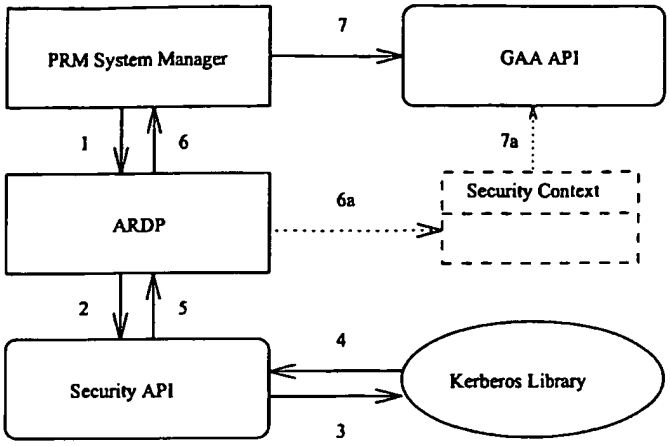


Figure 1. Creation of the GAA API security context

```
    USER  kerberos.v5  tom@ISI.EDU
    <HOST : * > <DEVICE : power_down> ;
ANYBODY <HOST : load> time_day : sat-sun,
                  time_window : 6AM-8PM,
                  cpu_load : 10% ;
```

When a job manager contacts a system manager with the request for resources, the system manager calls the `gaa_get_object_eacl` function to obtain a handle to the EACL of `kot.isi.edu`. The upcall function for retrieving the EACL for the specified object from the Prospero virtual file system is passed to the GAA API and is called by `gaa_get_object_eacl`, which returns the EACL handle. The system manager calls ARDP, which handles authentication as explained in Figure 1 and section 5.2.1. If Joe is authenticated successfully, then the verified identity credential is placed into the security context, specifying Joe as the Kerberos principal `joe@ISI.EDU`.

The `gaa_check_authorization` function is called by the system manager, which asks if Joe is authorized to load `matlab` to `kot.isi.edu`. In evaluating the EACL, the first entry applies. It grants the requested operation, but there two conditions that must be evaluated. The first condition `time_window : 6AM-8PM` is generic and is evaluated directly by the GAA API. Since the request was issued on Monday at 7:30 PM this condition is satisfied. The second condition `cpu_load : 20%` is PRM-specific. If the security context passed by PRM defined a condition evaluation function for upcall, then this function is invoked and if this condition is met then the final answer is YES (authorized).

During the execution of the task the node manager is monitoring if the task is abiding to the limits imposed on the local resources and authorization time. If the corresponding upcall function was not passed to the GAA API, the answer is MAYBE and the set of conditions is returned. Conditions are marked as either evaluated or not evaluated. In our example `time_window : 6AM-8PM` was evaluated and met; `cpu_load : 20%` was not evaluated and should be checked by PRM.

Next, we present an authorization scenario where additional credentials are needed. Let's consider a request from user Joe to run `matlab` on the host `kot.isi.edu` on Monday at 8:30 PM. In EACL evaluation, the first entry applies but does not grant this operation, since the first condition is not met. The temporary answer is NO (not authorized). The second entry grants this permission. If the security context defines a credential retrieval function for upcall, then this function is invoked and if either a group "operator" membership credential or delegated credential from user Tom for Joe is obtained, then the final answer is YES. If the credential retrieval upcall function was not passed to the GAA API, the answer is NO.

6. Managing the EACL using the Prospero Directory Service

We have mentioned in section 2 that PRM deals with scalability issues by splitting the task of managing the resources across the three types of managers. Our goal in designing a mechanism for the management of the EACL files was to enable easy sharing of a default authorization policy among node managers, while allowing customization of the policy at the level of individual hosts.

We use the Prospero Directory Service [7] to store the information associated with the EACL files. The EACL files themselves are objects stored in the Prospero directory service.

The following scenario shows how the management of

the files is accomplished:

1. The administrator of the domain whose resources are managed by a system manager running on host A creates an EACL file describing the default authorization policy which applies to the domain.

2. The administrator registers with the Prospero server. We supply a script which takes as input the location of the EACL file and creates a Prospero object representing a link to the file, together with two attributes for the link:

`SYSTEM_MANAGER A`

`EACL_DEFAULT True`

3. If the administrator of a particular host B in the domain managed by A wants to specify a local authorization policy different from the default one, a similar procedure is followed, except that the link to the local EACL file is created with the following attributes:

`NODE_MANAGER B`

`EXTEND_DEFAULT Prepend/Append/Replace`

(`Prepend` if the local policy is prepended to the default policy, `Append` if the local policy is appended to the default and `Replace` if the local policy completely replaces the default)

4. When a system manager is contacted by a job manager with a request for resources, it first authenticates the user, as was explained in the authorization scenario in section 5. Before requesting resources from a node manager running on a particular node B, the system manager retrieves the EACL file associated with that node by looking for a link with attribute `NODE_MANAGER = B`. If no such link is found, the default EACL file provided for the domain will be used and it will be obtained by retrieving a link with attributes `SYSTEM_MANAGER = A` and `EACL_DEFAULT = True`. If a link with `NODE_MANAGER = B` is found, then a second query is issued for the value of the attribute `EXTEND_DEFAULT`. If the value is `Prepend` or `Append`, the system manager will have to retrieve the default EACL file first, and then prepend or append to it the contents of the EACL file for node B (note that the distinction between the two cases `Prepend/Append` is necessary because the EACL evaluation takes into account the order of the EACL entries). If the value is `Replace`, then only the EACL file for node B will be retrieved and used.

5. After retrieval of the EACL file, evaluation of the conditions listed in the file follows, as detailed in the authorization scenario from section 5. If all the conditions are met, the job manager is allowed to use the resources on that particular host.

6. During the execution of tasks on a particular host, the node manager periodically checks whether the task is abiding to the limits imposed on the local resources. If it is not, then the task is interrupted and the job manager is notified.

7. Related Work

Nagaratnam and Byrne [5] present a model for Internet user agents to control access to client resources. This model protects client machines from hostile downloadable content and allows the client to selectively grant access to trusted agents. The authenticity of the code is based on digital signatures of principals certifying it. All access control requests are mediated by calling a security manager component and decisions are based on the user's access control specifications stored in the policy database.

The model is restricted to using the Javakey utility as an authentication mechanism based on public key digital signatures, while our model is general enough to use a variety of security mechanisms based on public or secret key cryptosystems.

Another disadvantage of that model is the duplication of common information. Each user has to maintain a database of any principals specified in the policy database and their public keys, as well as specification of groups. These databases should be properly integrity-protected. In contrast, PRM uses Kerberos to achieve strong authentication. The authentication database is maintained centrally by the KDC and stored on a physically secure machine. Our model also supports a group certification mechanism. A group server maintains and provides group membership information, and issues group membership and non-membership certificates. The certificates are placed into the GAA API security context and checked by the GAA API when making authorization decisions. There is no need for each user to maintain authentication and group specification databases locally.

The Generalized Access Control List framework described by Woo and Lam [10] presents a language-based approach for specifying authorization policies. The GACL model supports only system state-related conditions within which rights are granted, such as current system load and maximum number of copies of a program to be run concurrently. This may not be sufficient for distributed applications. Our model allows fine-grained control over the conditions.

Both restricted proxies [6] and the use-condition model [4] allow conditions and privilege attributes to be embedded in authorization credentials or certificates. These mechanisms can be readily integrated with the authorization model presented here: the restrictions or conditions caried in the proxy or certificate are evaluated by the GAA API in addition to the restrictions in the matching EACL entry.

The CRISIS architecture [1] provides ACLs that are related to the type of the protected object. For example, file ACLs list principals allowed read, write or execute access to the file, whereas node ACLs contain principals allowed to run jobs on the node. CRISIS ACLs do not support con-

straints on the resources that principals are allowed to consume. The emphasis of our work is on providing a general framework for representing security policies and facilitating authorization decisions for applications. Our model provides a uniform authorization mechanism that is capable of supporting different operations and different kinds of protected objects.

The Tivoli Management Environment (TME 10) is a commercially available system from IBM which takes a role-based approach to security [3]. TME roles are named capabilities, containing a list of objects and access permissions to those objects. Objects can have default access and can be associated with more than one role. Each role will have a different level of access to the object. Roles are defined to support a particular job function within an organization, e.g. customer support or management. Groups are assigned roles, thus giving members of those groups access capabilities to the objects assigned to those roles. The TME security model can be easily expressed by our EACL framework:

1) An EACL is associated with each object to be protected. Default access to the object is represented by including `ANYBODY default_rights` as the last entry in the EACL.

2) The object's EACL will contain entries listing groups and a set of access rights, granted by TME roles assigned to each group.

For example, in TME the group `Support_users` is assigned the `Customer_support` role which grants RWE permissions to file `/cust_supp_dir/*` In our system, an EACL associated with the object `/cust_supp_dir/*` will have the following entry:

```
GROUP sec_mech support_users FILE:R
FILE:W FILE:E ;
```

TME lacks flexibility in supporting user-defined security policies. It has a fixed predefined set of object types and generic access permissions that are available on each object type. In addition, the TME model requires the creation of a new role to include each new combination of objects and access rights. This becomes very cumbersome for systems where a large number of operations exist on various objects.

8. Conclusions

By extending the traditional Access Control Lists with restrictions on authorized rights, and by using General Authorization and General Authorization and Access API, it becomes possible to support a flexible distributed authorization mechanism allowing applications and users to define their own access control policy types, and integrating local and distributed security policies. The problem of translation of the policies is addressed by using general or PRM-specific evaluation functions. In this paper, we have omitted discussion of many practical details due to space limitation. A prototype of the presented model has been developed at the Information Sciences Institute of the University of Southern California.

Acknowledgments

This research was supported in part by the Defense Advanced Research Projects Agency under the Scalable Computing Infrastructure (SCOPE) Project, TNT, Contract No. DABT63-95-C-0095, Security Infrastructure for Large Distributed Systems (SILDS) Project, Contract No. DABT63-94-C-0034, and by a grant from Xerox Corporation. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Intelligence Center and Fort Huachuca Directorate of Contracting, the Defense Advanced Research Projects Agency, the U.S. Government, or Xerox Corporation.

References

[1] E. Belani, A. Vahdat, T. Anderson, and M. Dahlin. The CRISIS wide area security architecture. *Proceedings of the 7th USENIX Security Symposium, San Antonio, Texas*, January 1998.

[2] I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *International Journal of Supercomputer Applications*, Summer 1997.

[3] IBM. TME 10 security management. http://www.tivoli.com/redbooks/html/sg242021/2021fm.htm, October 1997.

[4] W. Johnston and C. Larsen. A use-condition centered approach to authenticated global capabilities: Security architectures for large-scale distributed collaboratory environments. LBNL Report 38850.

[5] N. Nagaratnam and S. Byrne. Resource access control for an Internet user agent. *Proceedings of the third USENIX Conference on Object-Oriented Technologies and Systems, Portland, Oregon*, June 1997.

[6] B. C. Neuman. Proxy-based authorization and accounting for distributed systems. *Proceedings of the 13th International Conference on Distributed Computing Systems, Pittsburgh*, May 1993.

[7] B. C. Neuman, S. Augart, and S. Upasani. Using Prospero to support integrated location-independent computing. *Proc. Symp.on Mobile and Location-Independent Computing, Cambridge, MA*, pages 29–34, August 1993.

[8] B. C. Neuman and S. Rao. The Prospero Resource Manager: A scalable framework for processor allocation in distributed systems. *Concurrency: Practice and Experience*, June 1994.

[9] B. C. Neuman and T. Ts'o. Kerberos: An authentication service for computer networks. *IEEE Communications Magazine*, pages 33–38, September 1994.

[10] T. Woo and S. Lam. A framework for distributed authorization. *Proc. ACM Conference on Computer and Communications Security, Fairfax, Virginia*, November 1993.