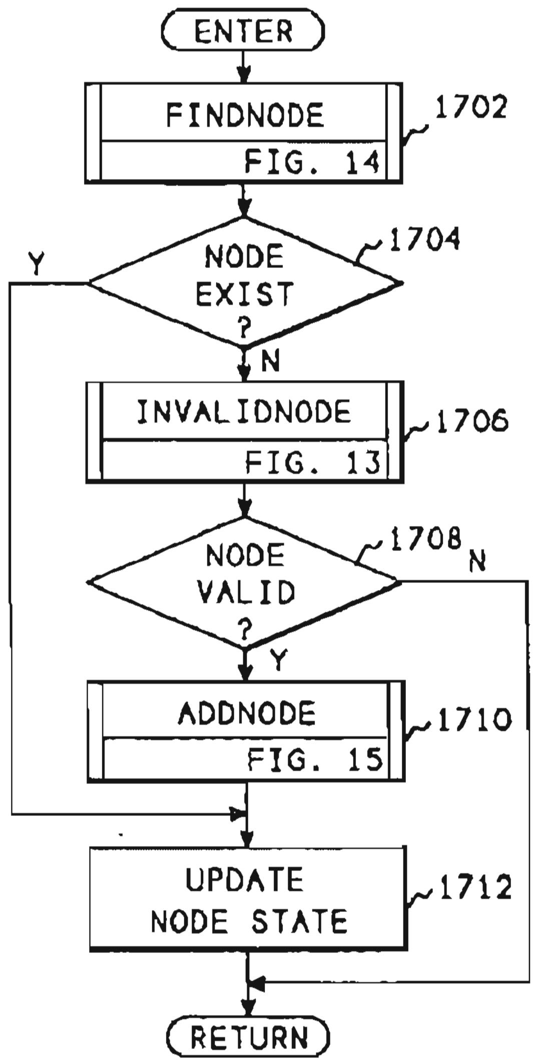

ENTER
FINDNODE
FIG. 14
1702
NODE
EXIST
?
1704
Y
N
INVALIDNODE
FIG. 13
1706
NODE
VALID
?
1708
N
Y
ADDNODE
FIG. 15
1710
UPDATE
NODE STATE
1712
RETURN

FIG. 17

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 1 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	1-1	Civ Action No. 06-2469 Appendix A (List of Prior Art References) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01618 -1657)	
	1-2	Civ Action No. 06-2469 Appendix B (Invalidty Claim Chart) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01658-1693)	
	1-3	Civ Action No. 06-2469 Appendix C (Obviousness Combinations Chart) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01694-01716)	
	1-4	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 1 of 5) (N2PIDS_00457-506)	
	1-5	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 2 of 5) (N2PIDS_00507-556)	
	1-6	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 3 of 5) (N2PIDS_00557-606)	
	1-7	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 4 of 5) (N2PIDS_00607-656)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 2 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	2-1	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 5 of 5) (N2PIDS_00657-733)	
	2-2	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 1 of 3) (N2PIDS_00734-768)	
	2-3	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 2 of 3) (N2PIDS_00769-802)	
	2-4	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 3 of 3) (N2PIDS_00803-844)	
	2-5	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Responsive Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (N2PIDS_00412-456)	
	2-6	Civ Action No. 06-2469 Declaration of David B. Johnson in Support of Skype's Responsive Claim Construction Brief (N2PIDS_00845-913)	
	2-7	Civ Action No. 06-2469 Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 Directed to Defendents Ebay Skype Tech and Skype Inc (N2PIDS_00387-411)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 3 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	3-1	Civ Action No. 06-2469 Joint Final Pretrial Order (part 1 of 8) (N2PIDS_00914-963)	
	3-2	Civ Action No. 06-2469 Joint Final Pretrial Order (part 2 of 8) (N2PIDS_00964-1013)	
	3-3	Civ Action No. 06-2469 Joint Final Pretrial Order (part 3 of 8) (N2PIDS_01014-1063)	
	3-4	Civ Action No. 06-2469 Joint Final Pretrial Order (part 4 of 8) (N2PIDS_01064-1113)	
	3-5	Civ Action No. 06-2469 Joint Final Pretrial Order (part 5 of 8) (N2PIDS_01114-1163)	
	3-6	Civ Action No. 06-2469 Joint Final Pretrial Order (part 6 of 8) (N2PIDS_01164-1213)	
	3-7	Civ Action No. 06-2469 Joint Final Pretrial Order (part 7 of 8) (N2PIDS_01214-1263)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 4 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	4-1	Civ Action No. 06-2469 Joint Final Pretrial Order (part 8 of 8) (N2PIDS_01264-1310)	
	4-2	Civ Action No. 06-2469 Opening Claim Construction Brief of Skype Tech Skype Inc. and Ebay (N2PIDS_01741-1790)	
	4-3	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 1 of 3) (N2PIDS_01311-1393)	
	4-4	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 2 of 3) (N2PIDS_01394-1451)	
	4-5	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 3 of 3) (N2PIDS_01452-1490)	
	4-6	Civ Action No. 06-2469 Plaintiff Net2Phone's Response Brief on Claim Construction (part 1 of 2) (N2PIDS_01491-1546)	
	4-7	Civ Action No. 06-2469 Plaintiff Net2Phone's Response Brief on Claim Construction (part 2 of 2) (N2PIDS_01547-1617)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided. EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 5 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	5-1	Civ Action No. 06-2469 Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (N2PIDS_01717-1740)	
	5-2	Civ Action No. 06-2469 Responsive Claim Construction Brief of Skype Tech Skype Inc. and Ebay (N2PIDS_01791-1825)	
	5-3	U.S. Control No. 90/010,423 - 2009-08-05 PTO Office Action	
	5-4		
	5-5		
	5-6		
	5-7		

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

Electronic Acknowledgement Receipt

EFS ID:	5870450
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael Raymond Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	11-AUG-2009
Filing Date:	17-FEB-2009
Time Stamp:	19:58:38
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal Letter	26550188Xmittal.pdf	268365 ac46e633e14c8aca04098d83c913d8d623d3bb88	no	2

Warnings:

Information:

2	Information Disclosure Statement (IDS) Filed (SB/08)	26550188_1449.pdf	794095 2b89e3d3ec63e9ecc4b98073ec81b4a36bb50ac6	no	5
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	Reexam Certificate of Service	20090811COS.pdf	58475 3e0d0b295f06538e5b37ee165c7654d83165ba4a	no	1
Warnings:					
Information:					
4	NPL Documents	N2PIDS_00387.pdf	1248303 103f829551fe55b548a541d669beda6cad272089	no	25
Warnings:					
Information:					
5	NPL Documents	N2PIDS_00412.pdf	3962848 1af848ed034593c08abb73fe5d189f1e6f7f5298	no	45
Warnings:					
Information:					
6	NPL Documents	N2PIDS_00457.pdf	2698601 ce8f9d9f9808e5c59d3b34fab8c77ca89aeb825f	no	50
Warnings:					
Information:					
7	NPL Documents	N2PIDS_00507.pdf	1838285 37c19558ce807b15bbead7b7636c9d0f18f07a6f	no	50
Warnings:					
Information:					
8	NPL Documents	N2PIDS_00557.pdf	1953095 3a1ccc92e0a0e581163917ecdc9acc66de6c098f	no	50
Warnings:					
Information:					
9	NPL Documents	N2PIDS_00607.pdf	3476450 b193aab143feb3aa818b3e418060396134387793	no	50
Warnings:					
Information:					
10	NPL Documents	N2PIDS_00657.pdf	3241310 70cbfb7c1eb6c50d82b67074772470e507b91cde	no	77

Warnings:					
Information:					
11	NPL Documents	N2PIDS_00734.pdf	3726939 d4c9f933c002ebde9c954298814937c980f65660	no	35
Warnings:					
Information:					
12	NPL Documents	N2PIDS_00769.pdf	4202699 b4c92116877442bcd04bd6debf7d14595fb9a731	no	34
Warnings:					
Information:					
13	NPL Documents	N2PIDS_00803.pdf	4152068 7b3a2e2ddcdeffd164f8a1cfc96d610e8b846617	no	42
Warnings:					
Information:					
14	NPL Documents	N2PIDS_00845.pdf	4330148 c43225536115 8b4e81c3b1bb0d51e0574242a578	no	69
Warnings:					
Information:					
15	NPL Documents	N2PIDS_00914.pdf	2944261 080bb981dd8437b6157018b86fc3befc6ea75db8	no	50
Warnings:					
Information:					
16	NPL Documents	N2PIDS_00964.pdf	4490177 7d1ee36dc21fea4ded879265154bb7945a02315d	no	50
Warnings:					
Information:					
17	NPL Documents	N2PIDS_01014.pdf	4260549 3cdd0ff0c98cae3e3cae281b22424495215e177f	no	50
Warnings:					
Information:					
18	NPL Documents	N2PIDS_01064.pdf	3030407 4ac0890d3df0b3d2e76bc806eea54f876a6dad9c	no	50
Warnings:					
Information:					
19	NPL Documents	N2PIDS_01114.pdf	2797371	no	50

Warnings:					
Information:					
20	NPL Documents	N2PIDS_01164.pdf	2757821 7b7842735401ba137dd0536afd9e0833119 a0e19	no	50
Warnings:					
Information:					
21	NPL Documents	N2PIDS_01214.pdf	2751710 ef893e52560ecd5e62dcd743fb915147bde 65a5b	no	50
Warnings:					
Information:					
22	NPL Documents	N2PIDS_01264.pdf	2069106 df95a3266f4b04540ff5230cbf06f520e2a80 bd1	no	47
Warnings:					
Information:					
23	NPL Documents	N2PIDS_01311.pdf	4119057 b220eddf06eed135dc679c8358b50fad44b 347d2	no	83
Warnings:					
Information:					
24	NPL Documents	N2PIDS_01394.pdf	3968881 4af783e9b0d04ff379bba2fc7dfcfa52b9b1d 311	no	58
Warnings:					
Information:					
25	NPL Documents	N2PIDS_01452.pdf	1920359 3b013b03ee0d9b44a7aad1c6e1d3b56344 860c7d	no	39
Warnings:					
Information:					
26	NPL Documents	N2PIDS_01491.pdf	3401591 364aed57efd1a9a5a36c818e1e65dc90a11 abd54	no	56
Warnings:					
Information:					
27	NPL Documents	N2PIDS_01547.pdf	4441250 15d16abc2d22fc670e1b3af1890142dbe41 d6c87	no	71
Warnings:					
Information:					
28	NPL Documents	N2PIDS_01618.pdf	2566762	no	40

Warnings:					
Information:					
29	NPL Documents	N2PIDS_01658.pdf	2270928 14e4e22968011b9eda59be02c0cc42a7d30cbef4	no	36
Warnings:					
Information:					
30	NPL Documents	N2PIDS_01694.pdf	1540394 6ec7be7e68a603a25c0fafd57a479064eac8d23f	no	23
Warnings:					
Information:					
31	NPL Documents	N2PIDS_01717.pdf	1150239 05ae46089b0835e058ab04f346c73aa6ea6ca128	no	24
Warnings:					
Information:					
32	NPL Documents	N2PIDS_01741.pdf	2620699 8878a7f238582ef777bb48052fa32342df91bc16	no	50
Warnings:					
Information:					
33	NPL Documents	N2PIDS_01791.pdf	1800489 59e0d76f3e3f44657ef083de4d41a22e7ff852f2	no	35
Warnings:					
Information:					
			Total Files Size (in bytes):	90853732	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION OF:

Net2Phone, Inc.

Control No.: 90/010,416

Issue Date: August 22, 2000

Title: **POINT-TO-POINT INTERNET PROTOCOL**

Attorney Docket: 2655-0188

Group Art Unit: 3992

Examiner: KOSOWSKI, Alexander J.

Date: August 11, 2009

Confirmation No.: 1061

INFORMATION DISCLOSURE STATEMENT

Hon. Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Pursuant to 37 C.F.R. § 1.56, the attention of the Patent and Trademark Office is hereby directed to the reference(s) listed on the attached PTO-1449. One copy of each non-U.S. Patent reference is attached. It is respectfully requested that the information be expressly considered during the prosecution of this application, and that the reference(s) be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

The submission of any document herewith, which is not a statutory bar, is not intended that any such document constitutes prior art against any of the claims of the present application or is considered to be material to patentability as defined in 37 C.F.R. § 1.56(b). Applicants do not waive any rights to take any action which would be appropriate to antedate or otherwise remove as a competent reference against the claims of the present application.

☒ The Examiner's attention is directed to co-pending U.S. Patent Control Nos. 90/010,424, 90/010,421, 90/010,422 and 90/010,423 which are involved in the same litigation as the patent corresponding to the present re-examination. The identification of this U.S. Patent Application is not to be construed as a waiver of secrecy as to that application now or upon issuance of the present application as a patent. The Examiner is respectfully requested to consider the cited application and the art cited therein during examination.

☐ Copies of the references were cited by or submitted to the Office in parent Application No. _____, filed _____, which is relied upon for an earlier filing date under 35 U.S.C. 120. Thus, Form PTO 1449 is attached without copies of these references. 37 C.F.R. § 1.98(d).

CHARGE STATEMENT: Deposit Account No. 501860, order no. **2655-0188**.

The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (missing or insufficiencies only) now or hereafter relative to this application and the resulting Official Document under Rule 20, or credit any overpayment, to our Accounting/Order Nos. shown above, for which purpose a duplicate copy of this sheet is attached

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal sheet is filed.

CUSTOMER NUMBER
42624

Davidson Berquist Jackson & Gowdey LLP
4300 Wilson Blvd., 7th Floor,
Arlington Virginia 22203
Main: (703) 894-6400 • FAX: (703) 894-6430

Respectfully submitted,

By: /Michael R. Casey /

Michael R. Casey, Ph.D. (Reg. No.: 40,294)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 1 of 1		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	1-1	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 1 of 3) (N2PIDS_02100-2166)	
	1-2	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 2 of 3) (N2PIDS_02166-2232)	
	1-3	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 3 of 3) (N2PIDS_02233-2292)	
	1-4		
	1-5		
	1-6		
	1-7		

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided. EA = English Abtract, T = Translation, PF = Patent Family.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that, on August 12, 2009, the Information Disclosure Statements (with references in electronic format, as agreed by requestor) filed in Reexam Control Numbers:

1) 90/010,422;

2) 90/010,424;

3) 90/010,421;

4) 90/010,416; and

5) 90/010,423

were served by First Class Mail, on Requestor:

Blakely, Sokoloff, Taylor & Zafman LLP
1279 Oakmead Parkway
Sunnyvale, CA 94085-4040

Michael R. Casey, Ph.D.

Electronic Acknowledgement Receipt

EFS ID:	5874408
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael Raymond Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	12-AUG-2009
Filing Date:	17-FEB-2009
Time Stamp:	14:28:47
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal Letter	N2PIDS_02072.pdf	252336 69c2ba3d6a84b45b625fef0888a6ac4c9233a46a	no	2

Warnings:

Information:

2	Information Disclosure Statement (IDS) Filed (SB/08)	N2PIDS_02074.pdf	134023 c188e2b31d80239bce01a98ff8ceefa047fcc338	no	1
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	NPL Documents	N2PIDS_02100.pdf	7508718 5746764fc5941ad1219fab6873a3f311ed436bf0	no	66
Warnings:					
Information:					
4	NPL Documents	N2PIDS_02166.pdf	6806160 e5f84cd3022bf393d6052237c341fc4a78616503	no	67
Warnings:					
Information:					
5	NPL Documents	N2PIDS_02233.pdf	6987289 25ba8ceecec040367e21e1fdfb5f5a6fcd50c600	no	60
Warnings:					
Information:					
6	Reexam Certificate of Service	N2PIDS_02078_COS.pdf	21706 1a8b052cea43fd90e6f3de54729dca4e47d2bd7a	no	1
Warnings:					
Information:					
		Total Files Size (in bytes):	21710232		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION OF:
Net2Phone, Inc.
Control No.: 90/010,416
Issue Date: August 22, 2000
Title: **POINT-TO-POINT INTERNET PROTOCOL**

Attorney Docket: 2655-0188
Group Art Unit: 3992
Examiner: KOSOWSKI, Alexander J.
Date: August 12, 2009
Confirmation No.: 1061

INFORMATION DISCLOSURE STATEMENT

Hon. Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Pursuant to 37 C.F.R. § 1.56, the attention of the Patent and Trademark Office is hereby directed to the reference(s) listed on the attached PTO-1449. One copy of each non-U.S. Patent reference is attached. It is respectfully requested that the information be expressly considered during the prosecution of this application, and that the reference(s) be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

The submission of any document herewith, which is not a statutory bar, is not intended that any such document constitutes prior art against any of the claims of the present application or is considered to be material to patentability as defined in 37 C.F.R. § 1.56(b). Applicants do not waive any rights to take any action which would be appropriate to antedate or otherwise remove as a competent reference against the claims of the present application.

☐ The Examiner's attention is directed to co-pending U.S. Patent Control Nos. 90/010,424, 90/010,421, 90/010,422 and 90/010,423 which are involved in the same litigation as the patent corresponding to the present re-examination. The identification of this U.S. Patent Application is not to be construed as a waiver of secrecy as to that application now or upon issuance of the present application as a patent. The Examiner is respectfully requested to consider the cited application and the art cited therein during examination.

CHARGE STATEMENT: Deposit Account No. 501860, order no. **2655-0188.**

The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (missing or insufficiencies only) now or hereafter relative to this application and the resulting Official Document under Rule 20, or credit any overpayment, to our Accounting/Order Nos. shown above, for which purpose a duplicate copy of this sheet is attached

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal sheet is filed.

CUSTOMER NUMBER

42624

Davidson Berquist Jackson & Gowdey LLP
4300 Wilson Blvd., 7th Floor,
Arlington Virginia 22203
Main: (703) 894-6400 • FAX: (703) 894-6430

Respectfully submitted,

By: /Michael R. Casey /

Michael R. Casey, Ph.D. (Reg. No.: 40,294)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 1 of 1		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	1-1	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 1 of 3) (N2PIDS_02100-2166)	
	1-2	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 2 of 3) (N2PIDS_02166-2232)	
	1-3	Civ Action No. 06-2469 Plaintiff Net2Phone, Inc.'s Opening Claim Construction Brief (part 3 of 3) (N2PIDS_02233-2292)	
	1-4		
	1-5		
	1-6		
	1-7		

Examiner Signature	/Alexander Kosowski/	Date Considered	08/19/2009

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02074

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 1 of 5		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	1-1	Civ Action No. 06-2469 Appendix A (List of Prior Art References) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01618 -1657)	
	1-2	Civ Action No. 06-2469 Appendix B (Invalidty Claim Chart) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01658-1693)	
	1-3	Civ Action No. 06-2469 Appendix C (Obviousness Combinations Chart) to Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 (N2PIDS_01694-01716)	
	1-4	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 1 of 5) (N2PIDS_00457-506)	
	1-5	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 2 of 5) (N2PIDS_00507-556)	
	1-6	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 3 of 5) (N2PIDS_00557-606)	
	1-7	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 4 of 5) (N2PIDS_00607-656)	

Examiner Signature	/Alexander Kosowski/	Date Considered	08/19/2009

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02023

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 2 of 5	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	2-1	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Opening Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 5 of 5) (N2PIDS_00657-733)	
	2-2	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 1 of 3) (N2PIDS_00734-768)	
	2-3	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 2 of 3) (N2PIDS_00769-802)	
	2-4	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (part 3 of 3) (N2PIDS_00803-844)	
	2-5	Civ Action No. 06-2469 Declaration of Alan J. Heinrich in Support of Responsive Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (N2PIDS_00412-456)	
	2-6	Civ Action No. 06-2469 Declaration of David B. Johnson in Support of Skype's Responsive Claim Construction Brief (N2PIDS_00845-913)	
	2-7	Civ Action No. 06-2469 Defendants' Fourth Amended Responses to Plaintiff's Interrogatory Nos. 17-19 Directed to Defendents Ebay Skype Tech and Skype Inc (N2PIDS_00387-411)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02024

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 3 of 5	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	3-1	Civ Action No. 06-2469 Joint Final Pretrial Order (part 1 of 8) (N2PIDS_00914-963)	
	3-2	Civ Action No. 06-2469 Joint Final Pretrial Order (part 2 of 8) (N2PIDS_00964-1013)	
	3-3	Civ Action No. 06-2469 Joint Final Pretrial Order (part 3 of 8) (N2PIDS_01014-1063)	
	3-4	Civ Action No. 06-2469 Joint Final Pretrial Order (part 4 of 8) (N2PIDS_01064-1113)	
	3-5	Civ Action No. 06-2469 Joint Final Pretrial Order (part 5 of 8) (N2PIDS_01114-1163)	
	3-6	Civ Action No. 06-2469 Joint Final Pretrial Order (part 6 of 8) (N2PIDS_01164-1213)	
	3-7	Civ Action No. 06-2469 Joint Final Pretrial Order (part 7 of 8) (N2PIDS_01214-1263)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02025

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 4 of 5	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	4-1	Civ Action No. 06-2469 Joint Final Pretrial Order (part 8 of 8) (N2PIDS_01264-1310)	
	4-2	Civ Action No. 06-2469 Opening Claim Construction Brief of Skype Tech Skype Inc. and Ebay (N2PIDS_01741-1790)	
	4-3	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 1 of 3) (N2PIDS_01311-1393)	
	4-4	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 2 of 3) (N2PIDS_01394-1451)	
	4-5	Civ Action No. 06-2469 Plaintiff Net2Phone's Reply Brief on Claim Construction (part 3 of 3) (N2PIDS_01452-1490)	
	4-6	Civ Action No. 06-2469 Plaintiff Net2Phone's Response Brief on Claim Construction (part 1 of 2) (N2PIDS_01491-1546)	
	4-7	Civ Action No. 06-2469 Plaintiff Net2Phone's Response Brief on Claim Construction (part 2 of 2) (N2PIDS_01547-1617)	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02026

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 5 of 5	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	5-1	Civ Action No. 06-2469 Reply Claim Construction Brief of Skype Tech, Skype Inc. and Ebay (N2PIDS_01717-1740)	
	5-2	Civ Action No. 06-2469 Responsive Claim Construction Brief of Skype Tech Skype Inc. and Ebay (N2PIDS_01791-1825)	
	5-3	U.S. Control No. 90/010,423 - 2009-08-05 PTO Office Action	
	5-4		
	5-5		
	5-6		
	5-7		

Examiner Signature	/Alexander Kosowski/	Date Considered	08/19/2009

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

N2PIDS_02027

Reexamination	Application/Control No. 90010416	Applicant(s)/Patent Under Reexamination 6108704
	Certificate Date	**Certificate Number**

Requester Correspondence Address: ☐ **Patent Owner** ☒ **Third Party**

Blakely Sokoloff Taylor & Zafman LLP
1279 Oakmead Parkway
Sunnyvale, CA 94085-4040

LITIGATION REVIEW ☒	AJK (examiner initials)	08/25/2009 (date)
Case Name		Director Initials
OPEN: 2:06cv2469 Net2phone v. Ebay		Eric Keasel for GM

COPENDING OFFICE PROCEEDINGS	
TYPE OF PROCEEDING	**NUMBER**
1. no copending proceeding	

Search Notes	Application/Control No. 90010416	Applicant(s)/Patent Under Reexamination 6108704
	Examiner ALEXANDER J KOSOWSKI	Art Unit 3992

SEARCHED

Class	Subclass	Date	Examiner

SEARCH NOTES

Search Notes	Date	Examiner
Reviewed proposed prior art and prosecution history	8/25/09	AJK

INTERFERENCE SEARCH

Class	Subclass	Date	Examiner

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/010,416	02/17/2009	6108704	2655-0188	1061

42624 7590 08/27/2009

DAVIDSON BERQUIST JACKSON & GOWDEY LLP
4300 WILSON BLVD., 7TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 08/27/2009

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

DO NOT USE IN PALM PRINTER

(THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS)

Blakely Sokoloff Taylor & Zafman LLP

1279 Oakmead Parkway

Sunnyvale, CA 94085-4040

EX PARTE REEXAMINATION COMMUNICATION TRANSMITTAL FORM

REEXAMINATION CONTROL NO. 90/010,416.

PATENT NO. 6108704.

ART UNIT 3992.

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified *ex parte* reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the *ex parte* reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

Office Action in Ex Parte Reexamination	Control No. 90/010,416	Patent Under Reexamination 6108704	
	Examiner ALEXANDER J. KOSOWSKI	Art Unit 3992	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

a☒ Responsive to the communication(s) filed on *2/17/09*. b☐ This action is made FINAL.
c☒ A statement under 37 CFR 1.530 has not been received from the patent owner.

A shortened statutory period for response to this action is set to expire 2 month(s) from the mailing date of this letter.
Failure to respond within the period for response will result in termination of the proceeding and issuance of an *ex parte* reexamination certificate in accordance with this action. 37 CFR 1.550(d). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).**
If the period for response specified above is less than thirty (30) days, a response within the statutory minimum of thirty (30) days will be considered timely.

Part I THE FOLLOWING ATTACHMENT(S) ARE PART OF THIS ACTION:

1. ☐ Notice of References Cited by Examiner, PTO-892.
2. ☒ Information Disclosure Statement, PTO/SB/08.
3. ☐ Interview Summary, PTO-474.
4. ☐ ____

Part II SUMMARY OF ACTION

1a. ☒ Claims *1-7 and 10-44* are subject to reexamination.
1b. ☐ Claims *8 and 9* are not subject to reexamination.
2. ☐ Claims ___ have been canceled in the present reexamination proceeding.
3. ☐ Claims ___ are patentable and/or confirmed.
4. ☒ Claims *1-7 and 10-44* are rejected.
5. ☐ Claims ___ are objected to.
6. ☐ The drawings, filed on ___ are acceptable.
7. ☐ The proposed drawing correction, filed on ___ has been (7a) ☐ approved (7b)☐ disapproved.
8. ☐ Acknowledgment is made of the priority claim under 35 U.S.C. § 119(a)-(d) or (f).

   a)☐ All b)☐ Some* c)☐ None of the certified copies have

   1☐ been received.

   2☐ not been received.

   3☐ been filed in Application No. ____

   4☐ been filed in reexamination Control No. ____

   5☐ been received by the International Bureau in PCT application No. ____

   * See the attached detailed Office action for a list of the certified copies not received.

9. ☐ Since the proceeding appears to be in condition for issuance of an *ex parte* reexamination certificate except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte* Quayle, 1935 C.D. 11, 453 O.G. 213.
10. ☐ Other: ____

cc: Requester (if third party requester)

DETAILED ACTION

1) This Office action addresses claims 1-7 and 10-44 of United States Patent Number 6,108,704 (Hutton et al), for which it has been determined in the Order Granting Ex Parte Reexamination (hereafter the "Order") mailed 3/11/09 that a substantial new question of patentability was raised in the Request for *Ex Parte* reexamination filed on 2/17/09 (hereafter the "Request"). Claims 8-9 are not subject to reexamination.

IDS

2) With regard to the IDS's filed 8/11/09 and 8/12/09:

These IDS's have been given due consideration. However, that which are not either prior art patents or prior art printed publications have been crossed out so as not to appear reprinted on the front page of the patent.

Rejections

3) The following three rejections are utilized by the examiner below, referencing the proposed prior art listed on pages 5-6 of the Request:

Issue 1: Claims 1-7 and 10-44 in view of NetBIOS, RFC 1531, Pinard and VocalChat User's Guide.

Issue 2: Claims 1-7 and 10-44 in view of Etherphone, Vin, RFC 1531, NetBIOS, Pinard and VocalChat User's Guide.

Issue 3: Claims 1-7 and 10-44 in view of VocalChat, RFC 1531, NetBIOS and Pinard.

Claim Rejection Paragraphs

4) ***Claim Rejections - 35 USC § 102***

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Issue 1

5) Claims 43-44 are rejected under 35 U.S.C. 102(b) as being unpatentable by NetBIOS (See claim mapping chart in Exhibit M, pages 36-40, incorporated by reference).

6) Claims 1-7 and 32-42 are rejected under 35 U.S.C. 103(a) as being unpatentable by NetBIOS, further in view of RFC 1531 (See claim mapping chart utilizing alternative 103 rejections in Exhibit M, pages 1-15 for claims 1-7 and pages 25-35 for claims 32-42, incorporated by reference).

<u>In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilize the limitations taught by RFC 1531 in the invention taught</u>

by NetBIOS above since this would allow for automatic reuse of an address that is no longer needed by the host to which it was assigned (RFC 1531, Pg. 2), and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers.

7) Claims 10-17, 19-28, 30-31 are rejected under 35 U.S.C. 103(a) as being unpatentable by NetBIOS, further in view of Pinard (See claim mapping chart in Exhibit M, pages 15-25, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilize the user-interface elements and interactions taught by Pinard in the invention taught by NetBIOS since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates (Pinard, col. 2 lines 43-46), since NetBIOS teaches that it can be implemented using different operating systems (NetBIOS, pg. 359), and since examiner notes that both NetBIOS and Pinard relate to communications between at least two users implemented in a computerized environment.

8) Claims 18, 29, are rejected under 35 U.S.C. 103(a) as being unpatentable by NetBIOS, further in view of Pinard, further in view of VocalChat User's Guide (See claim mapping chart in Exhibit M, pages 19 and 24, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilize the limitations taught by VocalChat User's Guide in the invention taught by NetBIOS and Pinard above since all three references relate to the field of communications over a computer network, since VocalChat and Pinard utilize a computer system for telephony features specifically, and since examiner notes that the use of a MUTE feature in telephone conversations is old and well known in the art.

Issue 2

9) Examiner notes the following will represent the Etherphone references utilized for the rejection below (All considered a single reference as published together):

"Zellweger": An Overview of the Etherphone System and its Applications

"Swinehart": Telephone Management in the Etherphone System

"Terry": Managing Stored Voice in the Etherphone System

"Swinehart 2": System Support Requirements for Multi-media Workstations

"Zellweger 2": Active Paths through Multimedia Documents

10) Claims 43-44 are rejected under 35 U.S.C. 102(b) as being unpatentable by Etherphone (See claim mapping chart in Exhibit N, pages 33-35, incorporated by reference).

11) Claims 1-2, 4-7, 32-42 are rejected under 35 U.S.C. 103(a) as being unpatentable by Etherphone, further in view of Vin, further in view of RFC 1531 (See claim mapping chart utilizing alternative 103 rejections in Exhibit N, pages 1-12 for claims 1-2 and 4-7, and pages 24-33 for claims 32-42, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilize the computer program product taught by Etherphone above in an Internet based system utilizing dynamically assigned IP addresses from Internet access servers as taught by Vin and RFC 1531 since Etherphone was intended for use in multiple networks and communication protocols (Terry, page 3), since Vin and Etherphone both describe the same Etherphone system, since examiner notes that Internet and IP address-based networks are old and well known in the art and would be a natural extension from an ethernet-based system, since dynamic allocation of IP addresses allows for automatic reuse of an address that is no longer needed by the host to which it was assigned (RFC 1531, Pg. 2), and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers.

12) Claims 10-17, 19-28, 30-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Etherphone, further in view of Pinard (See claim mapping in the Request pages 118-126, and also claim mapping chart utilizing alternative 103 rejections in Exhibit N, pages 12-17 for claims 10-17, pages 17-23 for claims 19-28, and page 24 for claims 30-31, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilizing the user-interface elements and interactions taught by Pinard in the invention taught by Etherphone since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates (Pinard, col.

2 lines 43-46), and since examiner notes that both Etherphone and Pinard relate to communications between at least two users implemented in a computerized environment.

13) Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over Etherphone, further in view of Vin, RFC 1531 and NetBIOS (See claim mapping in the Request, pages 112-113 and also claim chart in Exhibit N, page 8, incorporated by reference).

In addition, examiner adds that it would have been obvious to combine NetBIOS with Etherphone, Vin, and RFC 1531 as taught above since examiner notes that all references teach the use of computer networking, and since examiner notes that the use of timestamps would allow for determination of length of time a user has been online and would also allow for removal of stale entries in the database.

14) Claims 18 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Etherphone, further in view of Pinard and VocalChat User's Guide (see claim chart in Exhibit N, page 17 for claim 18, and page 23 for claim 29, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to utilize the limitations taught by VocalChat User's Guide in the invention taught by Etherphone and Pinard above since all three references relate to the field of communications over a computer network, since VocalChat User's Guide and Pinard utilize a computer system for telephony features specifically, and since examiner notes that the use of a MUTE feature in telephone conversations is old and well known in the art.

Issue 3

14) Examiner notes the following will represent the VocalChat references utilized for the rejection below:

"User's Guide": VocalChat User's Guide, Version 2.0

"Readme": VocalChat Readme File, Version 2.02

"Networking Information": VocalChat 1.01 Networking Information

"Help File": VocalChat Information, Version 2.02

"Troubleshooting Help File": VocalChat Troubleshooting Help File, Version 2.02

15) Claims 1-2, 4-7, 32-42 are rejected under 35 U.S.C. 103(a) as being unpatentable by the combination of all five VocalChat references listed above (hereafter "VocalChat References"), further in view of RFC 1531 (See claim mapping chart utilizing alternative 103 rejections in Exhibit O, pages 1-15 for claims 1-2 and 4-7, and pages 26-36 for claims 32-42, incorporated by reference).

In addition, examiner adds that it would have been obvious to one skilled in the art at the time the invention was made to combine all five VocalChat References utilized above since they all describe a VocalChat system which shares numerous common features including a central server to store addresses and VocalChat client software and which all interoperate in the same basic manner. In addition it would have been obvious to utilize the limitations taught by RFC 1531 in the invention taught by VocalChat above since this allows for automatic reuse of an address that is no longer needed by the host to which it was assigned (RFC 1531, Pg. 2), and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers.

16) Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable by the combination of VocalChat, further in view of RFC 1531, further in view of NetBIOS (See claim mapping chart in Exhibit O, pages 9-10, incorporated by reference).

In addition, examiner adds that it would have been obvious to combine NetBIOS with VocalChat and RFC 1531 as taught above since examiner notes that all references teach the use of computer networking, and since examiner notes that the use of timestamps would allow for determination of length of time a user has been online and would also allow for removal of stale entries in the database.

17) Claims 43-44 are rejected under 35 U.S.C. 103(a) as being unpatentable by VocalChat.

Referring to **(Claim 43)**, VocalChat teaches a computer program product for use with a computer system, the computer system executing a first process operatively coupled over a computer network to a second process and a server process (User's Guide, pg. 5, 7-8, Help File, pg. 2), the computer program product comprising a computer usable medium having computer readable program code embodied therein, the program code comprising:

a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network (Help File, pg. 2, 26, Network

Information, pg. 10, whereby the network protocol address of clients are transmitted to and stored in a Connection List / USERS file located on the network); and

b. program code responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network (Help File, pg. 17, User Guide, pg. 2, whereby user-to-user access is facilitated through a connection list file over a network).

In addition, examiner notes that it would have been obvious to one skilled in the art at the time the invention was made to combine all five VocalChat References utilized above since they all describe a VocalChat system which shares numerous common features including a central server to store addresses and VocalChat client software and which all interoperate in the same basic manner.

Referring to **(Claim 44)**, VocalChat teaches in a first computer process operatively coupled over a computer network to a second process and an address server (User's Guide, pg. 5, 7-8, Help File, pg. 2), a method of establishing a point-to-point communication between the first and second processes comprising the steps of:

A. following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network (Help File, pg. 2, 26, Network Information, pg. 10, whereby the network protocol address of clients are transmitted to and stored in a Connection List / USERS file located on the network);

B. querying the address server as to whether the second process is connected to the computer network (Help File, pg. 2, 26, Network Information, pg. 10, whereby clients query the network server and whereby the server stores addresses of logged in users);

C. receiving a network protocol address of the second process from the address server, when the second process is connected to the computer network (Help File, pg. 2, 22, whereby the server transmits network addresses from the directory database); and

D. in response to the network protocol address of the second process, establishing a point-to-point communication link with the second process over the computer network (Help File, pg. 17, User Guide, pg. 2, whereby user-to-user access is facilitated through a connection list file over a network).

In addition, examiner notes that it would have been obvious to one skilled in the art at the time the invention was made to combine all five VocalChat References utilized above since they all describe a VocalChat system which shares numerous common features including a central server to store addresses and VocalChat client software and which all interoperate in the same basic manner.

18) Claims 10-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over VocalChat, further in view of Pinard.

Referring to **(Claim 10)**, VocalChat teaches in a computer system, a method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network (Help File, pg. 17, User Guide, pg.

2), the method comprising establishing a point-to-point communication link from the caller process to the first callee process (Help File, pg. 14, 20-21, whereby calls are established between two parties via network address).

In addition, VocalChat teaches the use of multiple user interface elements (User Guide, pg. 12, 14, Help File, pg. 11, 20-21). However, VocalChat does not explicitly teach providing a user interface element representing a first communication line, providing a user interface element representing a first callee process; and establishing communication in response to a user associating the element representing the first callee process with the element representing the first communication line.

Pinard teaches a human machine interface for telephone feature invocation which is utilized on a personal computer and allows a user to make telephone calls by moving graphics around a screen. Pinard teaches a user interface element representing a first communication line and callee process (Pinard, Figure 6 and col. 5 lines 23-30), and also teaches clicking and dragging an icon representing a callee from a directory into a call setup icon to establish a call link (Pinard, Figure 3, col. 4 lines 38-51, Figure 6, col. 5 lines 36-37).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to utilize the user-interface elements and interactions taught by Pinard in the invention taught by VocalChat since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates (Pinard, col. 2 lines 43-46), and since examiner notes that both VocalChat and Pinard relate to communications between at least two users implemented in a computerized environment.

Referring to **(Claim 21)**, VocalChat teaches a computer program product for use with a computer system comprising a computer usable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network (User's Guide, pg. 5, 7-8, readme, pg. 1, Help File, pg. 2), the medium further comprising program code for establishing a point-to-point communication link from the caller process to the first callee process (Help File, pg. 14, 20-21, whereby calls are established between two parties via network address).
In addition, VocalChat teaches the use of multiple user interface elements (User Guide, pg. 12, 14, Help File, pg. 11, 20-21). However, VocalChat does not explicitly teach program code for generating an element representing a first communication line and a first callee process and establishing communication responsive to a user associating the element representing the first callee process with the element representing the first communication line.

Pinard teaches a human machine interface for telephone feature invocation which is utilized on a personal computer and allows a user to make telephone calls by moving graphics around a screen. Pinard teaches a user interface element representing a first communication line and callee process (Pinard, Figure 6 and col. 5 lines 23-30), and also teaches clicking and dragging an icon representing a callee from a directory into a call setup icon to establish a call link (Pinard, Figure 3, col. 4 lines 38-51, Figure 6, col. 5 lines 36-37).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to utilize the user-interface elements and interactions taught by Pinard in the invention taught by VocalChat since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates (Pinard, col. 2 lines 43-46), and since examiner notes that both VocalChat and Pinard relate to communications between at least two users implemented in a computerized environment.

Referring to **(Claims 11-20 and 22-31)**, See claim mapping chart in Exhibit O, pages 17-20 for claims 11-20 and pages 22-26 for claims 22-31, incorporated by reference.

In addition, examiner notes that it would have been obvious to one skilled in the art at the time the invention was made to combine all five VocalChat References utilized above since they all describe a VocalChat system which shares numerous common features including a central server to store addresses and VocalChat client software and which all interoperate in the same basic manner. In addition it would have been obvious to utilize the limitations taught by Pinard in the invention taught by VocalChat above since this allows for automatic reuse of an address that is no longer needed by the host to which it was assigned (RFC 1531, Pg. 2), and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers.

Conclusion

All correspondence relating to this ex parte reexamination proceeding should be directed as follows:

By U.S. Postal Service Mail to:

Mail Stop Ex Parte Reexam
ATTN: Central Reexamination Unit
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to:

(571) 273-9900
Central Reexamination Unit

By hand to:

Customer Service Window
Randolph Building
401 Dulany St.
Alexandria, VA 22314

By EFS-Web:

Registered users of EFS-Web may alternatively submit such correspondence via the electronic filing system EFS-Web, at

https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html

EFS-Web offers the benefit of quick submission to the particular area of the Office that needs to act on the correspondence. Also, EFS-Web submissions are "soft scanned" (i.e., electronically uploaded) directly into the official file for the reexamination proceeding, which offers parties the opportunity to review the content of their submissions after the "soft scanning" process is complete.

Any inquiry concerning this communication or earlier communications from the Reexamination Legal Advisor or Examiner, or as to the status of this proceeding, should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

/Alexander J Kosowski/

Primary Examiner, Art Unit 3992

Electronic Patent Application Fee Transmittal

Application Number:	90010416
Filing Date:	17-Feb-2009
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Filer:	Michael Raymond Casey
Attorney Docket Number:	2655-0188

Filed as Large Entity

ex parte reexam Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Petition fee- 37 CFR 1.17(g) (Group II)	1463			

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
	Total in USD ($)			200

Electronic Acknowledgement Receipt

EFS ID:	6284307
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael Raymond Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	19-OCT-2009
Filing Date:	17-FEB-2009
Time Stamp:	13:50:02
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$200
RAM confirmation Number	52
Deposit Account	501860
Authorized User	

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent ap

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Reexam Request for Extension of Time	90010416_EOT_Request.pdf	382840 94dfe5a4493abf9c276d7038ee1d83a927a49560	no	3
Warnings:					
Information:					
2	Reexam Certificate of Service	90010416_COS.pdf	52350 99c977a6b9c66dc7d5ddf3ae97637dd80adf8bc8	no	1
Warnings:					
Information:					
3	Fee Worksheet (PTO-875)	fee-info.pdf	29573 badfb57da1af9a8a395bdd55e7573f23a4721f2f	no	2
Warnings:					
Information:					
			Total Files Size (in bytes):	464763	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

<u>New Applications Under 35 U.S.C. 111</u>
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

<u>National Stage of an International Application under 35 U.S.C. 371</u>
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

<u>New International Application Filed with the USPTO as a Receiving Office</u>
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

MAILED

OCT 26 2009

CENTRAL REEXAMINATION UNIT

DAVIDSON BERQUIST JACKSON & GROWDEY LLP 4300 WILSON BLVD., 7TH FLOOR ARLINGTON VA 22203	: : : :	(For Patent Owner)
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP 1270 OAKMEAD PARKWAY SUNNYVALE, CA 94085-7070	: : : :	(For Third-Party Requester)

In re HUTTON et al. Reexamination Proceeding Control No. 90/010,416 Request Deposited: February 17, 2009 For: U.S. Patent No. 6,108,704	: DECISION GRANTING : PETITION FOR : EXTENSTION OF TIME : [37 CFR 1.550(c)] :

This is a decision on the October 8, 2009, "REQUEST FOR EXTENSION OF TIME PURSUANT TO 37 C.F.R §1.550(c)" requesting that the time to submit a Patent Owner response be extended one month.

The petition is before the Director of the Central Reexamination Unit for consideration.

The petition is **granted** and a one-month extension of time is granted for the reasons set forth below.

REVIEW OF RELEVANT FACTS

1. U.S. Patent 6,108,704 issued to Hutton *et alia* on August 22, 2000.

2. On February 17, 2009 a third party requester requested ex parte reexamination of US Patent No. 6,108,704 which is identified under control no. 90/010,416.

3. The Order was granted for reexamination on March 11, 2009.

4. A First Office action was issued on August 27, 2009 setting a two-month period for response.

5. A petition for extension of time under 37 CFR 1.550(c) was filed October 19, 2009 requesting a one month extension.

DECISION

The Patent Owner requests an extension of time in which to file a response to the outstanding Office action. The present petition for extension of time was timely filed on October 8, 2009 together with the petition fee required by 37 CFR 1.515(c).

37 CFR 1.550 (c) states:

> (c) The time for taking any action by a patent owner in an ex parte reexamination proceeding will be extended only for sufficient cause and for a reasonable time specified. Any request for such extension must be filed on or before the day on which action by the patent owner is due, but in no case will the mere filing of a request effect any extension. Any request for such extension must be accompanied by the petition fee set forth in § 1.17(g). See § 1.304(a) for extensions of time for filing a notice of appeal to the U.S. Court of Appeals for the Federal Circuit or for commencing a civil action.

Addressing the requirement of 37 CFR 1.550 (c) to make a showing of' "sufficient cause" to grant an extension of time request, MPEP 2265 states, in pertinent part:

> Evaluation of whether sufficient cause has been shown for an extension must be made in the context of providing the patent owner with a fair opportunity to present an argument against any attack on the patent, and the requirement of the statute (35 U.S.C. 305) that the proceedings be conducted with special dispatch ....
>
> Any request for an extension of time in a reexamination proceeding must fully state the reasons therefor ....
>
> The reasons stated in the request will be evaluated by the CRU Director, and the requests will be favorably considered where there is a factual accounting of reasonably diligent behavior by all those responsible for preparing a response within the statutory time period.

Patent Owner's Showing of Sufficient Cause to Grant an Extension of Time

The request notes what steps Owner has taken to date in preparation of a response to the Office action. Further, the request outlines why, in spite of the actions taken thus far, the requested additional time is believed necessary.

Analysis and Findings

On balance it is considered that the petition explains the "sufficient cause" for an extension of time. It is clear Patent Owner requires some additional time to prepare a response to the outstanding Office action. See pages 2-3 of the petition for details. An extension of time of one month is considered sufficient.

Patent Owner should expect that future requests for extensions will not be granted absent strong and compelling reasons that establish the existence of an extraordinary situation necessitating the additional time.

CONCLUSION

1. Petitioner's request is **granted**. The time for filing a response to the outstanding Office action is extended for one month and is due on or before November 27, 2009.

2. All correspondence relating to this *ex parte* reexamination proceeding should be directed:

By EFS: Registered users may submit via the electronic filing system EFS-Web, at **https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html**.

By Mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

For EFS-Web transmissions, 37 CFR 1.8(a)(1)(i) (C) and (ii) states that correspondence (except for a request for reexamination and a corrected or replacement request for reexamination) will be considered timely filed if (a) it is transmitted via the Office's electronic filing system in accordance with 37 CFR 1.6(a)(4), and (b) includes a certificate of transmission for each piece of correspondence stating the date of transmission, which is prior to the expiration of the set period of time in the Office action.

3. Telephone inquiries related to this decision should be directed to Eric Keasel, at (571) 272-4929, Jessica Harrison at (571) 272-4449 or Mark Reinhart, at (571) 272-1611.

/J. Harrison/ for

Gregory Morse
Director, Central Reexamination Unit

Electronic Patent Application Fee Transmittal

Application Number:	90010416
Filing Date:	17-Feb-2009
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Filer:	Michael Raymond Casey
Attorney Docket Number:	2655-0188

Filed as Large Entity

ex parte reexam Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Petition fee- 37 CFR 1.17(g) (Group II)	1463			

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
	Total in USD ($)			**200**

Electronic Acknowledgement Receipt	
EFS ID:	6503705
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael Raymond Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	23-NOV-2009
Filing Date:	17-FEB-2009
Time Stamp:	12:04:32
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$200
RAM confirmation Number	8169
Deposit Account	501860
Authorized User	

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent ap

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Reexam Request for Extension of Time	20091123_EOT_request.pdf	652300 a738821d2dd1ec62f6db703bc394ef6564b6666b	no	7
Warnings:					
Information:					
2	Reexam Certificate of Service	20091123_COS.pdf	53579 3d48b703ddeebe5e1ad7e0fb6475fc2d4b004358	no	1
Warnings:					
Information:					
3	Fee Worksheet (PTO-875)	fee-info.pdf	29576 84f06dd76e04ef0733c8cee86391a3b9f5caea51	no	2
Warnings:					
Information:					
			Total Files Size (in bytes):	735455	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

<u>New Applications Under 35 U.S.C. 111</u>
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

<u>National Stage of an International Application under 35 U.S.C. 371</u>
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

<u>New International Application Filed with the USPTO as a Receiving Office</u>
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/010,416	02/17/2009	6108704	2655-0188	1061

42624 7590 11/25/2009

DAVIDSON BERQUIST JACKSON & GOWDEY LLP
4300 WILSON BLVD., 7TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 11/25/2009

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patents and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS
Edwin H. Taylor
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP
1279 Oakmead Parkway
Sunnyvale, CA 94085-4040

Date:
MAILED
NOV 25 2009
CENTRAL REEXAMINATION UNIT

EX PARTE REEXAMINATION COMMUNICATION TRANSMITTAL FORM

REEXAMINATION CONTROL NO. : 90010416
PATENT NO. : 6108704
ART UNIT : 3992

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified ex parte reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the ex parte reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

Michael R Casey DAVIDSON BERQUIST JACKSON & GOWDEY LLP 4300 WILSON BLVD., 7TH FLOOR ARLINGTON VA 22203	:	(For Patent Owner
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP 1279 OAKMEAD PARKWAY SUNNYVALE, CA 94085-4040	:	(For Third Party Requester)
In re: Hutton *et alia* *Ex Parte* Reexamination Proceeding Control No. 90/010,416 Deposited: 17 February 2009 For: US Patent No. 6,108,704	:	DECISION DISMISSING PETITION FOR EXTENSION OF TIME 37 CFR § 1.550(c) & 1.181

MAILED

NOV 2 5 2009

CENTRAL REEXAMINATION UNIT

This is a decision on the 23 November 2009, "Supplemental Request for Extension of Time in a Re-Examination" filed under 37 CFR § 1.550(c) requesting that the time for responding to the non-final Office action dated 27 August 2009, be further extended by one (1) week. The petition was filed with the required certificate of service and petition fee. The petition was timely filed.

The petition is before the Director of the Central Reexamination Unit for consideration.

The petition is dismissed for the reasons set forth below.

DISCUSSION

The Patent Owner requests the period of time be further extended in which to file a response to non-final Office action dated 27 August 2009 which set a two (2) month period for response thereto. A first petition for extension of time was granted extending the response period by one month in the decision dated 26 October 2009. The subsequent petition was timely filed with the required certificate of service and petition fee pursuant to 37 CFR §§ 1.550(c) and 1.17(g). Second requests for extension of time will only be granted in exception circumstances.

The petition for extension of time dated 23 November 2009 is dismissed.

37 CFR § 1.550 (c) states:

(c) The time for taking any action by a patent owner in an *ex parte* reexamination proceeding will be extended only for sufficient cause and for a reasonable time specified. Any request for such extension must be filed on or before the day on which action by the patent owner is due, but in no case will the mere filing of a request effect any extension. Any request for such extension must be accompanied by the petition fee set forth in § 1.17(g). See § 1.304(a) for extensions of time for filing a notice of appeal to the U.S. Court of Appeals for the Federal Circuit or for commencing a civil action. (emphasis added)

MPEP § 2265 Extension of Time (in-part)

...

Ex parte prosecution will be conducted by initially setting either a 1-month or a 2-month shortened period for response, see MPEP § 2263. The patent owner also will be given a 2-month period after the order for reexamination to file a statement >(by statute (35 U.S.C. § 304), this period cannot be less than 2-months, even in a proceeding where the patent is being litigated). See 37 CFR § 1.530(b). First requests for extensions of these statutory time periods will be granted for sufficient cause, and for a reasonable time specified — usually 1 month. The reasons stated in the request will be evaluated by the CRU or TC Director, and the requests will be favorably considered where there is a factual accounting of reasonably diligent behavior by all those responsible for preparing a response within the statutory time period. Second or subsequent requests for extensions of time or requests for more than 1 month will be granted only in extraordinary situations.

...

ANALYSIS AND FINDINGS

The patent owner's representative petitions to extend the period for response by adding thirty (30) days to the period for response. The decision to extend the period for response is evaluated based upon a showing of "sufficient cause." There is always the consideration to balance the need for the patent owner to have a fair opportunity to respond to the Office action between the need for special dispatch.

The patent owner timely submitted a first petition for extension which was granted on 26 October 2009. This first extension of time granted one additional month for which the patent owner to respond to the outstanding Office action. Second or subsequent requests for extensions of time or requests for more than 1 month will be granted only in extraordinary situations. The factual accounting presented by the petitioner does not meet the level of extraordinary circumstances.

The petition request to extend the response time is hereby dismissed.

CONCLUSION

1. The patent owner's petition for extension of time is hereby **dismissed.**

2. The time to respond continues to run.

3. Response is due on **27 November 2009.**

4. Response and/or submissions to the Office should be addressed as follows:

By Mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P. O. Box 1450
Alexandria, VA 22313-1450

By Fax to: (571) 273-9900
Central Reexamination Unit

By Hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

By EFS: Registered users of EFS-Web may alternatively submit such correspondence via the electronic filing system EFS-Web, at https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html. EFS-Web offers the benefit of quick submission to the particular area of the Office that needs to act on the correspondence. Also, EFS-Web submissions are "soft scanned" (i.e., electronically uploaded) directly into the official file for the reexamination proceeding, which offers parties the opportunity to review the content of their submissions after the "soft scanning" process is complete.

5. Telephone inquiries with regard to this decision should be directed to Mark Reinhart, at (571) 272-1611, in the event that Mark Reinhart is unavailable Eric Keasel at (571) 272-4929, or Jessica Harrison at (571) 272-4449; all are Supervisory Patent Examiners in the Central Reexamination Unit, Art Unit 3992 may also be contacted..

/Mark Reinhart/
for

Gregory Morse
Director,
Central Reexamination Unit 3999

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION OF:

Net2Phone, Inc. (Patent No. 6,108,704)

Control No.: 90/010,416

Issue Date: August 22, 2000

Title: **POINT-TO-POINT INTERNET PROTOCOL**

Attorney Docket: 2655-0188

Group Art Unit: 3992

Examiner: KOSOWSKI, Alexander

Date: November 27, 2009

Confirmation No.: 1061

RESPONSE TO NON-FINAL REJECTION IN A RE-EXAMINATION

Hon. Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Office Action dated August 27, 2009 (and having had the deadline for responding extended one month), the Assignee hereby submits:

Claim Amendments beginning on page 2 of this paper; and

Remarks/Arguments beginning on page 6 of this paper.

AMENDMENTS TO THE CLAIMS

Please amend the claims in re-examination as follows:

10. (Canceled)

11. (Amended) In a computer system, a method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the method comprising the steps of:

A. providing a user interface element representing a first communication line;

B. providing a user interface element representing a first callee process; and

C. establishing a point-to-point communication link from the caller process to the first callee process, in response to a user associating the element representing the first callee process with the element representing the first communication line, [The method of claim 10] wherein step C further comprises the steps of:

c.1 querying the server as to the on-line status of the first callee [process] process; and

c.2 receiving a network protocol address of the first callee process over the computer network from the server.

12. (Amended) The method of claim [10] 11 further comprising the step of:

D. providing an element representing a second communication line.

14. (Amended) The method of claim [10] 11 further comprising the steps of:

D. providing a user interface element representing a second callee process; and

E. establishing a conference point-to-point communication link between the caller process and the first and second callee process, in response to the user associating the element representing the second callee process with the element representing the first communication line.

15. (Amended) The method of claim [10] <u>11</u> further comprising the step of:

F. removing the second callee process from the conference point-to-point communication link in response to the user disassociating the element representing the second callee process from the element representing the first communication line.

16. (Amended) The method of claim [10] <u>11</u> further comprising the steps of:

D. providing a user interface element representing a communication line having a temporarily disabled status; and

E. temporarily disabling a point-to-point communication link between the caller process and the first callee process, in response to the user associating the element representing the first callee process with the element representing the communication line having a temporarily disabled status.

19. (Amended) The method of claim [10] <u>11</u> wherein the caller process further comprises a visual display and the user interface comprises a graphic user interface.

21. (Canceled)

22. (Amended) <u>A computer program product for use with a computer system comprising:</u>

<u>a computer usable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the medium further comprising:</u>

<u>program code for generating an element representing a first communication line;</u>

<u>program code for generating an element representing a first callee process;</u>

<u>program code, responsive to a user associating the element representing the first callee process with the element representing the first communication line, for establishing a point-to-</u>

point communication link from the caller process to the first callee process. [The computer program product of claim 21] wherein the program code for establishing a point-to-point communication link further comprises:

program code for querying the server as to the on-line status of the first callee process; and

program code for receiving a network protocol address of the first callee process over the computer network from the server.

23. (Amended) A computer program product of claim [21] 22 further comprising: program code for generating an element representing a second communication line.

25. (Amended) The computer program product of claim [21] 22 further comprising:

program code for generating an element representing a second callee process; and

program code means, responsive to the user associating the element representing the second callee process with the element representing the first communication line, for establishing a conference communication link between the caller process and the first and second callee process.

27. (Amended) The computer program product of claim [21] 22 further comprising:

program code for generating an element representing a communication line having a temporarily disabled status; and

program code, responsive association of the element representing the first callee process with the element representing the communication line having a temporarily disabled status, for temporarily disabling the point-to-point communication link between the caller process and the first callee process.

30. (Amended) A computer program product of claim [21] 22 wherein the computer system further comprises a visual display and the user interface comprises a graphic user interface.

REMARKS/ARGUMENTS

Favorable reconsideration of the claims currently undergoing re-examination, in view of the present amendment and in light of the following discussion, is respectfully requested.

STATUS OF THE CLAIMS AND SUPPORT FOR THE CLAIM CHANGES

Claims 1-7, 11-20 and 22-44 are pending and the subject of this re-examination. Claims 11 and 22 have been rewritten in independent form without changing their scope, thus those amendments are self supporting. Their corresponding independent claims (i.e., claims 10 and 21) have been canceled. The dependencies of claims 12, 14, 15, 16, 19, 23, 25, 27 and 30 have been amended in light of the cancelation of claims 10 and 21 to depend from new independent claims 11 and 22. In light of the changes to the dependencies, these changes are self supporting. Thus, no new matter has been added. No other claims have been amended, added or canceled herewith.

RESPONSE TO REJECTIONS

In the outstanding Office Action, five main rejections were made as follows:

1. Claims 43 and 44 were alleged to be anticipated by NetBIOS;
2. Claims 1-7 and 10-42 were alleged to be rendered obvious by NetBIOS in combination with at least one other reference;
3. Claims 43 and 44 were alleged to be anticipated by the Etherphone papers;
4. Claims 1-7 and 10-42 were alleged to be rendered obvious by the Etherphone papers in combination with at least one other reference; and
5. Claims 1-7 and 10-44 were alleged to be rendered obvious by the VocalChat References, either alone or in combination with at least one other reference.

Each of those rejections is respectfully traversed for the reasons set forth below. Reference is made throughout this response to the Declaration Of Ketan Mayer-Patel Under 37 C.F.R. 1.132 (hereinafter the "Mayer-Patel Declaration") attached hereto as Exhibit 1.

The Rejection of Claims 43 and 44 over NetBIOS

Claim 43 recites "a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network." This limitation is not taught by the NetBIOS reference as NetBIOS does not provide dynamic addressing or on-line status. See Exhibit 1, Mayer-Patel Declaration, paragraph 21.

In rejecting claim 43, the Office Action adopts the positions of the third-party requester and states "See claim mapping chart in Exhibit M, pages 36-40, incorporated by reference." With respect to this limitation (a), the claim mapping does not allege, much less prove, that NetBIOS teaches "the network protocol address of each respective process forwarded to the database ***following connection to the computer network***." In fact, the Office Action appears to have agreed (e.g., with respect to claim 1) that the NetBIOS reference does not teach that the processes receive network protocol addresses "following connection to the computer network." The Office Action did this by rejecting the requester's arguments under 35 U.S.C. § 102 and instead adopting "claim chart mapping utilizing alternative 103 rejections" -- rejections that rely on RFC 1531 to teach dynamic addressing. See Exhibit 1, Mayer-Patel Declaration, paragraph 22.

Even assuming that the Office Action intended the rejection to be a rejection under 35 U.S.C. § 103 by combining NetBIOS with RFC 1531, the rejection would still not be proper. When alleging that one of ordinary skill in the art would have combined NetBIOS with RFC 1531, the Office Action states "it would have been obvious … to utilize the limitations taught by RFC 1531 in the invention taught by NetBIOS … since this would allow for automatic reuse of an address ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known ... and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers." The assignee respectfully submits that the conclusion drawn by the Office Action on the combinability of NetBIOS and RFC 1531 is mistaken in its conclusion on motivation. The Office Action speculates, with hindsight, as to why a person of ordinary skill might want to combine the two references, but does not acknowledge the problems that would arise in doing so, and does not provide any prior art that would indicate how the problems that dynamic addressing would bring into a NetBIOS type system could be resolved by those of ordinary skill at the time the patent was filed. See Exhibit 1, Mayer-Patel Declaration, paragraph 23. In the context of point-to-point communication, widespread use of dynamically assigned addresses does not solve NetBIOS's problems, it creates further problems. The assignee agrees that dynamically assigned addresses were known, and the patent in re-examination specifically states in that regard, "Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain." Col. 1, lines 53-56. See Exhibit 1, Mayer-Patel Declaration, paragraph 24.

But it is not enough to prove that the cause of a problem existed, namely the problematic use of changing addresses. The Office Action must show by citation of prior art that the problem was recognized, and that the solution for NetBIOS was either known or trivially apparent from the known art. See *Innogenetics, N.V. v. Abbott Laboratories*, 512 F.3d 1363, 1373 (Fed Cir. 2008) ("The district court was nevertheless correct that knowledge of a problem and motivation to solve it are entirely different from motivation to combine particular references to reach the particular claimed method."). If the requester of this reexamination had such prior art it would undoubtedly have been provided as part of its exhaustive reexamination request. The fact that there is none is testimony to the lack of teaching in the prior art to make the suggested combination.

The NetBIOS reference cited in the request, moreover, indicates the opposite. For example, Section 15.1.7 of the NetBIOS reference (entitled "Consistency of the NBNS Data Base") recognizes that the association between a node, a registered name and an IP address is tenuous, even in an environment that uses static IP addresses. "Even in a properly running NetBIOS scope the NBNS and its community of end-nodes may occasionally lose synchronization with respect to the true state of name registrations." To minimize the impact of this problem, the reference states, "Various approaches have been incorporated into the NetBIOS-over-TCP protocols" which it then proceeds to describe. See Exhibit 1, Mayer-Patel Declaration, paragraph 26.

However, by incorporating DHCP and adopting dynamic address allocation as used by Internet access providers, the synchronization problem would become more disruptive, not less. Dynamic addressing would have introduced a new uncertainty to the relationships among the NBNS and its community of end-nodes and a new set of obstacles to NetBIOS synchronization that ***are not addressed by the NetBIOS reference.*** Consider the case of a node that is turned-off and then subsequently turned back on, or a node that has simply lost its Internet connection for some technical reason or whose DHCP lease has expired and then re-established a connection. In a dynamic addressing environment, such a node would most likely obtain a new IP address when it was turned back on that was different than the one it had when it registered its name. This change could lead to any number of node-name-IP address synchronization problems for the disclosed NetBIOS protocols. See Exhibit 1, Mayer-Patel Declaration, paragraph 27.

For example, because the NBNS does not know the node's new address, the NBNS would be unable to send to the node a Name Release Request or a Name Conflict Demand or request that the node send it a Name Status Request. Because communication from the node would be originating at a new address that was not recognized by the NBNS, a node's response to a Name Query Request (assuming it somehow knew that its name had been challenged, perhaps from before it lost network connectivity) would not be recognized. A node would also be unable to confirm its association with registered names by sending Name Refresh Request packets to the NBNS. If a session between two NetBIOS applications were cut-off, re-establishing the communication would be especially difficult where the ability of a called entity to obtain both its associated name and its associated IP address were in doubt. As a result, the Office Action has not demonstrated that a solution to the problems created by exposure of NetBIOS to DHCP and dynamic addressing has been addressed by any of the applied references.[1] See Exhibit 1, Mayer-Patel Declaration, paragraph 28.

The Office Action also has not identified anything in the cited art that suggests how a person of ordinary skill is to go about the redesign of NetBIOS and the solving of obstacles to NetBIOS operation that are created by Internet access; ***problems that were recognized and left as warnings unresolved in the NetBIOS reference.***[2] See Exhibit 1, Mayer-Patel Declaration, paragraph 29.

Merely citing to dynamic addressing, i.e., the source of those problems, is insufficient as the Supreme Court and the Federal Circuit have repeatedly made clear. See *Depuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 567 F.3d 1314, 1326 (Fed. Cir. 2009) citing *inter alia KSR Int'l Co. v Teleflex Inc.*, 550 U.S. 398 (2007) and *U.S. v Adams*, 383 U.S. 39 (1966), for the proposition that obviousness requires not only "the expectation that prior art elements are capable of being physically combined, but also that the combination would have worked for its intended purpose," and quoting *In re ICON Health & Fitness, Inc.*, 496 F.3d 1374, 1382 (Fed.

[1] Besides dynamic addressing, Internet access would pose other challenges to a NetBIOS system. For example, because NetBIOS was designed for use on local area networks with small numbers of computers, trust among the network participants is assumed. That assumption cannot be transferred to a global Internet made up of unknown, and sometimes malevolent, entities. An implementation of NetBIOS on the public Internet would necessitate non-trivial adaptations to ensure that its services perform correctly and return accurate information. There is no discussion of security issues in the cited references. See Exhibit 2, from http://www.w3schools.com/Site/site_security.asp which instructs Microsoft Windows users whose computers access the Internet to disable NetBIOS over TCP/IP in order to solve their security problems. See Exhibit 1, Mayer-Patel Declaration, paragraph 28.

[2] The cited references go out of their way *to avoid* describing how a NetBIOS protocol might work in inter-connected network environments that that are *less complex* than the Internet and that *predate* DHCP. See Section 4.6 ("The proposed standard recognizes the need for NetBIOS operation across a set of networks interconnected by network (IP) level relays (gateways.) However, the standard assumes that this form of operation will be less frequent than on the local MAC bridged-LAN.")

Cir. 2007) as saying "[A] reference teaches away from a combination when using it in that combination would produce an inoperative result."

In view of the foregoing, a rejection of claim 43, even over the combination of NetBIOS and RFC 1531, can be compared to the rejection of a patent that claims a vehicle that travels on water where one piece of prior art shows a land vehicle and another shows water. The fact that water creates a problem for the land vehicle does not disclose that the person of ordinary skill would know how to build a vehicle capable of crossing the water. Thus, claim 43 is patentable over NetBIOS alone or over NetBIOS in combination with RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 30.

NetBIOS and RFC 1531 also do not teach "a. program code configured to access a directory database…having a network protocol address for a selected plurality of processes ***having on-line status with respect to the computer network.***" While NetBIOS uses name entries with "active" statuses as part of its name management process, an analysis of how that "active" status is used shows that "an active name" is not synonymous with an "on-line status with respect to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not on-line. As shown on page 447 (and reproduced below), the Node_Name entries stored with respect to a NetBIOS Name Server contain a series of fields including the "ACT" field.

```
The NAME_FLAGS field:

                                          1   1   1   1   1   1
  0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| G |  ONT  |DRG|CNF|ACT|PRM|          RESERVED                 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

The NAME_FLAGS field is defined as:

Symbol     Bit(s)   Description:

RESERVED    7-15    Reserved for future use.  Must be zero (0).
PRM            6    Permanent Name Flag.  If one (1) then entry
                    is for the permanent node name.  Flag is zero
                    (0) for all other names.
ACT            5    Active Name Flag.  All entries have this flag
                    set to one (1).
CNF            4    Conflict Flag.  If one (1) then name on this
                    node is in conflict.
DRG            3    Deregister Flag.  If one (1) then this name
                    is in the process of being deleted.
ONT          1,2    Owner Node Type:
                       00 = B node
                       01 = P node
                       10 = M node
                       11 = Reserved for future use
G              0    Group Name Flag.
                    If one (1) then the name is a GROUP NetBIOS
                    name.
                    If zero (0) then it is a UNIQUE NetBIOS name.
```

See Exhibit 1, Mayer-Patel Declaration, paragraph 31.

The ACT field is a single bit field (in bit 5) that signifies an "Active Name Flag. ***All entries have this flag set to one (1).***" (Emphasis added.) If all name entries have this flag set to one (1), then the NetBIOS name server cannot be using the Active Name Flag as a means of separately tracking whether the entity that owns the name is "active," let alone what its "on-line status" might be. See Exhibit 1, Mayer-Patel Declaration, paragraph 32.

The NetBIOS reference also does not teach that the active status of a name in the NetBIOS server is an indication of the active status of the owner of that name. To the contrary, when information about whether the owner of a name is "active" may be relevant, for example when a new entity seeks to register a name that has already been registered in the NetBIOS name server, the NetBIOS reference describes an elaborate set of interactions used to test whether the existing owner of the registered name is active or inactive. It does not rely on the fact that the name is active in the NetBIOS name server (See Section 15.2.2.2 and 15.2.2.3 entitled "Existing Name and Owner is Inactive"). See Exhibit 1, Mayer-Patel Declaration, paragraph 33.

The NetBIOS reference also does not teach that an acquired IP address can be reasonably relied upon by a requesting end-node to confirm the "on-line status" of an end-node associated with a sought name. The NetBIOS reference describes at least two different scenarios where a second end-node sends a ***rejection response*** to the first end-node ***notwithstanding the fact that a second end-node is connected*** to the computer network and active with respect to the sought name. See Section 16.1.1 ("There exists a NetBIOS LISTEN compatible with the incoming call, but there are inadequate resources to permit establishment of a session...The called name does, in fact, exist on the called node, but there is no pending NetBIOS LISTEN compatible with the incoming call."). No distinction is made in the reference between the rejection response in these cases and the rejection response in cases where the called name does not exist on the called end-node. *Id.*, ("In all but the first case, a rejection response is sent back over the TCP connection to the caller."). See Exhibit 1, Mayer-Patel Declaration, paragraph 34.

Thus, the limitation "program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network" of claim 43 is not taught by the NetBIOS reference alone or in combination with RFC 1531, and the rejection of claim 43 should be withdrawn. See Exhibit 1, Mayer-Patel Declaration, paragraph 35.

Claim 44 recites "***following connection of the first process to the computer network*** forwarding to the address server a network protocol address at which the first process is connected to the computer network." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach the claimed dynamic address assignment. Moreover, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 36.

Claim 44 also recites "querying the address server as to ***whether the second process is connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "whether the second process is connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 44 is patentable over both NetBIOS alone and the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 37.

Claims 1-7 and 32-42

Claim 1 recites "program code for transmitting to the server ***a network protocol address received by the first process following connection to the computer network***." Claim 1 further recites "program code for receiving a network protocol address of the second process from the server, ***when the second process is connected to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 38.

Claim 1 also recites "program code for transmitting, to the server, a query as to ***whether the second process is connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "whether the second process is connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 1 patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 39.

Claim 2 recites "each network protocol address stored in the memory ***following connection of a respective process to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 40.

Claim 2 also recites "means... for determining the ***on-line status of the second process*** and for ***transmitting a network protocol address of the second process*** to the first process ***in response to a positive determination of the on-line status of the second process***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "an on-line status of the second process." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 2 and its dependent claim 3 are patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 41.

Claim 4 recites "each of the network protocol addresses received following connection of the respective process to the computer network." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and

RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 42.

Claim 4 also recites "receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an ***on-line status*** with respect to the computer network." Claim 4 further recites "determining the ***on-line status*** of the second process." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "an on-line status" of a process. An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 4 and its dependent claims 5-7 are patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 43.

Claim 32 recites "the Internet Protocol address ***added to the list following connection of the process to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 44.

Claim 32 also recites "an Internet accessible list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of ***a process currently connected to the Internet***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process currently [being] connected to the Internet." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the Internet. Thus, claim 32 is patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 45.

Claim 33 recites "***the network protocol address*** of the corresponding process ***assigned to the process upon connection to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 46.

Claim 33 also recites "selected ... entries comprising **a network protocol address** and a corresponding identifier ***of a process connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process [being] connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 33 and its dependent claims 34-37 are patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 47.

Claim 38 recites "***the network protocol address*** of the corresponding process ***assigned to the process upon connection to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 48.

Claim 38 also recites "selected ... entries comprising a network protocol address and a corresponding identifier of ***a process connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process [being] connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 38 and its dependent claims 39-42 are patentable over the combination of NetBIOS and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 49.

Claims 10-31

Claims 10-31 are rejected under 35 U.S.C. § 103(a) as being obvious over NetBIOS in view of Pinard, either alone or in combination with VocalChat User's Guide. That ground for rejection is respectfully traversed.

Amended claim 11 recites "the caller process having a user interface and being operatively connectable to the callee process" and "establishing a point-to-point communication link ***from the caller process*** to the first callee process." When taken together, it is clear that the Office Action has not alleged, much less proven, that the caller process, having a user interface, establishes a point-to-point communication link with a first callee process. The Exhibit M of the Request for Re-examination (the "Request") states that "Pinard discloses that a point-to-point communication link is established in response to a user associating an element representing ***the first callee process*** with the element representing a first communication line." However, as shown in Figure 1, Pinard discloses that conventional telephones connected to a telephony server via telephony interface circuits make calls to other conventional telephones. Those telephones are not caller and callee processes. In fact, with respect to the limitation "providing a user interface element representing a first callee process," Exhibit M identifies, instead of a callee process, two people, Mary and John. Clearly neither is a callee ***process***. See Exhibit 1, Mayer-Patel Declaration, paragraph 51.

Exhibit M further states that "Applications which utilize NetBIOS application services ***inherently*** include 'user interfaces.'" This assertion is disputed. Page 356 of NetBIOS states "Protocols supporting NetBIOS services have been constructed on diverse protocol and ***hardware*** foundations." Thus, the NetBIOS applications could be running on a computer system (e.g., an embedded system) such that there are no user interfaces at all. In fact, the example provided in Exhibit M is disingenuous. Exhibit M states that "NetBIOS provides a vendor independent interface for the IBM Personal Computer (PC) and compatible systems." The "interface" discussed in that sentence is a software ***communications*** interface, between NetBIOS service elements and NetBIOS-compatible applications, not a ***user*** interface. Moreover, the reference to "IBM Personal Computer (PC) and compatible systems" is a reference to the hardware, not the software with user interfaces that is running on the hardware. Exhibit M itself admits the deficiency in the inherency argument when it states "it is ***expected*** that on computers operating under the PC-DOS and MS-DOS operating systems that the existing NetBIOS interface will be preserved." This says nothing about what processes with user interfaces can participate in the claimed point-to-point communications. See Exhibit 1, Mayer-Patel Declaration, paragraph 52.

With respect to the establishing step, Exhibit M does not even assert that one of the processes running on one of the DOS-style computers is the caller process. This is because neither the phone application not the device agent are part of the point-to-point communication -- the telephones are. Thus, this limitation is not met. See Exhibit 1, Mayer-Patel Declaration, paragraph 53.

In addition, the Office Action has not established that one of ordinary skill in the art would have made the proposed combination. The Office Action alleges that "it would have been obvious ... to utilize the user-interface elements and interaction taught by Pinard in the invention taught by NetBIOS since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates ... since NetBIOS teaches that it can be implemented using different operating systems ... and since examiner notes that both

NetBIOS and Pinard relate to communications between at least two users implemented in a computerized environment." However, Pinard already discloses a personal computer 1 interacting with a server 5 without any indication that there is something missing from Pinard. Thus, without an indication that Pinard was somehow deficient, one of ordinary skill in the art would not have been motivated to include NetBIOS with Pinard. Accordingly, amended claim 11 and its dependent claims 12-20 are not rendered obvious by the combination of NetBIOS and Pinard. See Exhibit 1, Mayer-Patel Declaration, paragraph 54.

Amended claim 22, like amended claim 11, recites "***the caller process having a user interface*** and being operatively connectable to the callee process" and "establishing a ***point-to-point communication link from the caller process*** to the first callee process." As was set forth for the patentability of amended claim 11 above, that combination of elements is not taught by the applied combination of references. Thus, amended claim 22 and its dependent claims 23-31 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 55.

<u>The rejection of claims 43 and 44 as anticipated by the Etherphone papers</u>

Claims 43 and 44 have been rejected under 35 U.S.C. § 102(b) as anticipated by the Etherphone papers. Those grounds for rejection are respectfully traversed.

Claim 43 recites "a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network." Exhibit N of the Request asserts that this limitation is taught by the system directory database of local Xerox employees and cites Zellweger, page 4. However, in the paragraph just prior to the section cited by Exhibit N, Zellweger states "Figure 3 shows an Etherphone control window, called Finch, and a personal telephone directory window." The caption for Figure 3 further explains that "the lower window [of Figure 3] shows a portion of a personal telephone directory, which is a set of speed-dialing buttons that can be created easily from an ordinary text file." See Exhibit 1, Mayer-Patel Declaration, paragraph 58.

Figure 3 does not show that the cited database includes the claimed "network protocol address for a selected plurality of processes having on-line status with respect to the computer network" -- rather it simply appears to include phone numbers for the various entries, including phone numbers outside of the Etherphone system. Thus, this limitation is not taught by the Etherphone references, and the rejection of claim 43 should be withdrawn. See Exhibit 1, Mayer-Patel Declaration, paragraph 59.

The Office Action also has not shown that the Etherphone papers disclose "the network protocol address of each respective process forwarded to the database following connection to the computer network" as the Office Action has not shown how such a dynamic addressing technique is taught by the Etherphone papers. Thus, this limitation is not anticipated by the Etherphone papers as well. See Exhibit 1, Mayer-Patel Declaration, paragraph 60.

As described above, claim 43 recites "a selected plurality of processes having on-line status with respect to the computer network." The Request and Exhibit N do not show how the Etherphone papers teach this recitation limitation. In fact, "on-line" status is not mentioned in any of the corresponding support sections for claim 43 in either the Request or Exhibit N. Further, the Etherphone papers do not teach an association between users' on-line status and the maintaining of a network address. To the contrary, the Etherphone system appears to always associate a user with a location, ***whether or not the user is currently at that location,*** and directs telephone calls for that user to that location ***without regard to the users' on-line status***. So, for example, while the Request points out that logging in to a workstation tells the Etherphone system—through unspecified mechanisms—where calls should be directed (citing Swinehart 1, page 2 as stating "Calls are to individuals, not locations...Logging in tells the telephone system where Karmen is."), the Request neglects to point out that ***logging out also tells the telephone system where calls should be directed.*** ("Since Karmen has signed off from her workstation, the call rings immediately at an attendant's workstation..." Swinehart page 3). See Exhibit 1, Mayer-Patel Declaration, paragraph 61.

Even assuming that the Office Action intended the rejection of claim 43 to be an obviousness rejection, claim 43 still is patentable over the Etherphone papers in view of Vin and RFC 1531. The Office Action alleges, with respect to claim 1, that it would have been obvious ... to utilize dynamically assigned IP address from Internet access servers as taught by Vin and RFC 1531 since Etherphone was intended for use in multiple networks and communication protocols (Terry, page 3), ... [and] since dynamic allocation of IP addresses allows for automatic reuse of an address that is no longer needed by the host to which it was assigned ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked components." See Exhibit 1, Mayer-Patel Declaration, paragraph 62.

Terry, page 3, however, does not indicate under what circumstances the "multiple networks and communications protocols" would be used. The full paragraph that appears to be referenced by the Office Action states:

> The Etherphone system is intended for use in a locally distributed computing environment containing multiple workstations and programming environments, multiple networks and communication protocols, and perhaps even multiple telephone transmission and switching choices. The system is intended to be extensible in that introducing new applications, network services, workstations, networks, and other components is possible.

See Exhibit 1, Mayer-Patel Declaration, paragraph 63.

Thus, it is probable that the "multiple networks and communications protocols" refers to uses of the workstations and not the Etherphones. Similarly, Vin does not disclose how/where the IP address of Figure 5 is obtained. In addition, Figure 5 shows that the IP address is below a "datagram multicast" layer, so there is no evidence that one of ordinary skill in the art would have been motivated to use -- let alone know how to adapt -- the techniques of a multicast system when implementing a "point-to-point communication link between the first and second processing units through the Internet." See Exhibit 1, Mayer-Patel Declaration, paragraph 64.

Furthermore, the Office Action has not shown that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with Vin and RFC 1531 without additional modifications to the combination -- modifications which are not specified in the Office Action. For example, the Office Action has not identified that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with RFC 1531 in light of the use of "leases" on network addresses within the RFC 1531 framework. As described in section 3.6 "If the lease expires before the host can contact a DHCP server, the host must immediately discontinue use of the previous network address and may inform local users of the problem." Similarly, section 4.4.4 of RFC 1531 states "If the lease expires before the client receives a DHCPACK, the client moves to INIT state, MUST immediately stop any other network processing and requests network initialization parameters as if the client were uninitialized. ... If the client is given a new network address, it MUST NOT continue using the previous network address and SHOULD notify the local users of the problem." However, if the system of the Etherphone papers and Vin is using "datagram multicast," how is the system supposed to deal with the loss of the address? Other processes that were communicating with the process that lost its lease would now begin sending the packets to the wrong destination. Moreover, by having to rely on another server that can go down such that a lease cannot be renewed and the computer has to stop using its network address, the combination is less robust than a system using static address assignment. Thus, the Office Action has not shown that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with Vin and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 65.

Thus, claim 43 is neither anticipated by the Etherphone papers nor rendered obvious by the Etherphone papers in combination with Vin and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 66.

Claim 44 recites "following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network." As discussed above with respect to claim 43, the Office Action has not shown how such a dynamic addressing technique is taught by the Etherphone papers or rendered obvious by the combination of the Etherphone papers and Vin and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 67.

Claim 44 also recites "querying the address server as to whether the second process is connected to the computer network." Exhibit N alleges that "conversations are established between two or more parties (Etherphones, servers, and so on) by performing remote procedure calls to the Voice Control Server." It then alleges that "when a first user at a first Etherphone (a first 'process') calls a second user at a second Etherphone (a second 'process'), the first Etherphone ***transmits a query*** in the form of a remote procedure call to determine the location of the second Etherphone." This allegation is made without citation to any portion of the Etherphone papers -- because the conclusion is not supported. See Exhibit 1, Mayer-Patel Declaration, paragraph 68.

As admitted in the Exhibit N, Swinehart states "The telephone control server manages voice switching by sending to each Etherphone or service the network addresses of the other participants. ... Thereafter, voice datagrams are transmitted directly among the participants, bypassing the control server." Neither the Exhibit N nor Swinehart discloses when this information is sent, thus there is no evidence that a query is sent to an address server (e.g., such that the network address information is returned as part of the result of the query). In fact, if the Voice Control Server were periodically sending out information to Etherphones or other services, there would be no need to "querying the address server as to whether the second process is connected to the computer network." Thus, this limitation is not inherently met by the Etherphone references, and the rejection of claim 44 should be withdrawn. See Exhibit 1, Mayer-Patel Declaration, paragraph 69.

As described above, claim 44 recites "querying the address server ***as to whether the second process is connected to the computer network***." As was discussed above with respect to claim 43, the Request and Exhibit N do not show how the Etherphone papers teach this tracking of on-line status. Thus, claim 44 is patentable over the Etherphone papers. See Exhibit 1, Mayer-Patel Declaration, paragraph 70.

<u>The rejection of claims 1-7 and 10-42 as obvious over the Etherphone papers in combination with at least one other reference</u>

Claims 1-7 and 10-42 have been rejected under 35 U.S.C. § 103(a) as obvious over the combination of the Etherphone papers in combination with (1) Vin and RFC 1531, (2) Pinard, (3) Vin, RFC 1531 and NetBIOS, and (4) Pinard and VocalChat User's Guide. Those rejections are respectfully traversed.

Claims 1, 2, 4-7 and 32-42 are rejected under 35 U.S.C. § 103(a) as being unpatentable over the Etherphone papers in view of Vin and RFC 1531. As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. See Exhibit 1, Mayer-Patel Declaration, paragraph 72.

Claim 1 also recites "program code for ***transmitting***, to the server, ***a query*** as to whether the second process is connected to the computer network." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining "whether the second process is connected to the computer network." Thus, claim 1 is not rendered obvious by the proposed combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 73.

Claim 2 recites "a memory, operatively coupled to the processor, for storing a network protocol address for selected of a plurality of processes, ***each network protocol address stored in the memory following connection of a respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. See Exhibit 1, Mayer-Patel Declaration, paragraph 74.

Claim 2 also recites "means, ***responsive to a query from the first process***, for determining the on-line status of the second process and for transmitting a network protocol address of the second process to the first process in response to a positive determination of the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach "determining the on-line status of the second process." Thus, claim 2 and its dependent claim 3 are not rendered obvious by the proposed combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 75.

Claim 4 recites "A. receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an on-line status with respect to the computer network, ***each of the network protocol addresses received following connection of the respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. See Exhibit 1, Mayer-Patel Declaration, paragraph 76.

Claim 4 also recites "B. ***receiving a query*** from the first process to determine the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach "determin[ing] the on-line status of the second process." Thus, claim 4 and its dependent claim 5-7 are not rendered obvious by the proposed combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 77.

Claim 32 recites "a. maintaining an Internet accessible list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of a process currently connected to the Internet, ***the Internet Protocol address added to the list following connection of the process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***currently connected*** to the Internet." Thus, claim 32 is patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 78.

Claim 33 recites "maintaining, in a computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, the network protocol address of the corresponding process assigned to the process upon connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***connected to*** the computer network." Thus, claim 33 and its dependent claims 34-37 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 79.

Claim 38 recites "program code configured to maintain, in the computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, ***the network protocol address of the corresponding process assigned to the process upon***

connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***connected to*** the computer network." Thus, claim 38 and its dependent claims 39-42 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 80.

Claims 10-30 are rejected under 35 U.S.C. § 103(a) as being unpatentable over the Etherphone papers in view of Pinard, either alone or in combination with the VocalChat User's Guide. Amended claim 11 recites "the caller process having a user interface and being operatively connectable to the callee process" and "establishing a point-to-point communication link ***from the caller process*** to the first callee process." When taken together, it is clear that the Request (including Exhibit N) and Office Action have not alleged, much less proven, that the caller process, having a user interface, establishes a point-to-point communication link with a first callee process. The Request states on page 119 that "Etherphone describes establishing a point-to-point communication link between a caller process and a callee process. ... Pinard discloses that a point-to-point communication link is established in response to a user associating an element representing the first callee process with the element representing a first communication line." Neither of those assertions actually state that any of the user interface elements described in the Request establish a point-to-point communication with a callee process. In fact, Exhibit N admits that the user interface elements are on the workstation not on the Etherphones that conduct the voice communications. Page 12 of Exhibit N directly states that the "GUI features [are] presented on the workstation display." See Exhibit 1, Mayer-Patel Declaration, paragraph 81.

In addition, the Office Action has not established that one of ordinary skill in the art would have made the proposed combination. The Office Action alleges that "it would have been obvious ... to utilize[e] the user-interface elements and interactions taught by Pinard in the invention taught by Etherphone since Pinard teaches that the invention can be used with any system in which a person computer in conjunction with a server operate." However, Pinard already discloses a personal computer 1 interacting with a server 5 without any indication that there is something missing from Pinard. Thus, without an indication that Pinard was somehow deficient, one of ordinary skill in the art would not have been motivated to include Etherphone with Pinard. Accordingly, amended claim 11 and its dependent claims 12-20 are not rendered obvious by the combination of NetBIOS and Pinard. See Exhibit 1, Mayer-Patel Declaration, paragraph 82.

Amended Claim 22, like amended claim 11, recites "***the caller process having a user interface*** and being operatively connectable to the callee process" and "establishing a ***point-to-point communication link from the caller process*** to the first callee process." As was set forth for the patentability of claim 11 above, that combination of elements is not taught by the applied combination of references. Thus, amended claim 22 and its dependent claims 23-31 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 83.

<u>The rejection of claims 1-7 and 10-42 over the combination of the VocalChat References, either alone or in combination with at least one other reference</u>

Claims 1-7 and 10-44 are rejected under 35 U.S.C. § 103(a) as obvious over VocalChat User's Guide in view of VocalChat Readme, VocalChat Networking, VocalChat Help File and VocalChat Troubleshooting Help file (collectively the "VocalChat References"), either alone or in combination with at least one other reference. Those rejections are respectfully traversed.

<u>The VocalChat References Are Not Printed Publications</u>

The Office Action appears to rely on, but does not expressly reference, Exhibit L of the Request for Re-examination (i.e., the Declaration of Alon Cohen), to establish that the

VocalChat References are, in fact, printed publications. However, the Office Action has not established that the VocalChat References constitute printed publications as required by statute. See 35 U.S.C. §§ 301 and 302.

As found by the Federal Circuit in *Carella v. Starlight Archery*, 804 F.2d 135, 139, 231 USPQ 644, 646-7 (Fed. Cir. 1986), "one who wishes to characterize the information, in whatever form it may be, as a 'printed publication' ... should produce sufficient proof of its dissemination or that it has otherwise been available and accessible to persons concerned with the art to which the document relates and thus most likely to avail themselves of its contents." (Citing *In re Wyer*, 655 F.2d 221, 227, 210 USPQ 790, 795 (CCPA 1981) as quoting *Phillips Electronics & Pharmaceutical Industries Corp. v. Thermal & Electronic Industries, Inc.*, 450 F.2d 1164, 1171, 171 USPQ 641, 646 (3rd. Cir. 1971).

Mr. Cohen states in paragraph 3 of his declaration that "the first version of the VocalChat product was commercially released to the public in 1993." However, this provides no indication of what information was distributed with that version (or even what the version number was of that version).

In paragraph 4 of his declaration, Mr. Cohen alleged that VocalChat 1.01 Networking Information "was publicly distributed in 1994 as part of the VocalChat version 1.01 software, which was commercially released and on sale to the general public in 1994." Mr. Cohen did not, however, allege the facts necessary to show that the files are actually printed publications For example, to whom was the software distributed, if anyone, outside of VocalTec? Second, how many copies were distributed and under what conditions? For example, were the copies distributed under a confidentiality agreement such that the associated files were not available to the general public? Were they distributed in such a way as to have been sufficiently available to one of ordinary skill in the art that she/he could have found them when trying to solve a similar problem? Without evidence on these factors, the mere allegation that VocalChat 1.01 Networking Information "was publicly distributed in 1994 as part of the VocalChat version 1.01

software, which was commercially released and on sale to the general public in 1994" is insufficient to show that this reference constitutes a printed publication.

Similarly, with respect to the VocalChat Help File and the VocalChat Troubleshooting Help file, Mr. Cohen alleges in paragraph 6 of his declaration that "Electronic copies of these documents were publicly distributed in 1994 as part of the VocalChat version 2.02 software, which was commercially released and on sale to the general public as a boxed product in 1994." However, this too fails to provide the same relevant facts required to make a prima facie case that the VocalChat Help file and VocalChat Troubleshooting Help file constitute printed publications.

As also described in *Carella*, "Although in some circumstances unsupported oral testimony can be sufficient to prove prior knowledge or use, it must be regarded with suspicion and subject to close scrutiny." 804 F.2d at 138, 231 USQP at 646. Although not disclosed in the declaration, the declarant, Mr. Cohen, is a paid consultant for the Defendants in the litigation relating to the patent in re-examination. See Exhibit 3 where the Court found Mr. Cohen to be a "consultant[] who the defendant has paid, see Deposition of Alon Cohen…" Mr. Cohen also co-founded a company named BitWine that partners with Defendant Skype. See Exhibit 4 (from http://techaddress.wordpress.com/2006/12/06/interview-with-alon-cohen-co-founder-and-co-ceo-of-bitwine.). Mr. Cohen also offers personal services to the public through the BitWine-Skype partnership. See Exhibit 5 (from http://www.bitwine.com/search?query=alon+cohen&=). Moreover, Mr. Cohen's company, VocalTec, produced Internet Phone, and the original patentee, NetSpeak, produced a competing product called WebPhone, thereby creating a potential for bias -- especially when at least one person compared the two products and stated "WebPhone may well become the killer app that puts to shame similar offerings from VocalTec (Internet Phone) and Quarterdeck (WebTalk). See Exhibit 6 (N2P-001-00005919).

TheVocalChat References Do Not Teach All of the Claim Limitations

Even assuming that the VocalChat References constitute printed publications (which has not been established), the combination of references still does not render obvious the claims under re-examination.

Claims 43 and 44

Claim 43 recites "program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network." The Office Action alleges that this limitation is met because "the network protocol address of clients are transmitted to and stored in a Connection List / USERS file located on the network." However, each VocalChat process actually reads the file and locally processes it, writing back the whole contents of the file when it has added its changes to it. Thus, in the context of multiple, independently-operating VocalChat clients each being able to read from and write to the "shared CONNLIST.VC" file, that file is not acting as a directory database, just a file whose contents could readily become inconsistent with the actual state of processes. For example, if a first process reads the file and then a second process reads the file, then the first process writes the file and then the second process writes the file, the changes written by the first process will be lost when the second process writes back its changes. Similarly, if the second process wrote back the file before the first, then the second process's changes would be lost when the first process wrote back its version of the file. Thus, the VocalChat References do not anticipate the claimed "program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network." See Exhibit 1, Mayer-Patel Declaration, paragraph 85.

In addition, claim 43 also recites "the network protocol address of each respective process forwarded to the database following connection to the computer network." This limitation is not taught by the VocalChat References, and the Office Action does not identify how this dynamic addressing is taught by the VocalChat References. Even assuming that the Office Action had intended to apply the combination of the VocalChat references and RFC 1531, the combination of references still would not have rendered obvious the subject matter of claim 43. See Exhibit 1, Mayer-Patel Declaration, paragraph 86.

With respect to claim 1, the Office Action asserts that "it would have been obvious to utilize the limitations taught by RFC 1531 in the invention taught by VocalChat ... since this allows for automatic reuse of an address that is no longer needed by the host to which it was assigned ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known ... and are useful to eliminate the burdensome task of manually assigning IP addresses for all network computers." The assignee respectfully submits that the Office Action is mistaken in its conclusion on motivation. In the context of point-to-point communication, widespread use of dynamically assigned addresses is not the solution to a problem, it is the problem itself. See Exhibit 1, Mayer-Patel Declaration, paragraphs 87-88. The assignee agrees that dynamically assigned addresses were known, and the patent in re-examination specifically states in that regard, "Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain." Col. 1, lines 53-56.

But it is not enough to prove that the cause of a problem existed. The Office Action must show by citation of prior art that the problem was recognized, and that the solution was either known or trivially apparent from the known art. See *Innogenetics, N.V. v. Abbott Laboratories*, 512 F.3d 1363, 1373 (Fed Cir. 2008) ("The district court was nevertheless correct that knowledge of a problem and motivation to solve it are entirely different from motivation to combine particular references to reach the particular claimed method.").

The development history of the VocalChat products indicates the opposite. See Exhibit 1, Mayer-Patel Declaration, paragraph 89. The Request cites a Generic version of the VocalChat client which, according to Mr. Cohen, was used on local area networks. See Cohen Declaration, paragraph 3. Absent from the Request, however, is any reference to the subsequent versions of

VocalChat that were released by VocalTec to the public for use on the Internet. The first of those versions was released in 1994, at least in beta, and was called VocalChat Gateway To Interent (or "VocalChat GTI"). This Internet version is believed to have required users to manually input callee addresses into static local address files. (See paragraph 393 of the Pre-Trial Order (filed with the IDS dated August 11, 2009) and Exhibit 7, SKYPE-N2P00286659.) Likewise, it is believed that VocalChat GTI did not utilize a server at all. See Pre-Trial Order at paragraph 390.

The use of manually input static addresses and the absence of a server suggests that the VocalTec designers—presumably software developers of at least ordinary skill in the art—did not consider the alleged combination of their own VocalChat references with RFC 1531, or it suggests that they did consider it but were unable to overcome the non-trivial obstacles to doing so. See Exhibit 1, Mayer-Patel Declaration, paragraph 90.

The next version of VocalChat was released soon thereafter and was also meant for use on the Internet. This version, again, did not combine the Request's disclosed versions of VocalChat with RFC 1531. Instead, it used the Internet Relay Chat (IRC) to help VocalChat clients with dynamically assigned IP addresses find one another. See Pre-Trial Order at paragraph 392 and Exhibit 7, SKYPE-N2P00286660. The development history of VocalChat—from the Generic version disclosed by the Request for use on local area networks to the GTI and IRC versions for use on the Interent—is strong, objective evidence of nonobviousness. If the designers of the VocalChat Generic implementation did not see fit to combine dynamic addressing with the implementation disclosed in the VocalChat references, it is respectfully submitted that one of ordinary skill in the art would not have done so either. Thus, the subject matter of claim 43 is not anticipated by or rendered obvious by the VocalChat References. See Exhibit 1, Mayer-Patel Declaration, paragraph 91.

Claim 44 recites "querying the address server as to whether the second process is connected to the computer network." The Office Action cites the Help file, pages 2 and 26, and the Network Information file, page 10, as disclosing this element and states "clients query the network server and … the server stores addresses of logged in users." However, the Office Action has not identified how the cited portions of those references teach the claimed "querying." At best, the references teach that a local process reads a "USERS" file or a Connections file. As can be seen from page 4 of the VocalChat Network Information (reproduced below), when the VocalChat system uses the Generic mode, a USERS file is used. See Exhibit 1, Mayer-Patel Declaration, paragraph 92.

2.5. Network parameters in the VocalChat INI files

These are the network parameters in the VocalChat VOCLCHAT.INI and VCSETUP.INI files (under the NetWork section):

[NetWork]	
NetWork=	/ description of the selected network
NetWorkUsers=	/ user services: NetWare / WinWorkgroups / Generic*
NetWorkProtocol=	/ network protocol: IPX / NetBIOS
NetWorkType=	/ name of the selected network, for future use
UsersFile=	/ path name of users file (when Generic is set)
MyUserName=	/ the name of the user (when Generic is set)**

* When Generic is set, a USERS file is used.

** This line appears only in the VOCALCHAT.INI file of each user.

The VOCLCHAT.INI files are in the windows directory of each user. The VCSETUP.INI file is in the VOCLCHAT directory, where VocalChat was installed, and is used only to supply default values for the different installations.

The USERS file configuration parameter includes a "UsersFile" entry that specifies the "path name of users file (when Generic is set)." However, it is also stated that "The VOCLCHAT.INI files are in the windows directory of each user." Thus, this "UsersFile" entry is a local configuration parameter such that the local VocalChat client reads and writes the USERS file on its own -- without performing the claimed query. See Exhibit 1, Mayer-Patel Declaration, paragraph 93.

Similarly, page 8 of the VocalChat Help file states "If your network type is not NetWare or Windows for Workgroups, the Setup program creates a Connection List file which is used to identify and access users." The Connection List file and the USERS file apparently have the same function. Thus, the identification and access enabled by the Connection List is performed by the local VocalChat client reading and writing the file itself -- without performing the claimed query. Accordingly, claim 44 is not rendered obvious by the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 94.

Claim 44 further recites "following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network." As discussed above with respect to the "querying" step, in the VocalChat Generic implementation, a local process reads a "USERS" file or a Connections file in its entirety and writes it back in its entirety rather than "forwarding to the address server a network protocol address at which the first process is connected to the computer network." This causes the VocalChat system to have to send an increasing amount of information as the number of users increases. Sending the whole file such that the new file replaces the old file also creates problems with consistency such that one user's changes could overwrite the changes of another user -- especially as networks got larger which would have increased the problem of inconsistent files being written. See Exhibit 1, Mayer-Patel Declaration, paragraph 95.

In addition, with respect to "following connection of the first process to the computer network forwarding ... a network protocol address at which the first process is connected to the computer network," the Office Action has not shown how this dynamic address-based limitation is taught by the VocalChat References. Thus, this limitation has not been shown to be taught by the applied references. See Exhibit 1, Mayer-Patel Declaration, paragraph 96.

Further, even if the Office Action had proposed a combination of the VocalChat References and RFC 1531, the combination would still not render obvious claim 44. As was discussed above with respect to claim 43, there is no motivation for combining VocalChat References and RFC 1531. Thus, claim 44 is not rendered obvious by the VocalChat

References, either alone or in combination with RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 97.

Claims 1-7 and 31-42

Claim 1 recites "program code for transmitting, to the server, a query as to whether the second process is connected to the computer network." As was discussed above with respect to claim 44, the VocalChat References do not teach "querying." In addition, the Office Action has not shown how the VocalChat references teach such a server capable of receiving a query. Thus, this limitation is not taught by the proposed combination. See Exhibit 1, Mayer-Patel Declaration, paragraph 98.

Claim 1 also recites "program code for transmitting to the server a network protocol address received by the first process ***following connection to the computer network***." As was discussed above with respect to claim 43, it would not have been obvious to combine the VocalChat References and RFC 1531. Thus, claim 1 is nonobvious in view of the applied references. See Exhibit 1, Mayer-Patel Declaration, paragraph 99.

Claim 2 recites "means, ***responsive to a query from the first process***, for determining the on-line status of the second process and for transmitting a network protocol address of the second process to the first process in response to a positive determination of the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the VocalChat References. See Exhibit 1, Mayer-Patel Declaration, paragraph 100.

Claim 2 further recites "each network protocol address stored in the memory ***following connection of a respective process to the computer network***." As was discussed above with respect to claim 43, the VocalChat References do not teach this dynamic address assignment. Further, it would not have been obvious to combine the VocalChat References and RFC 1531 for the reasons set forth above with respect to claim 43. Thus, claim 2 and its dependent claim 3 are patentable over the combination of the VocalChat References and RFC 1531. See Exhibit 1, Mayer-Patel Declaration, paragraph 101.

Claim 4 recites "A. receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an on-line status with respect to the computer network, ***each of the network protocol addresses received following connection of the respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 102.

Claim 4 also recites "B. ***receiving a query*** from the first process to determine the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the VocalChat references. Thus, claim 4 and its dependent claim 5-7 are not rendered obvious by the proposed combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 103.

Claim 32 recites "a. maintaining an ***Internet accessible*** list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of a process currently connected to the Internet." Exhibit O of the Request admits that "While VocalChat does not explicitly describe a server with stored names and addresses is accessible over 'the Internet,' it describes the use of TCP/IP, which is the protocol used on the Internet." Exhibit O then states that "VocalChat inherently describes that the list of users and network addresses is accessible over the Internet"; however, this is not inherently true. There may be any range of local area networks running TCP/IP communications that are not connected to the Internet. Thus, just because the Internet is a type of TCP/IP network does not mean that all TCP/IP networks must be the Internet. See Exhibit 1, Mayer-Patel Declaration, paragraph 104.

In fact, the security and privacy concerns that arise in a secure, local area network are very different than the issues that arise on the Internet. The VocalChat References describe that "All users must use the same Post-Office, otherwise they won't be able to communicate or leave messages to each other. This means that all users must ... have write permission for the Post-Office directory." This would mean opening up one's network to outsiders across the Internet. The Office Action has not cited any evidence that one of ordinary skill in the art would have been motivated to do so in light of the inherent risks. Further, the VocalChat development history discussed above is illuminating about how one of ordinary skill in the art would have modified the VocalChat system. The VocalChat designers went from a Generic implementation with a shared directory with a shared file that can be remotely updated, to locally processed files that were manually updated in the VocalChat GTI version. Thus, the trend on how to modify the VocalChat system was the opposite of what is being proposed. See Exhibit 1, Mayer-Patel Declaration, paragraph 105.

Claim 32 also recites "***the Internet Protocol address added to the list following connection of the process to the computer network.***" As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references. Thus, claim 32 is patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 106.

Claim 33 recites "maintaining, in a computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, the network protocol address of the corresponding process assigned to the process upon connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Thus, claim 33 and its dependent claims 34-37 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 107.

Claim 38 recites "program code configured to maintain, in the computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, ***the***

network protocol address of the corresponding process assigned to the process upon connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Thus, claim 38 and its dependent claims 39-42 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 108.

Amended claim 11 recites "c.1 querying the server as to the on-line status of the first callee process; and c.2 receiving a network protocol address of the first callee process over the computer network from the server." As was discussed above with respect to claim 44, the VocalChat References do not disclose such a querying step. Thus, amended claim 11 and its dependent claims 12-20 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 109.

Amended claim 22 recites "program code for querying the server as to the on-line status of the first callee process; and program code for receiving a network protocol address of the first callee process over the computer network from the server." As was discussed above with respect to claim 44, the VocalChat References do not disclose such a querying step. Thus, amended claim 22 and its dependent claims 23-30 are patentable over the applied combination of references. See Exhibit 1, Mayer-Patel Declaration, paragraph 110.

Objective Evidence of Non-Obviousness

In addition to the reasons set forth above showing that all of the elements of the claims under re-examination are not taught by the applied references, it is respectfully submitted that objective evidence supports a finding that the claims are non-obviousness. Objective indicia of non-obviousness, which include commercial success, licenses showing industry respect, and the failure of others, "provide evidence of how the patented device is viewed by the interested public: not the inventor, but persons concerned with the product in the objective arena of the marketplace." *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966); *WMS Gaming Inc. v.*

International Game Tech., 184 F.3d 1339, 1359 (Fed. Cir. 1999); *Arkie Lures, Inc. v. Gene Larew Tackle, Inc.*, 119 F.3d 953, 957 (Fed. Cir. 1997). Evidence supporting the objective indicia of non-obviousness is set out below.

Commercial Success

NetSpeak's WebPhone, an exemplary embodiment of the '704 patent (see, e.g., col. 3, ll. 6-10), was a commercial success as evidenced by the recognition it received in the industry. WebPhone's commercial success is attributable to the novelty and non-obviousness of the invention. *Demaco Corp. v. F. Von Langsdorff Licensing, Ltd.*, 851 F.2d 1387, 1393 (Fed. Cir. 1988) ("A prima facie case of nexus is generally made out when the patentee shows both that there is commercial success, and that the thing (product or method) that is commercially successful is the invention disclosed and claimed in the patent.").

NetSpeak's WebPhone won Internet Telephony's 1998 Product of the Year in the category of Internet Telephony Clients. Exhibit 8, page 6 (N2P-200-00012627).

NetSpeak's WebPhone product also won significant praise when compared to other products in the same timeframe. "WebPhone may well become the killer app that puts to shame similar offerings from VocalTec (Internet Phone) and Quarterdeck (WebTalk). See Exhibit 6 (N2P-001-00005919).

The importance of the claimed invention can also be seen in its praise by other companies in the industry. In a joint press release of NetSpeak and Durand Communications Network ("Durand"), Durand's president and CEO stated "NetSpeak's WebPhone is hands-down the best PC-voice communications package available in the market today. ... We wanted to work with a company whose leading edge technology would add value to our existing MindWire NT CommunityServer by offering unique telephony services so integral to fostering growth within online communities." Exhibit 9, page 1.

NetSpeak's WebPhone was also praised in the Computer Telephony Magazine. The July 1996 Edition included an article on the WebPhone trial version and stated "You've gotta try this

Internet telephony package. NetSpeak ... makes WebPhone. ... Does it work? Yes." Exhibit 10 (N2P-200-00012630).

As set forth in the original Assignee's Amended S-1 Registration form (Exhibit 11), NetSpeak's technology was a commercial success as further evidenced by the investments made in the company. At least three different stock offerings were made which raised millions of dollars for the company. The Amended S-1 Registration form describes on numbered page 19:

> In January and February of 1996, the Company sold 1,204,000 shares of Common Stock at $2.50 per share in a private offering raising $2,992,028 ....
>
> In June 1996, the Company issued 207,679 shares of Common Stock to Creative at a price of $5.05 per share raising $943,698 ....
>
> In August 1996, the Company issued 769,853 shares of Common Stock and the Motorola Warrant to purchase up to an additional 452,855 shares of Common Stock at a price of $5.50 per share for a six year period expiring in August 2002 to Motorola raising $3,993,864....

Later, in 1998, Motorola took an even larger interest in NetSpeak by acquiring an additional 27% of the stock that it did not already own at a cost of $90 million. See Press Release, Exhibit 12 (N2P-200-00012891). See also March 30, 1998 article from Telephony online describing strategic alliance between Motorola and NetSpeak. See Exhibit 13 (N2P-102-00000048).

Also in 1998, the Company issued approximately 1.3 million shares of common stock to Bay Networks for $36.8 million. See Exhibit 14, NetSpeak Form 10-K for the Fiscal Year ending December 31, 1997.

See also, the 8-K related to the acquisition of NetSpeak by Net2Phone. Exhibit 15.

As more fully detailed in NetSpeak's 10-K for Fiscal Year 1997 (Exhibit 14), NetSpeak's communications technology was a commercial success as further evidenced by the strategic alliances it made with "with leaders in various segments of the telecommunications and networking industries," including Siemens (whereby Siemens agrees to market NetSpeak's "IP telephony server products"), Bay Networks (whereby the Company agrees "not to provide its source code to…competitors for a period of three years), Fujitsu and Rockwell International (whereby NetSpeak was "integrating its software into the[se] companies' proprietary hardware platforms"), MCI (see Exhibit 16 announcing that MCI signs contract with NetSpeak to incorporate WebPhone in networkMCI Click'NConnect Web-Based Service) and NTC (whereby NetSpeak would "supply IP telephony products and systems"), and others.

NetSpeak's WebPhone client software products were a commercial success as further evidenced by the number and extent of the channels through which they were sold, including "distribution agreements with over 900 ISPs worldwide." See Exhibit 14, 10-K cited above. Details of the operation of the WebPhone client can be found in Exhibits 17 and 18. For example, Exhibit 17 states "the CS [i.e., connection server] updates the user e-mail address, IP address, and online status fields, and uses them to perform IP address resolution and track account activation information. … When a user calls … using a WebPhone, the CS is used to resolve the target e-mail address to an IP address." Similarly, Exhibit 18 states "Connection and Information servers are the addresses here at NetSpeak that your WebPhone uses to find and call other parties. …Connection Server: is used when you dial someone by e-mail address. If you try to dial someone by e-mail address, the WebPhone, calls the connection server, matches the desired e-mail address to an IP address, disconnects from the Connection server, and dials the IP address."

Licenses Showing Industry Respect

In connection with Motorola's 1998 investment described above, and as set forth more fully in the NetSpeak Form 10-K for the Fiscal Year 1997 (Exhibit 14), NetSpeak and Motorola entered into a joint development and licensing agreement pursuant to which the two companies would seek to join their technologies to enable Internet Protocol multimedia communications on wireless networks. Under that agreement, Motorola obtained a license to develop RF products using NetSpeak's technology, to include NetSpeak's technology in wireless devices such as cellular phones, pagers, satellite phones and two way radios to support real-time multimedia communications (voice, audio, video, data, etc.), and to manufacture and sell NetSpeak products. See description of NetSpeak's technology at page 6 of Exhibit 14 under the header "NETSPEAK'S CORE COMMUNICATIONS TECHNOLOGY" (reciting, *inter alia*, "allows users to connect to other users in a point-to-point fashion, rather than through an intermediate routing mechanism."). NetSpeak's licenses included a license to the WebPhone product and network address resolution technology, see Exhibit 14, which are commercial embodiments of the patented claims. NetSpeak's success in licensing is attributable to the novelty and non-obviousness of the invention. *Demaco Corp.*, 851 F.2d at 1393.

Failure of Others

The inventions claimed in the '704 Patent resolved the problem of locating a computer process connected to a network, where the computer process was assigned a temporary network address. See, e.g., specification at col. 5, lines 21-24. Each time a particular computer process connected to the network, it would have a different address. Such addresses were largely a by-product of the near-universal adoption of the Dynamic Host Configuration Protocol ("DHCP"), described in RFC 1531. As discussed above, others, including the developers of the VocalChat references cited by the Request and the writer of Exhibit 23, attempted to resolve the problem of locating a computer process with a dynamically assigned address and failed to suggest the claimed methods.

Recognition in the Patent Literature

The Federal Circuit has left itself open to acknowledging that the patent citations of later patent applicants and examiners can be objective evidence of an earlier patent's nonobviousness. See *In re: Mettke*, 570 F.3d 1356, 1361 (Fed. Cir. 2009). This position is supported by the academic literature. See, e.g., Trajtenberg, Manuel , "A Penny for Your Quotes: Patent Citations and the Value of Innovations," The RAND Journal of Economics, Vol. 21, No. 1 (Spring 1990), pp. 172-187 at 174. ("Thus, if citations keep coming, it must be that the innovation originating in the cited patent had indeed proven to be valuable.") (Exhibit 19.) As shown in Exhibit 20, according to the USPTO's own records, the '704 patent and its continuations and divisionals have been cited in 76 issued patents. This supports an inference that the '704 patent in re-examination advanced the art in a nonobvious way that was neither cumulative of the art that came before it nor predictable in its view.

This inference of nonobviousness is especially compelling over the NetBIOS references. Not one issued patent that cites the patent in re-examination (or one of its related patents) also cites a NetBIOS reference. See Exhibit 21 (including variations on the name for NetBIOS such that it includes RFC 1001 and RFC 1002). This phenomenon is especially significant given that NetBIOS is a well known piece of networking art that has been cited frequently in the patent literature -- 33 times according to the USPTO's records.[3] The assignee respectfully submits that there is a simple explanation for this otherwise highly improbably dichotomy: NetBIOS and the patent in re-examination do not overlap because the scope and content of what they disclose are distinct.

The assignee also notes in this regard that the cover page of U.S. Patent No. 6,389,127, assigned to ICQ Inc., an unrelated company, and entitled "Telephone Status Notification System," references the '704 Patent in re-examination, but does not cite to any of the references submitted in the Request. Their absence from the ICQ patent is especially significant since both

[3] In fact, there are 43 references to NetBIOS if the search includes any of: NetBIOS, RFC 1001, RFC 1002, NBT and NetBT (excluding references to "NBT" in the medical field). See Exhibit 22.

NetBIOS and Etherphone are well known pieces of art, and each has been cited frequently in the patent literature—33 times and 135 times, respectively. The assignee respectfully submits that there is a simple explanation for this difference: the references in the Request were not cited by the ICQ patent because they did not teach anything plausibly related to "Status Notification," whereas 6,108,704 was cited because it plainly did.

Consequently, in light of the above discussions, the outstanding grounds for rejection are believed to have been overcome and the patentability of the claims subject to re-examination should be indicated as confirmed. An early and favorable action to that effect is respectfully requested.

CHARGE STATEMENT: Deposit Account No. 501860, order no. **2655-0188**.

The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (missing or insufficiencies only) now or hereafter relative to this application and the resulting Official Document under Rule 20, or credit any overpayment, to our Accounting/ Order Nos. shown above, for which purpose a duplicate copy of this sheet is attached.

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal sheet is filed.

CUSTOMER NUMBER 42624	Respectfully submitted, By: / Michael R. Casey / Michael R. Casey, Ph.D. Registration No.: 40,294

Davidson Berquist Jackson & Gowdey LLP
4300 Wilson Blvd., 7th Floor, Arlington, Virginia 22203
Main: (703) 894-6400 • FAX: (703) 894-6430

Electronic Acknowledgement Receipt

EFS ID:	6532617
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael Raymond Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	27-NOV-2009
Filing Date:	17-FEB-2009
Time Stamp:	16:58:34
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	NPL Documents	Exhibit_02_Web_Security.pdf	556594 4bd74423389a55af9ac258de7e54aa5b64f82a6d	no	4

Warnings:

Information:

2	NPL Documents	Exhibit_03_ALONCOHEN.pdf	580490 94578c7fd6d923478abb785d39ae17761d08296b	no	6
Warnings:					
Information:					
3	NPL Documents	Exhibit_04_Alon_Cohen_Interview.pdf	228810 be6d458709bfb99e535798ef246de67d832430f0	no	6
Warnings:					
Information:					
4	NPL Documents	Exhibit_05_Alon_Cohen_Bio.pdf	146074 99bf6aa049b59daabb4e9a5b5d2e50284e6db580	no	2
Warnings:					
Information:					
5	NPL Documents	Exhibit_06_WebPhoneKillerApp.pdf	456394 4848fd08e2b060256d3efd20838c7f3a4b5dcd38	no	5
Warnings:					
Information:					
6	NPL Documents	Exhibit_07_VocalChatHistory.pdf	556244 1f2972379c1ff1f97726a600fb7cb1acb8b03767	no	16
Warnings:					
Information:					
7	NPL Documents	Exhibit_08_InternetTelephony_POTYear.pdf	373169 c6f517edf4328aba54c24f7335ac1aeca3d852c5	no	9
Warnings:					
Information:					
8	NPL Documents	Exhibit_09_Durand_PressRelease.pdf	159588 cadcdd5dd535e29b0f018f001bdc3f5f90926a5d	no	4
Warnings:					
Information:					
9	NPL Documents	Exhibit_10_Computer_Telephony.pdf	67848 73983ba053354169aed612bd4c5c16bedc588635	no	2
Warnings:					
Information:					
10	NPL Documents	Exhibit_11_NetSpeak_S-1A.pdf	533615 d519038515799365915b09680239d470ebd3dfb7	no	119
Warnings:					
Information:					

11	NPL Documents	Exhibit_12_Motorola_PressRelease.pdf	143729 5be5574cb711e71e2236ba7fe0266926c8dfe28d	no	3
Warnings:					
Information:					
12	NPL Documents	Exhibit_13_Motorola2.pdf	404162 3dadaf91670273b74a64265b6c264a024c3a89ba	no	4
Warnings:					
Information:					
13	NPL Documents	Exhibit_14_NetSpeak_10K.pdf	144246 11cf8d3b11ba77c1aba129d0742ee1ba3e5663f3	no	37
Warnings:					
Information:					
14	NPL Documents	Exhibit_15_Net2Phone_8-K.pdf	170814 0aca9a1a4c1076a5d6763cd2431ec48a3ecf24be	no	40
Warnings:					
Information:					
15	NPL Documents	Exhibit_16_MCI.pdf	362841 0a897704787d9f249e440324fb2a62e452b7d76	no	3
Warnings:					
Information:					
16	NPL Documents	Exhibit_17_NetSpeak_CS.pdf	1107085 32c9865c4181a0280e3d93dcb9a48bf545c33343	no	7
Warnings:					
Information:					
17	NPL Documents	Exhibit_18_How_do_I.pdf	2719628 6d1ba9f24834940f8748db648a9ec88e8bd26694	no	16
Warnings:					
Information:					
18	NPL Documents	Exhibit_19_Penny.pdf	3498198 41fb5202c9ecf9a1e3ccecd93c7d6bfe32ce37bc	no	18
Warnings:					
Information:					
19	NPL Documents	Exhibit_20_704_and_family_results.pdf	531202 7616363ed3875788f07fbf116e3afbd8929f72d8	no	4
Warnings:					
Information:					

20	NPL Documents	Exhibit_21_NetBIOS_ReferencesWith_704.pdf	111308 b48c9c979e6ad1cff00d337072b8ce2e2e8448cc	no	2
Warnings:					
Information:					
21	NPL Documents	Exhibit_22_NetBIOS_References.pdf	425223 36870b9db80692a5b18003d908348732fd145f6f	no	3
Warnings:					
Information:					
22	NPL Documents	Exhibit_23_Dynamic_IP_Addresses.pdf	371495 c1dbda9767483c48a40ac518b72d5b6943bbc475	no	5
Warnings:					
Information:					
23	Rule 130, 131 or 132 Affidavits	Exhibit_01_Ketan_704_Declaration.pdf	7412403 a04d64f15529c560e8b93e7236fc0b1beb0ebb6	no	50
Warnings:					
Information:					
24	Amendment/Req. Reconsideration-After Non-Final Reject	20091127_cover.pdf	82103 a1dd5a76e6e7344e8e1bd6c3abaac9b1ec3cd7dc	no	1
Warnings:					
Information:					
25	Claims	20091127_claims.pdf	468863 f0626d48f22b75defce91195442a91b1813ec987	no	4
Warnings:					
Information:					
26	Applicant Arguments/Remarks Made in an Amendment	20091127_remarks.pdf	7153482 0fcbe1ff1f495288d345efe97fa80d5f33ed09b4	no	41
Warnings:					
Information:					
27	Reexam Certificate of Service	20091127_COS.pdf	63994 409d88c7a3a40be3cd2ed363d2f21569f0c3c3b9	no	1
Warnings:					
Information:					
			Total Files Size (in bytes):	28829602	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Exhibit 1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION OF: Net2Phone, Inc. (Patent No. 6,108,704)

Control No.: 90/010,416

Issue Date: August 22, 2000

Title: **POINT-TO-POINT INTERNET PROTOCOL**

Attorney Docket: 2655-0188

Group Art Unit: 3992

Examiner: KOSOWSKI, Alexander

Confirmation No.: 1061

DECLARATION OF KETAN MAYER-PATEL UNDER 37 C.F.R. 1.132

Hon. Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

I. INTRODUCTION

1. I have been retained as an independent expert witness by Net2Phone, Inc., the assignee of the patent presently undergoing re-examination (i.e., U.S. Patent No. 6,108,704 (hereinafter "the '704 patent")).

2. I am an expert in the field of networking protocols including networking protocols supporting multimedia streams including digital audio data. See Curriculum Vitae attached as Exhibit 1.

3. I received Bachelors of Arts degrees in Computer Science and Economics in 1992, a Masters of Science in 1997 from the Department of Electrical Engineering and Computer Science and a Ph.D. in 1999 from the Department of Electrical Engineering and Computer Science, all from the University of California, Berkeley.

4. I received the National Science Foundation CAREER Award in 2003 while an Assistant Professor at the University of North Carolina, Chapel Hill.

5. I have had extensive experience in both industry and academia as it relates to the technical fields relevant here. For example, I have been a programmer, a visiting researcher, and an Assistant and Associate professor.

6. I am a co-author of numerous articles that have appeared in a number of refereed publications and proceedings.

7. Governmental agencies, such as the National Science Foundation and the Office of Naval Research, have provided funding for my research.

II. RETENTION AND COMPENSATION

8. I have been retained to offer an expert opinion on the prior art relevant to the '704 patent (and other patents currently under re-examination) and the validity of the claims undergoing re-examination.

9. My work on this case is being billed at a rate of $400 per hour, with reimbursement for actual expenses. My compensation is not contingent upon the outcome of the case.

III. BASIS OF MY OPINION AND MATERIALS CONSIDERED

10. In preparation for this report, I have considered and relied on data or other documents identified in this report. For example, I have reviewed the Office Action dated August 27, 2009 as well as the Request for Re-examination that was filed for the '704 patent including the Exhibits to the Request for Re-examination. I have also reviewed the file history of the '704 patent.

11. I have familiarized myself with the state of the art at the time the '704 patent was filed by reviewing both patent and non-patent references from prior to the filing date of the application that became the '704 patent.

12. My opinions are also based upon my education, training, research, knowledge, and experience in this technical field.

IV. SUMMARY OF MY OPINIONS

13. Based on my prior experience in the field of computer systems and networking, including network communication protocols, and based on my review of the documents relating to the

pending re-examination proceeding, I have developed an understanding of the '704 patent and the claimed inventions.

14. I have been asked to compare the claims of the '704 patent to the references applied in the outstanding Office Action. The results of my comparison are provided below.

15. I understand that the assignee's response includes the following amendments to the claims:

10. (Canceled)

11. (Amended) <u>In a computer system, a method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the method comprising the steps of:</u>

<u>A. providing a user interface element representing a first communication line;</u>

<u>B. providing a user interface element representing a first callee process; and</u>

<u>C. establishing a point-to-point communication link from the caller process to the first callee process, in response to a user associating the element representing the first callee process with the element representing the first communication line,</u> [The method of claim 10] wherein step C further comprises the steps of:

c.1 querying the server as to the on-line status of the first callee [process] process; and

c.2 receiving a network protocol address of the first callee process over the computer network from the server.

12. (Amended) The method of claim [10] 11 further comprising the step of:

D. providing an element representing a second communication line.

14. (Amended) The method of claim [10] 11 further comprising the steps of:

D. providing a user interface element representing a second callee process; and

E. establishing a conference point-to-point communication link between the caller process and the first and second callee process, in response to the user associating the element representing the second callee process with the element representing the first communication line.

15. (Amended) The method of claim [10] 11 further comprising the step of:

F. removing the second callee process from the conference point-to-point communication link in response to the user disassociating the element representing the second callee process from the element representing the first communication line.

16. (Amended) The method of claim [10] 11 further comprising the steps of:

D. providing a user interface element representing a communication line having a temporarily disabled status; and

E. temporarily disabling a point-to-point communication link between the caller process and the first callee process, in response to the user associating the element representing the first callee process with the element representing the communication line having a temporarily disabled status.

19. (Amended) The method of claim [10] 11 wherein the caller process further comprises a visual display and the user interface comprises a graphic user interface.

21. (Canceled)

22. (Amended) A computer program product for use with a computer system comprising:

a computer usable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the medium further comprising:

program code for generating an element representing a first communication line;

program code for generating an element representing a first callee process;

program code, responsive to a user associating the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process, [The computer program product of claim 21] wherein the program code for establishing a point-to-point communication link further comprises:

program code for querying the server as to the on-line status of the first callee process; and

program code for receiving a network protocol address of the first callee process over the computer network from the server.

23. (Amended) A computer program product of claim [21] 22 further comprising:

program code for generating an element representing a second communication line.

25. (Amended) The computer program product of claim [21] 22 further comprising:

program code for generating an element representing a second callee process; and

program code means, responsive to the user associating the element representing the second callee process with the element representing the first communication line, for establishing a conference communication link between the caller process and the first and second callee process.

27. (Amended) The computer program product of claim [21] 22 further comprising:

program code for generating an element representing a communication line having a temporarily disabled status; and

program code, responsive association of the element representing the first callee process with the element representing the communication line having a temporarily disabled status, for temporarily disabling the point-to-point communication link between the caller process and the first callee process.

30. (Amended) A computer program product of claim [21] 22 wherein the computer system further comprises a visual display and the user interface comprises a graphic user interface.

16. In general, it is my opinion that all of the pending claims undergoing re-examination (i.e., claims 1-7, 11-20 and 22-44) are patentable over the applied references for at least the reasons set forth below.

The rejection of the claims over NetBIOS, either alone or in combination with at least one other reference

17. Claims 43 and 44 were rejected under 35 U.S.C. § 102 as being anticipated by Protocols for X/Open PC Interworking SMB, Version 2, The Open Group (1992) (hereinafter "NetBIOS").

18. I understand that a claim is rejected under 35 U.S.C. § 102(b) when an examiner believes that each and every limitation of the claim is taught by the applied reference.

19. Pending claims 1-7, 11-20 and 22-42 were rejected under 35 U.S.C. § 103(a) as being obvious over NetBIOS in combination with at least one other reference.

20. I understand that a rejection under 35 U.S.C. § 103(a) means that an examiner believes that although no single reference includes all of the claimed limitations, nonetheless the combination of references made by the examiner would have been obvious to one of ordinary skill in the art at the time the invention was made.

The Rejection of Claims 43 and 44 over NetBIOS

21. Claim 43 recites "a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network." This limitation is not taught by the NetBIOS reference as NetBIOS does not provide dynamic addressing or on-line status.

22. In rejecting claim 43, the Office Action adopts the positions of the third-party requester and states "See claim mapping chart in Exhibit M, pages 36-40, incorporated by reference."

With respect to this limitation (a), the claim mapping does not show that NetBIOS teaches "the network protocol address of each respective process forwarded to the database ***following connection to the computer network***." In fact, the Office Action appears to have agreed (e.g., with respect to claim 1) that the NetBIOS reference does not teach that the processes receive network protocol addresses "following connection to the computer network." The Office Action did this by rejecting the requester's arguments under 35 U.S.C. § 102 and instead adopting "claim chart mapping utilizing alternative 103 rejections" -- rejections that rely on RFC 1531 to teach dynamic addressing.

23. Even assuming that the Office Action intended the rejection to be a rejection under 35 U.S.C. § 103 by combining NetBIOS with RFC 1531, the rejection would still not show that the claims were unpatentable. When alleging that one of ordinary skill in the art would have combined NetBIOS with RFC 1531, the Office Action states "it would have been obvious ... to utilize the limitations taught by RFC 1531 in the invention taught by NetBIOS ...since this would allow for automatic reuse of an address ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known ... and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked computers."

24. I do not agree with the conclusion drawn by the Office Action on the combinability of NetBIOS and RFC 1531. The Office Action speculates, with hindsight, as to why a person of ordinary skill might want to combine the two references, but does not acknowledge the problems that would arise in doing so, and does not provide any prior art that would indicate how the problems that dynamic addressing would bring into a NetBIOS type system could be resolved by those of ordinary skill at the time the patent was filed. In the context of point-to-point communication, widespread use of dynamically assigned addresses does not solve NetBIOS's problems, it creates further problems.

25. Dynamically assigned addresses were known, and the patent in re-examination specifically states in that regard, "Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video

have been generally difficult to attain." Col. 1, lines 53-56. However, the Office Action has not shown that one of ordinary skill in the art would have made the proposed combination of dynamically assigned addresses with NetBIOS.

26. Section 15.1.7 of the NetBIOS reference (entitled "Consistency of the NBNS Data Base") recognizes that the association between a node, a registered name and an IP address is tenuous, even in an environment that uses static IP addresses. "Even in a properly running NetBIOS scope the NBNS and its community of end-nodes may occasionally lose synchronization with respect to the true state of name registrations." To minimize the impact of this problem, the reference states, "Various approaches have been incorporated into the NetBIOS-over-TCP protocols" which it then proceeds to describe.

27. However, by incorporating DHCP and adopting dynamic address allocation as used by Internet access providers, the synchronization problem would become more disruptive, not less. Dynamic addressing would have introduced a new uncertainty to the relationships among the NBNS and its community of end-nodes and a new set of obstacles to NetBIOS synchronization that ***are not addressed by the NetBIOS reference***. Consider the case of a node that is turned-off and then subsequently turned back on, or a node that has simply lost its Internet connection for some technical reason or whose DHCP lease has expired and then re-established a connection. In a dynamic addressing environment, such a node would most likely obtain a new IP address when it was turned back on that was different than the one it had when it registered its name. This change could lead to any number of node-name-IP address synchronization problems for the disclosed NetBIOS protocols.

28. For example, because the NBNS does not know the node's new address, the NBNS would be unable to send to the node a Name Release Request or a Name Conflict Demand or request that the node send it a Name Status Request. Because communication from the node would be originating at a new address that was not recognized by the NBNS, a node's response to a Name Query Request (assuming it somehow knew that its name had been challenged, perhaps from before it lost network connectivity) would not be recognized. A node would also

be unable to confirm its association with registered names by sending Name Refresh Request packets to the NBNS. If a session between two NetBIOS applications were cut-off, re-establishing the communication would be especially difficult where the ability of a called entity to obtain both its associated name and its associated IP address were in doubt. As a result, the Office Action has not demonstrated that a solution to the problems created by exposure of NetBIOS to DHCP and dynamic addressing has been addressed by any of the applied references.[1]

29. The Office Action also has not identified anything in the cited art that suggests how a person of ordinary skill is to go about the redesign of NetBIOS and the solving of obstacles to NetBIOS operation that are created by Internet access; problems that were recognized and left as warnings unresolved in the NetBIOS reference.[2]

30. In view of the foregoing, claim 43 is patentable over NetBIOS alone or over NetBIOS in combination with RFC 1531.

31. NetBIOS and RFC 1531 also do not teach "a. program code configured to access a directory database…having a network protocol address for a selected plurality of processes ***having on-line status with respect to the computer network.***" While NetBIOS uses name entries with "active" statuses as part of its name management process, an analysis of how that "active" status is used shows that "an active name" is not synonymous with an "on-line status with respect to the computer network." An active name simply refers to a name that has been

[1] Besides dynamic addressing, Internet access would pose other challenges to a NetBIOS system. For example, because NetBIOS was designed for use on local area networks with small numbers of computers, trust among the network participants is assumed. That assumption cannot be transferred to a global Internet made up of unknown, and sometimes malevolent, entities. An implementation of NetBIOS on the public Internet would necessitate non-trivial adaptations to ensure that its services perform correctly and return accurate information. There is no discussion of security issues in the cited references. See Exhibit 2, from http://www.w3schools.com/Site/site_security.asp which instructs Microsoft Windows users whose computers access the Internet to disable NetBIOS over TCP/IP in order to solve their security problems.

[2] The cited references go out of their way *to avoid* describing how a NetBIOS protocol might work in inter-connected network environments that that are *less complex* than the Internet and that *predate* DHCP. See Section 4.6 ("The proposed standard recognizes the need for NetBIOS operation across a set of networks interconnected by network (IP) level relays (gateways.) However, the standard assumes that this form of operation will be less frequent than on the local MAC bridged-LAN.")

registered and that has not yet been de-registered, independent of whether the associated computer is or is not on-line. As shown on page 447 (and reproduced below), the Node_Name entries stored with respect to a NetBIOS Name Server contain a series of fields including the "ACT" field.

```
The NAME_FLAGS field:

                                          1   1   1   1   1   1
  0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| G |  ONT  |DRG|CNF|ACT|PRM|          RESERVED                 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

The NAME_FLAGS field is defined as:

Symbol      Bit(s)   Description:

RESERVED     7-15    Reserved for future use.  Must be zero (0).
PRM             6    Permanent Name Flag.  If one (1) then entry
                     is for the permanent node name.  Flag is zero
                     (0) for all other names.
ACT             5    Active Name Flag.  All entries have this flag
                     set to one (1).
CNF             4    Conflict Flag.  If one (1) then name on this
                     node is in conflict.
DRG             3    Deregister Flag.  If one (1) then this name
                     is in the process of being deleted.
ONT           1,2    Owner Node Type:
                        00 = B node
                        01 = P node
                        10 = M node
                        11 = Reserved for future use
G               0    Group Name Flag.
                     If one (1) then the name is a GROUP NetBIOS
                     name.
                     If zero (0) then it is a UNIQUE NetBIOS name.
```

32. The ACT field is a single bit field (in bit 5) that signifies an "Active Name Flag. ***All entries have this flag set to one (1).***" (Emphasis added.) If all name entries have this flag set to one (1), then the NetBIOS name server cannot be using the Active Name Flag as a means of separately tracking whether the entity that owns the name is "active," let alone what its "on-line status" might be.

33. The NetBIOS reference also does not teach that the active status of a name in the NetBIOS server is an indication of the active status of the owner of that name. To the contrary, when information about whether the owner of a name is "active" may be relevant, for example when a new entity seeks to register a name that has already been registered in the NetBIOS name server, the NetBIOS reference describes an elaborate set of interactions used to test whether the existing owner of the registered name is active or inactive. It does not rely on the fact that the name is active in the NetBIOS name server (See Section 15.2.2.2 and 15.2.2.3 entitled "Existing Name and Owner is Inactive").

34. The NetBIOS reference also does not teach that an acquired IP address can be reasonably relied upon by a requesting end-node to confirm the "on-line status" of an end-node associated with a sought name. The NetBIOS reference describes at least two different scenarios where a second end-node sends a ***rejection response*** to the first end-node ***notwithstanding the fact that a second end-node is connected*** to the computer network and active with respect to the sought name. See Section 16.1.1 ("There exists a NetBIOS LISTEN compatible with the incoming call, but there are inadequate resources to permit establishment of a session...The called name does, in fact, exist on the called node, but there is no pending NetBIOS LISTEN compatible with the incoming call."). No distinction is made in the reference between the rejection response in these cases and the rejection response in cases where the called name does not exist on the called end-node. See also section 16.1.1 which state "In all but the first case, a rejection response is sent back over the TCP connection to the caller."

35. Thus, the limitation "program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network" of claim 43 is not taught by the NetBIOS reference alone or in combination with RFC 1531, and I believe that the patentability of claim 43 should be confirmed.

36. Claim 44 recites "***following connection of the first process to the computer network*** forwarding to the address server a network protocol address at which the first process is connected to the computer network." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach the claimed dynamic address assignment. Moreover, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

37. Claim 44 also recites "querying the address server as to ***whether the second process is connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "whether the second process is connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 44 is patentable over both NetBIOS alone and the combination of NetBIOS and RFC 1531.

<u>Claims 1-7 and 32-42</u>

38. Claim 1 recites "program code for transmitting to the server ***a network protocol address received by the first process following connection to the computer network***." Claim 1 further recites "program code for receiving a network protocol address of the second process from the server, ***when the second process is connected to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

39. Claim 1 also recites "program code for transmitting, to the server, a query as to ***whether the second process is connected to the computer network***." As was discussed above with

respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "whether the second process is connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 1 patentable over the combination of NetBIOS and RFC 1531.

40. Claim 2 recites "each network protocol address stored in the memory ***following connection of a respective process to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

41. Claim 2 also recites "means... for determining the ***on-line status of the second process*** and for ***transmitting a network protocol address of the second process*** to the first process ***in response to a positive determination of the on-line status of the second process***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "an on-line status of the second process." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 2 and its dependent claim 3 are patentable over the combination of NetBIOS and RFC 1531.

42. Claim 4 recites "each of the network protocol addresses received following connection of the respective process to the computer network." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

43. Claim 4 also recites "receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an ***on-line status*** with respect to the computer network." Claim 4 further recites "determining the ***on-line status*** of the second process." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "an on-line status" of a process. An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 4 and its dependent claims 5-7 are patentable over the combination of NetBIOS and RFC 1531.

44. Claim 32 recites "the Internet Protocol address ***added to the list following connection of the process to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

45. Claim 32 also recites "an Internet accessible list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of ***a process currently connected to the Internet***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process currently [being] connected to the Internet." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the Internet. Thus, claim 32 is patentable over the combination of NetBIOS and RFC 1531.

46. Claim 33 recites "***the network protocol address*** of the corresponding process ***assigned to the process upon connection to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the

database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

47. Claim 33 also recites "selected ... entries comprising **a network protocol address** and a corresponding identifier ***of a process connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process [being] connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 33 and its dependent claims 34-37 are patentable over the combination of NetBIOS and RFC 1531.

48. Claim 38 recites "***the network protocol address*** of the corresponding process ***assigned to the process upon connection to the computer network***." As was discussed above with respect to the recitation of "the network protocol address of each respective process forwarded to the database following connection to the computer network" in claim 43, NetBIOS does not teach this dynamic address assignment. Further, it would not have been obvious to combine NetBIOS and RFC 1531 for the reasons set forth above with respect to claim 43.

49. Claim 38 also recites "selected ... entries comprising a network protocol address and a corresponding identifier of ***a process connected to the computer network***." As was discussed above with respect to the recitation of "processes having on-line status with respect to the computer network" in claim 43, NetBIOS does not teach that an active name in NetBIOS is synonymous with "a process [being] connected to the computer network." An active name simply refers to a name that has been registered and that has not yet been de-registered, independent of whether the associated computer is or is not connected to the computer network. Thus, claim 38 and its dependent claims 39-42 are patentable over the combination of NetBIOS and RFC 1531.

Claims 10-31

50. Claims 10-31 are rejected under 35 U.S.C. § 103(a) as being obvious over NetBIOS in view of Pinard, either alone or in combination with VocalChat User's Guide. In light of the cancelation of claim 10 and the amendment to claim 11, I will address this rejection with respect to claim 11.

51. Amended claim 11 recites "the caller process having a user interface and being operatively connectable to the callee process" and "establishing a point-to-point communication link ***from the caller process*** to the first callee process." When taken together, it is clear that the Office Action has not shown that the caller process, having a user interface, establishes a point-to-point communication link with a first callee process. The Exhibit M of the Request for Re-examination (the "Request") states that "Pinard discloses that a point-to-point communication link is established in response to a user associating an element representing ***the first callee process*** with the element representing a first communication line." However, as shown in Figure 1, Pinard discloses that conventional telephones connected to a telephony server via telephony interface circuits make calls to other conventional telephones. Those telephones are not caller and callee processes. In fact, with respect to the limitation "providing a user interface element representing a first callee process," Exhibit M identifies, instead of a callee process, two people, Mary and John. Clearly neither is a callee ***process***.

52. Exhibit M further states that "Applications which utilize NetBIOS application services ***inherently*** include 'user interfaces.'" This assertion is disputed. Page 356 of NetBIOS states "Protocols supporting NetBIOS services have been constructed on diverse protocol and ***hardware*** foundations." Thus, the NetBIOS applications could be running on a computer system (e.g., an embedded system) such that there are no user interfaces at all. In fact, the example provided in Exhibit M is disingenuous. Exhibit M states that "NetBIOS provides a vendor independent interface for the IBM Personal Computer (PC) and compatible systems." The "interface" discussed in that sentence is a software ***communications*** interface, between NetBIOS

service elements and NetBIOS-compatible applications, not a ***user*** interface. Moreover, the reference to "IBM Personal Computer (PC) and compatible systems" is a reference to the hardware, not the software with user interfaces that is running on the hardware. Exhibit M itself admits the deficiency in the inherency argument when it states "it is ***expected*** that on computers operating under the PC-DOS and MS-DOS operating systems that the existing NetBIOS interface will be preserved." This says nothing about what processes with user interfaces can participate in the claimed point-to-point communications.

53. With respect to the establishing step, Exhibit M does not even assert that one of the processes running on one of the DOS-style computers is the caller process. This is because neither the phone application not the device agent are part of the point-to-point communication -- the telephones are. Thus, this limitation is not met.

54. In addition, the Office Action has not established that one of ordinary skill in the art would have made the proposed combination. The Office Action alleges that "it would have been obvious … to utilize the user-interface elements and interaction taught by Pinard in the invention taught by NetBIOS since Pinard teaches that the invention can be used with any system in which a personal computer in conjunction with a server operates … since NetBIOS teaches that it can be implemented using different operating systems … and since examiner notes that both NetBIOS and Pinard relate to communications between at least two users implemented in a computerized environment." However, Pinard already discloses a personal computer 1 interacting with a server 5 without any indication that there is something missing from Pinard. Thus, without an indication that Pinard was somehow deficient, one of ordinary skill in the art would not have been motivated to include NetBIOS with Pinard. Accordingly, amended claim 11 and its dependent claims 12-20 are not rendered obvious by the combination of NetBIOS and Pinard.

55. Amended claim 22, like amended claim 11, recites "***the caller process having a user interface*** and being operatively connectable to the callee process" and "establishing a ***point-to-point communication link from the caller process*** to the first callee process." As was set forth

for the patentability of amended claim 11 above, that combination of elements is not taught by the applied combination of references. Thus, amended claim 22 and its dependent claims 23-31 are patentable over the applied combination of references.

The rejection of claims the pending claims over the Etherphone papers, either alone or in combination with at least one other reference

56. Claims 43 and 44 were rejected under 35 U.S.C. § 102 as being anticipated by Etherphone: Collected Papers 1987-1988 (May 1989) (hereinafter "the Etherphone papers"). The Etherphone Collected Papers include *An Overview of the Etherphone System and its Applications* (hereinafter "Zellweger"), *Telephone Management in the Etherphone System* (hereinafter "Swinehart"), and *Managing Stored Voice in the Etherphone System* (hereinafter "Terry").

57. Pending claims 1-7, 11-20 and 22-44 were rejected under 35 U.S.C. § 103(a) as being obvious over the combination of the Etherphone papers with at least one other reference (e.g., Harrick M. Vin, et al. *Multimedia Conferencing in the Etherphone* Environment, IEEE Computer Society (October 1991) (hereinafter "Vin") and RFC 1531).

The rejection of claims 43 and 44 as anticipated by the Etherphone papers

58. Claim 43 recites "a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network." Exhibit N of the Request asserts that this limitation is taught by the system directory database of local Xerox employees and cites Zellweger, page 4. However, in the paragraph just prior to the section cited by Exhibit N, Zellweger states "Figure 3 shows an Etherphone control window, called Finch, and a personal telephone directory window." The caption for Figure 3 further explains that "the

lower window [of Figure 3] shows a portion of a personal telephone directory, which is a set of speed-dialing buttons that can be created easily from an ordinary text file."

59. Figure 3 does not show that the cited database includes the claimed "network protocol address for a selected plurality of processes having on-line status with respect to the computer network" -- rather it simply appears to include phone numbers for the various entries, including phone numbers outside of the Etherphone system. Thus, this limitation is not taught by the Etherphone references, and I believe that the patentability of claim 43 should be confirmed.

60. The Office Action also has not shown that the Etherphone papers disclose "the network protocol address of each respective process forwarded to the database following connection to the computer network" as the Office Action has not shown how such a dynamic addressing technique is taught by the Etherphone papers. Thus, this limitation is not anticipated by the Etherphone papers as well.

61. As described above, claim 43 recites "a selected plurality of processes having on-line status with respect to the computer network." The Request and Exhibit N do not show how the Etherphone papers teach this recitation limitation. In fact, "on-line" status is not mentioned in any of the corresponding support sections for claim 43 in either the Request or Exhibit N. Further, the Etherphone papers do not teach an association between users' on-line status and the maintaining of a network address. To the contrary, the Etherphone system appears to always associate a user with a location, ***whether or not the user is currently at that location,*** and directs telephone calls for that user to that location ***without regard to the users' on-line status***. So, for example, while the Request points out that logging in to a workstation tells the Etherphone system—through unspecified mechanisms—where calls should be directed (citing Swinehart 1, page 2 as stating "Calls are to individuals, not locations...Logging in tells the telephone system where Karmen is."), the Request neglects to point out that ***logging out also tells the telephone system where calls should be directed***. ("Since Karmen has signed off from her workstation, the call rings immediately at an attendant's workstation..." Swinehart page 3).

62. Even assuming that the Office Action intended the rejection of claim 43 to be an obviousness rejection, claim 43 still is patentable over the Etherphone papers in view of Vin and RFC 1531. The Office Action alleges, with respect to claim 1, that it would have been obvious ... to utilize dynamically assigned IP address from Internet access servers as taught by Vin and RFC 1531 since Etherphone was intended for use in multiple networks and communication protocols (Terry, page 3), ... [and] since dynamic allocation of IP addresses allows for automatic reuse of an address that is no longer needed by the host to which it was assigned ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known in the art, and are useful to eliminate the burdensome task of manually assigning IP addresses for all networked components."

63. Terry, page 3, however, does not indicate under what circumstances the "multiple networks and communications protocols" would be used. The full paragraph that appears to be referenced by the Office Action states:

> The Etherphone system is intended for use in a locally distributed computing environment containing multiple workstations and programming environments, multiple networks and communication protocols, and perhaps even multiple telephone transmission and switching choices. The system is intended to be extensible in that introducing new applications, network services, workstations, networks, and other components is possible.

64. Thus, it is probable that the "multiple networks and communications protocols" refers to uses of the workstations and not the Etherphones. Similarly, Vin does not disclose how/where the IP address of Figure 5 is obtained. In addition, Figure 5 shows that the IP address is below a "datagram multicast" layer, so there is no evidence that one of ordinary skill in the art would have been motivated to use -- let alone know how to adapt -- the techniques of a multicast system when implementing a "point-to- point communication link between the first and second processing units through the Internet."

65. Furthermore, the Office Action has not shown that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with Vin and RFC 1531 without additional modifications to the combination -- modifications which are not specified in the Office Action. For example, the Office Action has not identified that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with RFC 1531 in light of the use of "leases" on network addresses within the RFC 1531 framework. As described in section 3.6 "If the lease expires before the host can contact a DHCP server, the host must immediately discontinue use of the previous network address and may inform local users of the problem." Similarly, section 4.4.4 of RFC 1531 states "If the lease expires before the client receives a DHCPACK, the client moves to INIT state, MUST immediately stop any other network processing and requests network initialization parameters as if the client were uninitialized. ... If the client is given a new network address, it MUST NOT continue using the previous network address and SHOULD notify the local users of the problem." However, if the system of the Etherphone papers and Vin is using "datagram multicast," how is the system supposed to deal with the loss of the address? Other processes that were communicating with the process that lost its lease would now begin sending the packets to the wrong destination. Moreover, by having to rely on another server that can go down such that a lease cannot be renewed and the computer has to stop using its network address, the combination is less robust than a system using static address assignment. Thus, the Office Action has not shown that one of ordinary skill in the art would have been motivated to combine the Etherphone papers with Vin and RFC 1531.

66. Thus, claim 43 is neither anticipated by the Etherphone papers nor rendered obvious by the Etherphone papers in combination with Vin and RFC 1531.

67. Claim 44 recites "following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network." As discussed above with respect to claim 43, the Office Action has not shown how such a dynamic addressing technique is taught by the Etherphone papers or rendered obvious by the combination of the Etherphone papers and Vin and RFC 1531.

68. Claim 44 also recites "querying the address server as to whether the second process is connected to the computer network." Exhibit N alleges that "conversations are established between two or more parties (Etherphones, servers, and so on) by performing remote procedure calls to the Voice Control Server." It then alleges that "when a first user at a first Etherphone (a first 'process') calls a second user at a second Etherphone (a second 'process'), the first Etherphone ***transmits a query*** in the form of a remote procedure call to determine the location of the second Etherphone." This allegation is made without citation to any portion of the Etherphone papers -- because the conclusion is not supported.

69. As admitted in the Exhibit N, Swinehart states "The telephone control server manages voice switching by sending to each Etherphone or service the network addresses of the other participants. ... Thereafter, voice datagrams are transmitted directly among the participants, bypassing the control server." Neither the Exhibit N nor Swinehart discloses when this information is sent, thus there is no evidence that a query is sent to an address server (e.g., such that the network address information is returned as part of the result of the query). In fact, if the Voice Control Server were periodically sending out information to Etherphones or other services, there would be no need to "querying the address server as to whether the second process is connected to the computer network." Thus, this limitation is not inherently met by the Etherphone references, and I believe that the patentability of claim 44 should be confirmed.

70. As described above, claim 44 recites "querying the address server ***as to whether the second process is connected to the computer network***." As was discussed above with respect to claim 43, the Request and Exhibit N do not show how the Etherphone papers teach this tracking of on-line status. Thus, claim 44 is patentable over the Etherphone papers.

<u>The rejection of claims 1-7 and 10-42 as obvious over the Etherphone papers in combination with at least one other reference</u>

71. Claims 1-7 and 10-42 have been rejected under 35 U.S.C. § 103(a) as obvious over the combination of the Etherphone papers in combination with (1) Vin and RFC 1531, (2) Pinard,

(3) Vin, RFC 1531 and NetBIOS, and (4) Pinard and VocalChat User's Guide. However, I believe that each of the pending claims in that group of claim is non-obvious as set forth below.

72. Claims 1, 2, 4-7 and 32-42 are rejected under 35 U.S.C. § 103(a) as being unpatentable over the Etherphone papers in view of Vin and RFC 1531. As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses.

73. Claim 1 also recites "program code for ***transmitting***, to the server, ***a query*** as to whether the second process is connected to the computer network." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining "whether the second process is connected to the computer network." Thus, claim 1 is not rendered obvious by the proposed combination of references.

74. Claim 2 recites "a memory, operatively coupled to the processor, for storing a network protocol address for selected of a plurality of processes, ***each network protocol address stored in the memory following connection of a respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses.

75. Claim 2 also recites "means, ***responsive to a query from the first process***, for determining the on-line status of the second process and for transmitting a network protocol address of the second process to the first process in response to a positive determination of the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach "determining the on-line status of the second

process." Thus, claim 2 and its dependent claim 3 are not rendered obvious by the proposed combination of references.

76. Claim 4 recites "A. receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an on-line status with respect to the computer network, ***each of the network protocol addresses received following connection of the respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses.

77. Claim 4 also recites "B. ***receiving a query*** from the first process to determine the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the Etherphone papers. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach "determine[ing] the on-line status of the second process." Thus, claim 4 and its dependent claim 5-7 are not rendered obvious by the proposed combination of references.

78. Claim 32 recites "a. maintaining an Internet accessible list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of a process currently connected to the Internet, ***the Internet Protocol address added to the list following connection of the process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***currently connected*** to the Internet." Thus, claim 32 is patentable over the applied combination of references.

79. Claim 33 recites "maintaining, in a computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, the network protocol address of the corresponding process assigned to the process upon connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***connected to*** the computer network." Thus, claim 33 and its dependent claims 34-37 are patentable over the applied combination of references.

80. Claim 38 recites "program code configured to maintain, in the computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, ***the network protocol address of the corresponding process assigned to the process upon connection to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Further, as was discussed with respect to claim 43, the Office Action has not shown that the Etherphone papers teach determining the on-line status of processes such that the Etherphone papers would disclose entries for processes that are "***connected to*** the computer network." Thus, claim 38 and its dependent claims 39-42 are patentable over the applied combination of references.

81. Claims 10-30 are rejected under 35 U.S.C. § 103(a) as being unpatentable over the Etherphone papers in view of Pinard, either alone or in combination with the VocalChat User's Guide. Amended claim 11 recites "the caller process having a user interface and being operatively connectable to the callee process" and "establishing a point-to-point communication

link ***from the caller process*** to the first callee process." When taken together, it is clear that the Request (including Exhibit N) and Office Action have not shown that the caller process, having a user interface, establishes a point-to-point communication link with a first callee process. The Request states on page 119 that "Etherphone describes establishing a point-to-point communication link between a caller process and a callee process. ... Pinard discloses that a point-to-point communication link is established in response to a user associating an element representing the first callee process with the element representing a first communication line." Neither of those assertions actually state that any of the user interface elements described in the Request establish a point-to-point communication with a callee process. In fact, Exhibit N admits that the user interface elements are on the workstation, not on the Etherphones that conduct the voice communications. Page 12 of Exhibit N directly states that the "GUI features [are] presented on the workstation display."

82. In addition, the Office Action has not established that one of ordinary skill in the art would have made the proposed combination. The Office Action alleges that "it would have been obvious ... to utilize[e] the user-interface elements and interactions taught by Pinard in the invention taught by Etherphone since Pinard teaches that the invention can be used with any system in which a person computer in conjunction with a server operate." However, Pinard already discloses a personal computer 1 interacting with a server 5 without any indication that there is something missing from Pinard. Thus, without an indication that Pinard was somehow deficient, one of ordinary skill in the art would not have been motivated to include Etherphone with Pinard. Accordingly, amended claim 11 and its dependent claims 12-20 are not rendered obvious by the combination of NetBIOS and Pinard.

83. Amended Claim 22, like amended claim 11, recites "***the caller process having a user interface*** and being operatively connectable to the callee process" and "establishing a ***point-to-point communication link from the caller process*** to the first callee process." As was set forth for the patentability of claim 11 above, that combination of elements is not taught by the applied

combination of references. Thus, amended claim 22 and its dependent claims 23-31 are patentable over the applied combination of references.

The rejection of the pending claims over the VocalChat References, either alone or in combination with at least one other reference

84. Claims 43 and 44 were rejected under 35 U.S.C. § 103(a) as being obvious over VocalChat User's Guide in view of VocalChat Readme, VocalChat Networking, VocalChat Help File and VocalChat Troubleshooting Help file (collectively the "VocalChat References"), either alone or in combination with at least one other reference.

Claims 43 and 44

85. Claim 43 recites "program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network." The Office Action alleges that this limitation is met because "the network protocol address of clients are transmitted to and stored in a Connection List / USERS file located on the network." However, each VocalChat process actually reads the file and locally processes it, writing back the whole contents of the file when it has added its changes to it. Thus, in the context of multiple, independently-operating VocalChat clients each being able to read from and write to the "shared CONNLIST.VC" file, that file is not acting as a directory database, just a file whose contents could readily become inconsistent with the actual state of processes. For example, if a first process reads the file and then a second process reads the file, then the first process writes the file and then the second process writes the file, the changes written by the first process will be lost when the second process writes back its changes. Similarly, if the second process wrote back the file before the first, then the second process's changes would be lost when the first process wrote back its version of the file. Thus, the VocalChat References do not anticipate the claimed "program code configured to access a

directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network."

86. In addition, claim 43 also recites "the network protocol address of each respective process forwarded to the database following connection to the computer network." This limitation is not taught by the VocalChat References, and the Office Action does not identify how this dynamic addressing is taught by the VocalChat References. Even assuming that the Office Action had intended to apply the combination of the VocalChat references and RFC 1531, the combination of references still would not have rendered claim 43 obvious.

87. With respect to claim 1, the Office Action asserts that "it would have been obvious to utilize the limitations taught by RFC 1531 in the invention taught by VocalChat ... since this allows for automatic reuse of an address that is no longer needed by the host to which it was assigned ... and since examiner notes the use of dynamic IP address assignment in a TCP/IP network are old and well known ... and are useful to eliminate the burdensome task of manually assigning IP addresses for all network computers." I do not agree with the conclusion drawn in by the Office Action on the combinability of the references.

88. In the context of point-to-point communication, widespread use of dynamically assigned addresses is not the solution to a problem, it is the problem itself. Dynamically assigned addresses were known, and the patent in re-examination specifically states in that regard, "Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain." Col. 1, lines 53-56.

89. The development history of the VocalChat products indicates that the developers of the VocalChat products had problems addressing dynamic addressing issues. The Request cites a Generic version of the VocalChat client which, according to Mr. Cohen, was used on local area networks. See Cohen Declaration, paragraph 3. There apparently was a subsequent version of VocalChat that was also released by VocalTec to the public in 1994, at least in beta. This version, called VocalChat Gateway To Interent (or "VocalChat GTI") was designed for use on

the Internet, and I have been informed that Net2Phone believes that VocalChat GTI used static local address files into which static callee addresses were manually input. I have also been informed that Net2Phone believes that VocalChat GTI did not utilize a server at all.

90. Based on the above, I believe the use of manual inputting of static addresses and the absence of a server suggests that the VocalTec designers—presumably software developers of at least ordinary skill in the art—did not consider the alleged combination of their own VocalChat references with RFC 1531, or it suggests that they did consider it but were unable to overcome the non-trivial obstacles to doing so.

91. I have also been informed that Net2Phone believes that soon after the release of the VocalChat GTI version, VocalTec released another VocalChat version that used Internet Relay Chat (IRC) to help VocalChat clients with dynamically assigned IP addresses find one another. This change from VocalChat GTI to VocalChat IRC appears to be further objective evidence that even the VocalChat designers recognized that the "improvement" to the Generic VocalChat implementation was still deficient. If the designers of the VocalChat Generic implementation did not see fit to combine dynamic addressing with the Generic implementation disclosed in the VocalChat references, then I do not believe that one of ordinary skill in the art would not have done so either. Thus, claim 43 is not anticipated by or rendered obvious by the VocalChat References.

92. Claim 44 recites "querying the address server as to whether the second process is connected to the computer network." The Office Action cites the Help file, pages 2 and 26, and the Network Information file, page 10, as disclosing this element and states "clients query the network server and … the server stores addresses of logged in users." However, the Office Action has not identified how the cited portions of those references teach the claimed "querying." At best, the references teach that a local process reads a "USERS" file or a Connections file. As can be seen from page 4 of the VocalChat Network Information (reproduced below), when the VocalChat system uses the Generic mode, a USERS file is used.

2.5. Network parameters in the VocalChat INI files

These are the network parameters in the VocalChat VOCLCHAT.INI and VCSETUP.INI files (under the `NetWork` section):

```
[NetWork]
NetWork=             / description of the selected network
NetWorkUsers=        / user services: NetWare/WinWorkgroups/Generic*
NetWorkProtocol=     / network protocol: IPX/NetBIOS
NetWorkType=         / name of the selected network, for future use
UsersFile=           / path name of users file (when Generic is set)
MyUserName=          / the name of the user (when Generic is set)**
```

* When `Generic` is set, a USERS file is used.

** This line appears only in the VOCALCHAT.INI file of each user.

The VOCLCHAT.INI files are in the windows directory of each user. The VCSETUP.INI file is in the VOCLCHAT directory, where VocalChat was installed, and is used only to supply default values for the different installations.

93. The USERS file configuration parameter includes a "UsersFile" entry that specifies the "path name of users file (when Generic is set)." However, it is also stated that "The VOCLCHAT.INI files are in the windows directory of each user." Thus, this "UsersFile" entry is a local configuration parameter such that the local VocalChat client reads and writes the USERS file on its own -- without performing the claimed query.

94. Similarly, page 8 of the VocalChat Help file states "If your network type is not NetWare or Windows for Workgroups, the Setup program creates a Connection List file which is used to identify and access users." The Connection List file and the USERS file apparently have the same function. Thus, the identification and access enabled by the Connection List is performed by the local VocalChat client reading and writing the file itself -- without performing the claimed query. Accordingly, claim 44 is not rendered obvious by the applied combination of references.

95. Claim 44 further recites "following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is

connected to the computer network." As discussed above with respect to the "querying" step, in the VocalChat Generic implementation, a local process reads a "USERS" file or a Connections file in its entirety and writes it back in its entirety rather than "forwarding to the address server a network protocol address at which the first process is connected to the computer network." This causes the VocalChat system to have to send an increasing amount of information as the number of users increases. Sending the whole file such that the new file replaces the old file also creates problems with consistency such that one user's changes could overwrite the changes of another user -- especially as networks got larger which would have increased the problem of inconsistent files being written.

96. In addition, with respect to "following connection of the first process to the computer network forwarding … a network protocol address at which the first process is connected to the computer network," the Office Action has not shown how this dynamic address-based limitation is taught by the VocalChat References. Thus, this limitation has not been shown to be taught by the applied references.

97. Further, even if the Office Action had proposed a combination of the VocalChat References and RFC 1531, the combination would still not render obvious claim 44. As was discussed above with respect to claim 43, there is no motivation for combining VocalChat References and RFC 1531. Thus, claim 44 is not rendered obvious by the VocalChat References, either alone or in combination with RFC 1531.

<u>Claims 1-7 and 31-42</u>

98. Claim 1 recites "program code for transmitting, to the server, a query as to whether the second process is connected to the computer network." As was discussed above with respect to claim 44, the VocalChat References do not teach "querying." In addition, the Office Action has not shown how the VocalChat references teach such a server capable of receiving a query. Thus, this limitation is not taught by the proposed combination.

99. Claim 1 also recites "program code for transmitting to the server a network protocol address received by the first process ***following connection to the computer network***." As was discussed above with respect to claim 43, it would not have been obvious to combine the VocalChat References and RFC 1531. Thus, claim 1 is nonobvious in view of the applied references.

100. Claim 2 recites "means, ***responsive to a query from the first process***, for determining the on-line status of the second process and for transmitting a network protocol address of the second process to the first process in response to a positive determination of the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the VocalChat References.

101. Claim 2 further recites "each network protocol address stored in the memory ***following connection of a respective process to the computer network***." As was discussed above with respect to claim 43, the VocalChat References do not teach this dynamic address assignment. Further, it would not have been obvious to combine the VocalChat References and RFC 1531 for the reasons set forth above with respect to claim 43. Thus, claim 2 and its dependent claim 3 are patentable over the combination of the VocalChat References and RFC 1531.

102. Claim 4 recites "A. receiving and storing into a computer memory a respective network protocol address for selected of a plurality of processes that have an on-line status with respect to the computer network, ***each of the network protocol addresses received following connection of the respective process to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references.

103. Claim 4 also recites "B. ***receiving a query*** from the first process to determine the on-line status of the second process." As was discussed above with respect to claim 44, the Office Action has not provided any evidence to support the allegation that such a query is taught by the

VocalChat references. Thus, claim 4 and its dependent claim 5-7 are not rendered obvious by the proposed combination of references.

104. Claim 32 recites "a. maintaining an ***Internet accessible*** list having a plurality of selected entries, each entry comprising an identifier and a corresponding Internet protocol address of a process currently connected to the Internet." Exhibit O of the Request admits that "While VocalChat does not explicitly describe a server with stored names and addresses is accessible over 'the Internet,' it describes the use of TCP/IP, which is the protocol used on the Internet." Exhibit O then states that "VocalChat inherently describes that the list of users and network addresses is accessible over the Internet"; however, this is not inherently true. There may be any range of local area networks running TCP/IP communications that are not connected to the Internet. Thus, just because the Internet is a type of TCP/IP network does not mean that all TCP/IP networks must be the Internet.

105. In fact, the security and privacy concerns that arise in a secure, local area network are very different than the issues that arise on the Internet. The VocalChat References describe that "All users must use the same Post-Office, otherwise they won't be able to communicate or leave messages to each other. This means that all users must ... have write permission for the Post-Office directory." This would mean opening up one's network to outsiders across the Internet. The Office Action has not cited any evidence that one of ordinary skill in the art would have been motivated to do so in light of the inherent risks. Further, the VocalChat development history discussed above is illuminating about how one of ordinary skill in the art would have modified the VocalChat system. The VocalChat designers went from a Generic implementation with a shared directory with a shared file that can be remotely updated, to locally processed files that were manually updated in the VocalChat GTI version. Thus, the trend on how to modify the VocalChat system was the opposite of what is being proposed.

106. Claim 32 also recites "***the Internet Protocol address added to the list following connection of the process to the computer network.***" As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been

motivated to make the proposed combination of references. Thus, claim 32 is patentable over the applied combination of references.

107. Claim 33 recites "maintaining, in a computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, the network protocol address of the corresponding process assigned to the process upon connection to the computer network." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Thus, claim 33 and its dependent claims 34-37 are patentable over the applied combination of references.

108. Claim 38 recites "program code configured to maintain, in the computer memory, a network accessible compilation of entries, selected of the entries comprising a network protocol address and a corresponding identifier of a process connected to the computer network, ***the network protocol address of the corresponding process assigned to the process upon connection to the computer network***." As was discussed above with respect to claim 43, the Office Action has not shown that one of ordinary skill in the art would have been motivated to make the proposed combination of references in light of the problems associated with the use of dynamic addresses. Thus, claim 38 and its dependent claims 39-42 are patentable over the applied combination of references.

109. Amended claim 11 recites "c.1 querying the server as to the on-line status of the first callee process; and c.2 receiving a network protocol address of the first callee process over the computer network from the server." As was discussed above with respect to claim 44, the VocalChat References do not disclose such a querying step. Thus, amended claim 11 and its dependent claims 12-20 are patentable over the applied combination of references.

110. Amended claim 22 recites "program code for querying the server as to the on-line status of the first callee process; and program code for receiving a network protocol address of the first callee process over the computer network from the server." As was discussed above with respect

to claim 44, the VocalChat References do not disclose such a querying step. Thus, amended claim 22 and its dependent claims 23-30 are patentable over the applied combination of references.

111. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Dated: November 27, 2009

Ketan Mayer-Patel, Ph.D.

EXHIBIT 1 TO MAYER-PATEL DECLARATION

Ketan Mayer-Patel

154 Fred Brooks Building
Department of Computer Science, CB #3175
University of North Carolina, Chapel Hill

kmp@cs.unc.edu
http://www.cs.unc.edu/~kmp

Education

Ph.D. University of California, Berkeley, 1999
Parallel Software-only Video Effects Processing

M.S. University of California, Berkeley, 1997
Design and Performance of the Berkeley Continuous Media Toolkit

B.A. University of California, Berkeley, 1992
Majors: Computer Science and Economics

Professional Experience

Associate Professor
University of North Carolina, Chapel Hill, NC (August 2005 – present)

Assistant Professor
University of North Carolina, Chapel Hill, NC. (January 2000 – August 2005)

Visiting Researcher
Microsoft Bay Area Research Center (BARC), San Francisco, CA. (June 2003 – December 2003)

Graduate Student Researcher
University of California, Berkeley, CA. (June 1993 – November 1999)

Graduate Student Instructor
University of California, Berkeley, CA. (August 1997 – December 1997)

Programmer
University of California, Berkeley, CA. (June 1992 – June 1993)

Programmer
United States Department of Agriculture, Albany, CA. (May 1991 – June 1992)

Honors and Notables

- National Science Foundation CAREER Award, 2003
- Computer Science Student Association Teaching Award, 2003
- Invited to three major meetings (one domestic and two international) of top multimedia researchers to discuss future directions for the field.
- In the sixteen-year history of the ACM SIGMultimedia Conference, considered to be the premier conference in the field of multimedia, I have published twelve papers in ten different years.

Publications

Refereed Journals

K. Mayer-Patel and D. Gotz, "Scalable, Adaptive Streaming for Nonlinear Media," *IEEE Multimedia*, vol. 14, no. 3 (15 pages).

D. Ott and K. Mayer-Patel, "An open architecture for transport-level protocol coordination for distributed multimedia applications," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 3, no. 3 (22 pages).

D. Gotz and K. Mayer-Patel, "GAL: A middleware library for multidimensional adaptation," under review for *ACM Transactions on Multimedia Computing, Communications, and Applications* (21 pages).

K. Mayer-Patel, B. Smith, and L.A. Rowe, "The Berkeley software MPEG-1 video decoder," to appear in *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 1, no. 1 (23 pages).

K. Mayer-Patel and S.-U. Kum, "Real-time multi depth stream compression," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 1, no. 2 (26 pages).

D. Gotz and K. Mayer-Patel, "A Framework for Scalable Delivery of Digitized Spaces," *International Journal on Digital Libraries*, vol. 5, no. 3 (14 pages).

J. Considine, K. Mayer-Patel, and J. Byers, "A case for testbed embedding services," *Computer Communication Review*, vol. 34, no. 1, January 2004, pp. 137-142.

Refereed Conferences and Workshops

K. Mayer-Patel, "Systems challenges of media collectives: Supporting media collectives with adaptive MDC," *Proceedings of the 15th International ACM Conference on Multimedia*, Augsberg, Germany, 2007, pp. 625-630.

S. Krishnan and K. Mayer-Patel, "A utility-driven framework for loss and encoding aware video adaptation," *Proceedings of the 15th International ACM Conference on Multimedia*, Augsberg, Germany, 2007, pp. 1026-1035.

D. Gotz and K. Mayer-Patel, "A general framework for multidimensional adaptation," *Proceedings of the 12th International ACM Conference on Multimedia*, New York, 2004, pp 612-619.

D. Ott and K. Mayer-Patel, "Coordinated multi-streaming for 3D tele-immersion," *Proceedings of the 12th International ACM Conference on Multimedia*, New York, NY, 2004, pp. 596-603.

D. Ott and K. Mayer-Patel, "Aggregate congestion control for distributed multimedia applications," *Proceedings of IEEE Infocom '04*, Hong Kong, 7-11 March 2004, vol. 1, pp. 13-23.

K. Mayer-Patel and W. Miaw, "Evaluating the effectiveness of automatic PVR management," *Proceedings of the SPIE Conference on Storage and Retrieval Methods and Applications for Multimedia*, San Jose, CA, January 2004, vol. 5307, pp. 360-365.

S.-U. Kum, K. Mayer-Patel and H. Fuchs, "Real-time compression for dynamic 3D environments," *Proceedings of the 11th International ACM Conference on Multimedia*, Berkeley, CA, 2003, pp. 185-194.

N. Kelshikar, X. Zabulis, J. Mulligan, K. Daniilidis, V. Sawant, S. Sinha, T. Sparks, S. Larsen, H. Towles, K. Mayer-Patel, H. Fuchs, J. Urbanic, K. Benninger, R. Reddy and G. Huntoon, "Real-time terascale implementation of tele-immersion," *Proceedings of the International Conference on Computation Science*, Melbourne, Australia, 2003, Springer-Verlag Lecture Notes in Computer Science vol. 2660, pp. 33-42.

K. Mayer-Patel, L. Le and G. Carle, "An MPEG performance model and its application to adaptive forward error correction," *Proceedings of the 10th International ACM Conference on Multimedia*, Juan-les-Prins, France, 2002, pp. 1-10.

D. Gotz and K. Mayer-Patel, "IRW: an incremental representation for image-based walkthroughs," *Proceedings of the 10th International ACM Conference on Multimedia*, Juan-les-Prins, France, 2002, pp. 67-76.

D. Ott and K. Mayer-Patel, "A mechanism for TCP-friendly transport-level protocol coordination," *Proceedings of the USENIX Technical Conference*, Monterrey, CA, 2002 (14 pages).

A. Wilson, K. Mayer-Patel and D. Manocha, "Spatially-encoded far-field representations for interactive walkthroughs," *Proceedings of the 9th International ACM Conference on Multimedia*, Ottawa, Canada, 2001, pp. 348-357.

D. Ott and K. Mayer-Patel, "Transport-level protocol coordination in cluster-to-cluster applications," *Proceedings of the 8th International Workshop on Interactive Distributed Multimedia Systems (Lecture Notes in Computer Science)*, vol. 2158, Springer, 2001, pp. 10-22.

D. Yu, D. Wu, K. Mayer-Patel and L.A. Rowe, "dc: a live webcast control system," *Proceedings of the SPIE Conference on Multimedia Computing and Networking*, vol. 4312, San Jose, CA, 2001, pp. 111-122.

K. Mayer-Patel, "Incorporating application-level knowledge into the MPEG-2 coding model," *Proceedings of the Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Chapel Hill, CA, 2000, (6 pages).

K. Mayer-Patel and L.A. Rowe, "Exploiting spatial parallelism for software-only video effects processing," *Proceedings of the SPIE Conference on Multimedia Computing and Networking*, vol. 3654, San Jose, CA, 1999, pp. 252-263.

K. Mayer-Patel and L.A. Rowe, "A multicast control scheme for parallel software-only video effects processing," *Proceedings of the 7th International ACM Conference on Multimedia*, Orlando, FL, 1999, pp. 409-418.

K. Mayer-Patel and L.A. Rowe, "Exploiting temporal parallelism for software-only video effects processing," *Proceedings of the 6th International ACM Conference on Multimedia*, Bristol, England, 1998, pp. 161-169.

T.H. Wong, K. Mayer-Patel and L.A. Rowe, "A software-only video production switcher for the Internet MBone," *Proceedings of the SPIE conference on Multimedia Computing and Networking*, vol. 3310, San Jose, CA, 1998, pp. 28-41.

K. Mayer-Patel and L.A. Rowe, "Design and performance of the Berkeley Continuous Media Toolkit," *Proceedings of the SPIE conference on Multimedia Computing and Networking*, vol. 3020, San Jose, CA, 1997, pp. 194-206.

K. Mayer-Patel, D. Simpson, D. Wu, and L.A. Rowe, "Synchronized continuous media playback through the World Wide Web," *Proceedings of the 4th International ACM Conference on Multimedia*, Boston, MA, 1997, pp. 435-436.

L.A. Rowe, K. Patel, and B. Smith, "MPEG video in software: representation, transmission, and playback," *Proceedings of the SPIE conference on High-Speed Networking and Multimedia Computing*, vol. 2188, San Jose, CA, 1994, pp. 134-144.

K. Patel, B. Smith, and L.A. Rowe, "Performance of a software MPEG video decoder," *Proceedings of the 1st International ACM Conference on Multimedia*, Los Angeles, CA, 1993, pp. 75-82.

Software Artifacts

mpeg_play

The first publicly available MPEG-1 video decoder originally released in 1993. Over 1,000,000 copies of this program have been downloaded. It has been used as a code base for innumerable research and open source systems. Mayer-Patel was the architect of the original code that was later refactored and maintained by a number of other individuals.

The Berkeley Continuous Media Toolkit

The Berkeley CMT provided a framework within which to develop experimental multimedia tools and applications. Although primarily used by researchers at UC Berkeley, it was employed by a number of different research groups world-wide. Development of CMT ended in approximately 1998.

MPEG2Event

This recently released C# library allows researchers to rapidly develop MPEG-2 analysis tools that are interested in the details of bit-level coding elements. Although currently in use by only a small number of researchers, it is freely available at http://www.cs.unc.edu/~kmp/mpeg2event. Further development of the library is on-going.

Teaching

COMP 416: Introduction to Web Programming
My goal with this course is to pique student interest for more detailed upper-division courses in operating systems, networking, databases, security, etc. while satisfying their practical interest in developing web programming skills.

COMP 426: Advanced Web Programming
A follow-on course to COMP 416, this course expands on client-server programming concepts and concentrates more attention to the design and use of databases and XML-related technologies.

COMP 249: Multimedia Computing and Networking
This course is an advanced graduate-level course that covers the fundamental concepts in multimedia computing and networking. Students are expected to complete an extensive final project, some of which have led to publications in refereed conferences and workshops.

COMP 249-080: Topics in Multimedia Systems
This seminar course provides students with an opportunity to read and present the most research literature in multimedia systems.

Research Areas

Coordinated Multistreaming
In this project, we are developing mechanisms to address the needs of distributed multimedia applications that employ many (i.e., 10's or 100's) of different media flows with complex inter-stream semantics and adaptation requirements. This project addresses fundamental problems in protocol coordination and aggregate congestion control.

Multidimensional Adaptation
We are developing a framework for compactly expressing and evaluating adaptation policies that must negotiate tradeoffs in real-time within very large multiresolutional datasets with high dimensionality.

StrandCast
StrandCast is an application-layer multicast protocol intended for latency-insensitive multimedia applications such as receiver-driven layered multicast and pyramid broadcasting. The design and implementation of StrandCast exploits the lax latency requirements of these applications to optimize for link stress, rapid joins and leaves, and robustness in the face of node failure.

Encoding and Transmission of 3D Scenes from Multiple Cameras
The project explores ways to efficiently transmit video data from a set of cameras viewing the same scene. This problem is at the heart of most tele-immersion applications. Our hypothesis is that it is possible to exploit depth information (even if imperfect) derived from stereo correlation between cameras to more efficiently encode the original color information.

Recoverable Video Adaptation
Existing video adaptation techniques generally lead to irreversibly loss of video quality. In this project, we are exploring adaptation techniques that can be used to recover high (or at least higher) quality video from a set of independently constructed lower quality representations.

Funding

CAREER: Enabling Futuristic Distributed Applications with Integrative Multistream Networking
PI's: K. Mayer-Patel
Agency: National Science Foundation (ANI-0238260)
Amount: $404,387
Duration: 8/15/2003 – 8/14/2008

ITR: Protocol Coordination for Multi-Stream Applications
PI's: K. Mayer-Patel
Agency: National Science Foundation (ANI-0219780)
Amount: $368, 047
Duration: 10/1/2002 – 9/30/2005

RI: Tera-Pixels - Using High-Resolution Pervasive Displays to Transform Collaboration and Teaching
PI's: K. Jeffay, A. Lastra, F.D. Smith, K. Mayer-Patel and L. McMillan
Agency: National Science Foundation (EIA-0303590)
Amount: $590,986
Duration: 8/15/2003 – 8/14/2008

3D Telepresence for Medical Consultation: Extending Medical Expertise Throughout, Between, and Beyond Hospitals
PI's: H. Fuchs, B. Cairns, K. Mayer-Patel, D. Sonnenwald, G. Welch
Agency: National Library of Medicine
Amount: $2,549,980
Duration: 09/30/2003-09/29/2006

Video-Based Representation and Rendering of Large Real and Synthetic Environments
PI's: D. Manocha and K. Mayer-Patel
Agency: Office of Naval Research
Amount: $112,384
Duration: 01/01/2001-12/31/2003

Video Quality Metric Oracle

PI's: K. Mayer-Patel
Agency: North Carolina Networking Initiative Fellowship Program
Amount: $20,000
Duration: 08/15/2001 – 5/15/2002

SCOUT: An On-Line Network Path Measurement and Characterization Tool

PI's: K. Mayer-Patel
Agency: North Carolina Networking Initiative Fellowship Program
Amount: $20,000
Duration: 08/15/2000 – 5/15/2001

Professional Activities

Editorships

- Associate Editor, ACM Transactions on Multimedia Communications, Computing, and Applications (TOMCCAP)
- Associate Editor, IEEE Multimedia Magazine

Executive Committees

- Co-Chair, International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)

Organizing Committees

- Program Chair, ACM Multimedia Systems 2010
- General Co-Chair, Multimedia Networking and Computing 2009
- Program Co-Chair, Multimedia Modeling (MMM) 2009
- General Co-Chair, NOSSDAV 2005
- Program Co-Chair, ACM Multimedia, 2006
- Open Source Software Competition Chair, ACM Multimedia (2004, 2005)
- Tutorial Program Chair, ACM Multimedia (2003)
- Doctoral Symposium Chair, ACM Multimedia (2000, 2001)

Program Committees

- ACM Multimedia
- NOSSDAV
- Multimedia Computing and Networking (MMCN)
- Multimedia Interactive Protocols and Systems Workshop
- IFIP Networking Conference
- Multimedia Information Systems Conference
- International World Wide Web Conference
- SPIE Conference on Multimedia Computing and Networking
- IEEE International Conference on Distributed Computing Systems
- Interactive Distributed Multimedia Systems Workshop
- Global Internet Symposium

Other Professional Service

- Guest Editor, Special Issue of Multimedia Systems Journal featuring expanded papers from the SPIE Conference on Multimedia Computing and Networking, 2003.
- In 2004, participated in a by invitation-only meeting of leaders within ACM SIGMultimedia. A report of the meeting outlining important directions for multimedia research will appear in Transactions on Multimedia Computing, Communications, and Applications.
- Invited to an international meeting of leading multimedia researchers being organized for Spring 2005 in Dagstuhl, Germany to discuss the future of multimedia research.

Past Ph.D. Students

- David Gotz, *Supporting adaptive scalable access to multiresolutional multidimensional data,* May 2005.
- David Ott, *Coordination mechanisms for distributed multistream applications*, November 2005.
- Sang-Uok Kum, *Encoding and transmission of 3D depth streams*, November 2008.

University Service

University Committees

- Tar Heel Bus Tour Advisory Committee (Fall 2001).

Department Service

- Chair of Undergraduate Curriculum Committee (Fall 2009 – present).
- Chair of Graduate Admissions Committee (Spring 2005 – Fall 2009).
- Member of Graduate Admissions Committee (Spring 2001 – Spring 2005).

Other Service

- Project UPLIFT participant (recruitment of minority high school students)
- Co-coach of the UNC ACM Programming Competition team (Fall 2000 – present).

EXHIBIT 2 TO MAYER-PATEL DECLARATION

Web Building
Web HOME
Web Building
Web Design
Web Users
Web Standards
Web Validation
Web W3C
Web Security
Web Glossary
Web Search

Web Building Security

« Previous   Next Chapter »

You are offering your IP address to the entire world at this very moment.

Make sure you are not offering access to your private data at the same time.

YOUR IP ADDRESS IS PUBLIC

Accessing the Internet is a security risk.

When you are connected to the Internet, an IP address is used to identify your computer. If you don't protect yourself, this IP address can be used to access your computer from the outside world.

A fixed IP address is a larger security risk.

If you're using a modem with a dial-up connection, you will get a new IP address every time you connect to Internet.

With an ADSL or cable connection users sometimes keep the same IP address for several months, this represents an increased security risk.

If you have a fixed IP address, you give Internet hackers all the time they need to search for entrances on your computer, and to store and share (with other hackers) information they find on your computer.

Your Network Shares

Personal computers are often connected to a shared network. Personal computers in large companies are connected to large corporate networks. Personal computers in small companies are connected to a small local network, and computers in private homes often share a network between family members.

Most often networks are used to share resources like printers, files and disk storage.

When you are connected to the Internet, your shared resources can be accessed by the rest of the world.

A Common Windows Security Problem

Unfortunately, many Microsoft Windows users are unaware of a common security leak in their network settings.

This is a common setup for network computers in Microsoft Windows:

- Client for Microsoft Networks
- File and Printer Sharing for Microsoft Networks
- NetBEUI Protocol
- Internet Protocol TCP/IP

If your setup allows NetBIOS over TCP/IP, you have a security problem:

- Your files can be shared all over the Internet
- Your logon-name, computer-name, and workgroup-name are visible to others

If your setup allows File and Printer Sharing over TCP/IP, you also have a problem:

- Your files can be shared all over the Internet

Computers that are not connected to any network can also have unsecure network settings, because the settings were changed when Internet was installed.

Solving the Problem

For Windows 2000 users:

You can solve your security problem by **disabling NetBIOS over TCP/IP**:

- Open Windows Explorer
- Right-click on My Network Places
- Select: Properties
- Right-click on Local Area Network
- Select: Properties

- Select: Internet Protocol TCP/IP
- Click on Properties
- Click on Advanced
- Select the WINS tab
- Select Disable NetBIOS over TCP/IP
- Click OK

If you get the message: "This connection has an empty......", ignore the message and click on YES to continue, and click OK to close the other setup windows.

You should restart your computer after the changes.

For Windows 95, 98, or ME users:

You can solve your security problem by **disabling NetBIOS over TCP/IP**:

- Open Windows Explorer
- Right-click on My Network Places
- Select: Properties
- Select: Internet Protocol TCP/IP
- Click on Properties
- Select the NetBIOS tab
- Uncheck: Enable NetBIOS over TCP/IP
- Click OK

You must also **disable the TCP/IP Bindings** to Client for Microsoft Networks and File and Printer Sharing:

- Open Windows Explorer
- Right-click on My Network Places
- Select: Properties
- Select: Internet Protocol TCP/IP
- Click on Properties
- Select the Bindings tab
- Uncheck: Client for Microsoft Networks
- Uncheck: File and Printer Sharing
- Click OK

If you get a message with something like: "You must select a driver........", ignore the message and click on YES to continue, and click OK to close the other setup windows.

If you still want to **share your Files and Printer** over the network, you must use the NetBEUI protocol instead of the TCP/IP protocol. Make sure you have enabled it for your local network:

- Open Windows Explorer
- Right-click on My Network Places
- Select: Properties
- Select: NetBEUI
- Click on Properties
- Select the Bindings tab
- Check: Client for Microsoft Networks
- Check: File and Printer Sharing
- Click OK

You should restart your computer after the changes.

Protect Your Server

« Previous Next Chapter »

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

MAILED

DEC 03 2009

CENTRAL REEXAMINATION UNIT

Michael R Casey DAVIDSON BERQUIST JACKSON & GOWDEY LLP 4300 WILSON BLVD., 7TH FLOOR ARLINGTON VA 22203	:	(For Patent Owner)
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP 1279 OAKMEAD PARKWAY SUNNYVALE, CA 94085-4040	:	(For Third Party Requester)
In re: Hutton *et alia* *Ex Parte* Reexamination Proceeding Control No. 90/010,416 Deposited: 17 February 2009 For: US Patent No. 6,108,704	:	DECISION DISMISSING PETITION FOR EXTENSION OF TIME 37 CFR § 1.550(c) & 1.181

This is a decision on the 23 November 2009, "Supplemental Request for Extension of Time in a Re-Examination" filed under 37 CFR § 1.550(c) requesting that the time for responding to the non-final Office action dated 27 August 2009, be further extended by one (1) week. The petition was filed with the required certificate of service and petition fee. The petition was timely filed.

The petition is before the Director of the Central Reexamination Unit for consideration.

The petition is dismissed for the reasons set forth below.

DISCUSSION

The Patent Owner requests the period of time be further extended in which to file a response to non-final Office action dated 27 August 2009 which set a two (2) month period for response thereto. A first petition for extension of time was granted extending the response period by one month in the decision dated 26 October 2009. The subsequent petition was timely filed with the required certificate of service and petition fee pursuant to 37 CFR §§ 1.550(c) and 1.17(g). Second requests for extension of time will only be granted in exception circumstances.

The petition for extension of time dated 23 November 2009 is dismissed.

37 CFR § 1.550 (c) states:

> (c) The time for taking any action by a patent owner in an *ex parte* reexamination proceeding will be extended only for sufficient cause and for a reasonable time specified. Any request for such extension must be filed on or before the day on which action by the patent owner is due, but in no case will the mere filing of a request effect any extension. Any request for such extension must be accompanied by the petition fee set forth in § 1.17(g). See § 1.304(a) for extensions of time for filing a notice of appeal to the U.S. Court of Appeals for the Federal Circuit or for commencing a civil action. (emphasis added)

MPEP § 2265 Extension of Time (in-part)

> ...
> *Ex parte* prosecution will be conducted by initially setting either a 1-month or a 2-month shortened period for response, see MPEP § 2263. The patent owner also will be given a 2-month period after the order for reexamination to file a statement >(by statute (35 U.S.C. § 304), this period cannot be less than 2-months, even in a proceeding where the patent is being litigated). See 37 CFR § 1.530(b). First requests for extensions of these statutory time periods will be granted for sufficient cause, and for a reasonable time specified — usually 1 month. The reasons stated in the request will be evaluated by the CRU or TC Director, and the requests will be favorably considered where there is a factual accounting of reasonably diligent behavior by all those responsible for preparing a response within the statutory time period. Second or subsequent requests for extensions of time or requests for more than 1 month will be granted only in extraordinary situations.
> ...

ANALYSIS AND FINDINGS

The patent owner's representative petitions to extend the period for response by adding thirty (30) days to the period for response. The decision to extend the period for response is evaluated based upon a showing of "sufficient cause." There is always the consideration to balance the need for the patent owner to have a fair opportunity to respond to the Office action between the need for special dispatch.

The patent owner timely submitted a first petition for extension which was granted on 26 October 2009. This first extension of time granted one additional month for which the patent owner to respond to the outstanding Office action. Second or subsequent requests for extensions of time or requests for more than 1 month will be granted only in extraordinary situations. The factual accounting presented by the petitioner does not meet the level of extraordinary circumstances.

The petition request to extend the response time is hereby dismissed.

CONCLUSION

1. The patent owner's petition for extension of time is hereby **dismissed.**

2. The time to respond continues to run.

3. Response is due on **27 November 2009.**

4. Response and/or submissions to the Office should be addressed as follows:

By Mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P. O. Box 1450
Alexandria, VA 22313-1450

By Fax to: (571) 273-9900
Central Reexamination Unit

By Hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

By EFS: Registered users of EFS-Web may alternatively submit such correspondence via the electronic filing system EFS-Web, at https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html. EFS-Web offers the benefit of quick submission to the particular area of the Office that needs to act on the correspondence. Also, EFS-Web submissions are "soft scanned" (i.e., electronically uploaded) directly into the official file for the reexamination proceeding, which offers parties the opportunity to review the content of their submissions after the "soft scanning" process is complete.

5. Telephone inquiries with regard to this decision should be directed to Mark Reinhart, at (571) 272-1611, in the event that Mark Reinhart is unavailable Eric Keasel at (571) 272-4929, or Jessica Harrison at (571) 272-4449; all are Supervisory Patent Examiners in the Central Reexamination Unit, Art Unit 3992 may also be contacted..

/Mark Reinhart/

Mark Reinhart,
Supervisory Patent Examiner,
AU 3992,
Central Reexamination Unit
571-272-1611

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 1 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	1-1	US-2003/0050075	2003/13/03	Rangarajan et al.
	1-2	US-2004/0204146	2004/14/10	Deeds
	1-3	US-2005/0032435	2005/10/02	Tischer et al.
	1-4	US-2005/0130611	2005/16/06	Lu et al.
	1-5	US-4332982	1982/01/06	Thomas
	1-6	US-4410765	1983/18/10	Hestad et al.
	1-7	US-4446519	1984/01/05	Thomas
	1-8	US-4450554	1984/22/05	Steensma, et al.
	1-9	US-4468529	1984/28/08	Samuel et al.
	1-10	US-4528659	1985/09/07	Jones, Jr.
	1-11	US-4589107	1986/13/05	Middleton, et al.
	1-12	US-4594477	1986/10/06	Noirot
	1-13	US-4598397	1986/01/07	Nelson, et al.
	1-14	US-4630262	1986/16/12	Callens, et al.
	1-15	US-4652703	1987/24/03	Lu, et al.
	1-16	US-4654483	1987/03	Imai et al.
	1-17	US-4694492	1987/09	Wirstrom et al.
	1-18	US-4740963	1988/26/04	Eckley
	1-19	US-4782485	1988/01/11	Gollub
	1-20	US-4799153	1989/01	Hann et al.
	1-21	US-4809271	1989/28/02	Kondo, et al.
	1-22	US-4813040	1989/14/03	Futato
	1-23	US-4819228	1989/04/04	Baran, et al.
	1-24	US-4821263	1989/04	Lundh

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 2 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	2-1	US-4829554	1989/09/05	Barnes et al.
	2-2	US-4837797	1989/06	Freeny, Jr., Charles C.
	2-3	US-4866704	1989/12/09	Bergman
	2-4	US-4866732	1989/12/09	Carey, et al.
	2-5	US-4873715	1989/10/10	Shibata
	2-6	US-4887265	1989/12/12	Felix
	2-7	US-4890282	1989/26/12	Lambert, et al.
	2-8	US-4912705	1990/27/03	Paneth, et al.
	2-9	US-4932022	1990/05/06	Keeney, et al.
	2-10	US-4981371	1991/01/01	Gurak, et al.
	2-11	US-4989230	1991/29/01	Gillig et al.
	2-12	US-4995074	1991/02	Goldman et al.
	2-13	US-5031089	1991/07	Liu et al.
	2-14	US-5036513	1991/30/07	Greenblatt
	2-15	US-5040141	1991/13/08	Yazima et al
	2-16	US-5056140	1991/10	Kimbell
	2-17	US-5065425	1991/12/11	Lecomte, et al.
	2-18	US-5107443	1992/04	Smith et al.
	2-19	US-5121385	1992/09/06	Tominaga, et al.
	2-20	US-5127003	1992/30/06	Dell Jr. et al
	2-21	US-5130985	1992/14/07	Kondo, et al.
	2-22	US-5150410	1992/22/09	Bertrand
	2-23	US-5155726	1992/13/10	Spinney, et al.
	2-24	US-5157592	1992/20/10	Walters

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 3 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	3-1	US-5187591	1993/16/02	Guy, et al.
	3-2	US-5212789	1993/18/05	Rago
	3-3	US-5214650	1993/25/05	Renner, et al.
	3-4	US-5220599	1993/06	Sasano et al
	3-5	US-5241594	1993/31/08	Kung
	3-6	US-5241625	1993/08	Epard et al.
	3-7	US-5247620	1993/09/21	Fukuzawa et al.
	3-8	US-5249290	1993/28/09	Heizer
	3-9	US-5274635	1993/12	Rahman et al.
	3-10	US-5282197	1994/25/01	Kreitzer
	3-11	US-5283819	1994/01/02	Glick, et al.
	3-12	US-5287103	1994/15/02	Kasprzyk Marlon Z et al
	3-13	US-5305312	1994/19/04	Fornek, et al.
	3-14	US-5327486	1994/07	Wolff et al.
	3-15	US-5335276	1994/02/08	Thompson et al.
	3-16	US-5341374	1994/23/08	Lewen et al.
	3-17	US-5347632	1994/09	Filepp et al.
	3-18	US-5377260	1994/12	Long
	3-19	US-5396485	1995/03	Ohno et al.
	3-20	US-5410754	1994/15/02	Favreau Keith et al
	3-21	US-5428608	1995/06	Freeman et al.
	3-22	US-5432846	1995/11/07	Norio
	3-23	US-5440547	1995/08/08	Esaki Hiroshi et al
	3-24	US-5446891	1995/29/08	Kaplan et al.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 4 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	4-1	US-5446919	1995/29/08	Wilkins
	4-2	US-5457738	1995/10/10	Sylvan
	4-3	US-5459864	1995/10	Brent et al.
	4-4	US-5461611	1995/10	Drake, Jr. et al.
	4-5	US-5465286	1995/07/11	Clare et al.
	4-6	US-5467388	1995/11	Redd et al.
	4-7	US-5473531	1995/10	Flora-Holmquist et al.
	4-8	US-5474741	1995/12/12	Mikeska et al.
	4-9	US-5474819	1995/12/12	Chambers et al.
	4-10	US-5475741	1995/12/12	Davis et al.
	4-11	US-5483524	1996/01	Lev et al.
	4-12	US-5487100	1996/01	Kane
	4-13	US-5491800	1996/13/02	Goldsmith et al.
	4-14	US-5500890	1996/03	Rogge et al.
	4-15	US-5509058	1996/16/04	Sestak et al.
	4-16	US-5517432	1996/05	Chandra et al.
	4-17	US-5524141	1996/04/06	Braun et al.
	4-18	US-5528671	1996/18/06	Ryu et al.
	4-19	US-5533102	1996/02/07	Robinson et al.
	4-20	US-5544164	1996/08	Baran
	4-21	US-5544322	1996/06/08	Cheng et al.
	4-22	US-5546448	1996/08	Caswell et al.
	4-23	US-5546452	1996/13/08	Andrews et al.
	4-24	US-5548636	1996/08	Bannister et al.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 5 of 30	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	5-1	US-5548694	1996/08	Frisken Gibson
	5-2	US-5563882	1996/10	Bruno et al.
	5-3	US-5572643	1996/11	Judson, David H.
	5-4	US-5574774	1996/11	Ahlberg et al.
	5-5	US-5574934	1996/12/11	Mirashrafi et al.
	5-6	US-5581522	1996/12/03	Shibuya et al.
	5-7	US-5581702	1996/12	McArdle et al.
	5-8	US-5586257	1996/17/12	Perlman
	5-9	US-5586260	1996/17/12	Hu
	5-10	US-5591800	1996/02	Goldsmith et al.
	5-11	US-5604737	1997/02	Iwami et al.
	5-12	US-5606669	1997/25/02	Bertin et al.
	5-13	US-5614940	1997/25/03	Cobbley et al.
	5-14	US-5619557	1997/08/04	Van Berkum
	5-15	US-5623483	1997/22/04	Agrawal et al.
	5-16	US-5623490	1997/04	Richter et al.
	5-17	US-5623605	1997/22/04	Keshav et al.
	5-18	US-5625407	1997/29/04	Biggs et al.
	5-19	US-5636282	1997/06	Holmquist et al.
	5-20	US-5636346	1997/06	Saxe
	5-21	US-5642156	1997/24/06	Saiki
	5-22	US-5644629	1997/07	Chow
	5-23	US-5651006	1997/07	Fujino et al.
	5-24	US-5652759	1997/07	Stringfellow, Jr.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609 Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 6 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	6-1	US-5655120	1997/08	Witte et al.
	6-2	US-5659542	1997/08	Bell et al.
	6-3	US-5659596	1997/19/08	Dunn
	6-4	US-5668862	1997/09	Bannister et al.
	6-5	US-5671428	1997/23/09	Muranaga et al.
	6-6	US-5675507	1997/10	Bobo
	6-7	US-5680392	1997/10	Semaan
	6-8	US-5684800	1997/11	Dobbins et al.
	6-9	US-5689553	1997/18/11	Ahuja et al.
	6-10	US-5692192	1997/11	Sudo
	6-11	US-5694594	1997/02/12	Chang
	6-12	US-5701463	1997/23/12	Malcolm
	6-13	US-5708422	1998/13/01	Blonder et al.
	6-14	US-5708655	1998/01	Toth et al.
	6-15	US-5710884	1998/20/01	Dedrick
	6-16	US-5717923	1998/10/02	Dedrick
	6-17	US-5719786	1998/02	Nelson et al.
	6-18	US-5721827	1998/24/02	Logan et al.
	6-19	US-5724092	1998/03/03	Davidsohn et al.
	6-20	US-5724412	1998/03	Srinivasan
	6-21	US-5724506	1998/03/03	Cleron et al.
	6-22	US-5726984	1998/10/03	Kubler et al.
	6-23	US-5729748	1998/03	Robbins et al.
	6-24	US-5732078	1998/03	Arango

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 7 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	7-1	US-5736968	1998/04	Tsakiris, Alexander L.
	7-2	US-5742668	1998/04	Pepe et al.
	7-3	US-5742675	1998/21/04	Kilander et al.
	7-4	US-5742762	1998/21/04	Scholl et al.
	7-5	US-5742905	1998/04	Pepe et al.
	7-6	US-5745642	1998/28/04	Ahn
	7-7	US-5745702	1998/04	Morozumi
	7-8	US-5745711	1998/04	Kitahara et al.
	7-9	US-5751712	1998/05	Farwell et al.
	7-10	US-5751961	1998/05	Smyk
	7-11	US-5754636	1998/05	Bayless et al.
	7-12	US-5754939	1998/19/05	Herz et al.
	7-13	US-5758110	1998/05	Boss et al.
	7-14	US-5758257	1998/26/03	Herz et al.
	7-15	US-5761606	1998/06	Wolzien
	7-16	US-5764736	1998/06	Shachar et al.
	7-17	US-5764741	1998/09/06	Barak
	7-18	US-5764756	1998/09/06	Onweller
	7-19	US-5767897	1998/16/06	Howell
	7-20	US-5768527	1998/06	Zhu et al.
	7-21	US-5771355	1998/23/06	Kuzma
	7-22	US-5774660	1998/30/06	Brendel et al.
	7-23	US-5774666	1998/30/06	Portuesi
	7-24	US-5778181	1998/07	Hidary et al.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 8 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	8-1	US-5778187	1998/07	Monteiro et al.
	8-2	US-5784564	1998/21/07	Camaisa et al.
	8-3	US-5784619	1998/21/07	Evans et al.
	8-4	US-5787253	1998/07	McCreery et al.
	8-5	US-5790548	1998/08	Sistanizadeh et al.
	8-6	US-5790792	1998/08	Dudgeon et al.
	~~8-7~~	~~US-5790793~~	~~1998/08~~	~~Higley~~
	8-8	US-5793365	1998/11/08	Tang et al.
	8-9	US-5794018	1998/08	Vrvilo et al.
	8-10	US-5794257	1998/08	Liu et al.
	8-11	US-5796394	1998/08	Wicks et al.
	8-12	US-5799063	1998/08	Krane
	8-13	US-5799072	1998/08	Vulcan et al.
	8-14	US-5799150	1998/08	Hamilton et al.
	8-15	US-5805587	1998/09	Norris et al.
	8-16	US-5805810	1998/09	Maxwell
	8-17	US-5805822	1998/08/09	Long et al.
	8-18	US-5809233	1998/15/09	Shur
	8-19	US-5812819	1998/22/08	Rodwin et al.
	8-20	US-5816919	1998/10	Scagnelli et al.
	8-21	US-5818510	1998/10	Cobbley et al.
	8-22	US-5818836	1998/10	DuVal
	8-23	US-5822524	1998/10	Chen et al.
	8-24	US-5828837	1998/10	Eikeland

Examiner Signature		Date Considered	

*Examiner. Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
FORM PTO-1449 (modified)

Sheet 9 of 30

Reexam number	90/010,416
First Named Inventor	Hutton
Patent Under Re-Exam	6108704
Issue Date	2000/08/22
Group Art Unit	3992
Examiner Name	KOSOWSKI, ALEXANDER J
Attorney Docket No.	2655-0188
Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	9-1	US-5828843	1998/27/10	Grimm et al.
	9-2	US-5828846	1998/10	Kirby et al.
	9-3	US-5832119	1998/11	Rhoads
	9-4	US-5832240	1998/11	Larsen et al.
	9-5	US-5835720	1998/10/11	Nelson et al.
	9-6	US-5835723	1998/11	Andrews et al.
	~~9-7~~	~~US-5835725~~	~~1998/10/11~~	~~Chiang et al.~~
	9-8	US-5838683	1998/11	Corley et al.
	9-9	US-5838970	1998/17/11	Thomas
	9-10	US-5841769	1998/11	Okanoue et al.
	9-11	US-5842216	1998/11	Anderson et al.
	9-12	US-5848143	1998/08/12	Andrews et al.
	9-13	US-5848396	1998/12	Gerace
	9-14	US-5854901	1998/12	Cole et al.
	9-15	US-5857072	1999/01	Crowle
	9-16	US-5864684	1999/26/01	Nielsen
	9-17	US-5867156	1999/02	Beard et al.
	9-18	US-5867654	1999/02/02	Ludwig et al.
	9-19	US-5867665	1999/02/02	Butman et al.
	9-20	US-5872850	1999/16/02	Klein et al.
	9-21	US-5872922	1999/02	Hogan et al.
	9-22	US-5872972	1999/02	Boland et al.
	9-23	US-5884032	1999/03	Bateman et al.
	9-24	US-5884035	1999/16/03	Butman et al.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 10 of 30	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	10-1	US-5884077	1999/03	Suzuki
	10-2	US-5890162	1999/03	Huckins
	10-3	US-5892825	1999/06/04	Mages et al.
	10-4	US-5892903	1999/06/04	Klaus
	10-5	US-5892924	1999/04	Lyon et al.
	10-6	US-5903721	1999/11/05	Sixtus
	~~10-7~~	~~US-5903723~~	~~1999/11/05~~	~~Beck et al.~~
	10-8	US-5903727	1999/11/05	Nielsen
	10-9	US-5905719	1999/05	Arnold et al.
	10-10	US-5905736	1999/18/05	Ronen et al.
	10-11	US-5905865	1999/05	Palmer et al.
	10-12	US-5905872	1999/05	DeSimone et al.
	10-13	US-5915001	1999/06	Uppaluru
	10-14	US-5923736	1999/07	Shachar, Yuval
	10-15	US-5924093	1999/07	Potter et al.
	10-16	US-5925103	1999/10/07	Magallanes et al.
	10-17	US-5928327	1999/07	Wang et al.
	10-18	US-5929849	1999/07	Kikinis
	10-19	US-5937162	1999/08	Funk et al.
	10-20	US-5946386	1999/08	Rogers et al.
	10-21	US-5946629	1999/08	Sawyer et al.
	10-22	US-5950123	1999/09	Schwelb et al.
	10-23	US-5950172	1999/09	Klingman
	10-24	US-5956482	1999/09	Agraharam et al.

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609 Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 11 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	11-1	US-5961584	1999/10	Wolf
	11-2	US-5964872	1999/12/10	Turpin
	11-3	US-5969967	1999/10	Aahlad et al.
	11-4	US-5982774	1999/11	Foladare et al
	11-5	US-5983005	1999/11	Monteiro et al.
	11-6	US-5999965	1999/07/12	Kelly
	11-7	US-6005870	1999/21/12	Leung
	11-8	US-6006257	1999/12	Slezak
	11-9	US-6014379	2000/01	White et al.
	11-10	US-6014710	2000/01	Talluri et al.
	11-11	US-6016393	2000/18/01	White et al.
	11-12	US-6018768	2000/01	Ullman et al.
	11-13	US-6018771	2000/25/01	Hayden
	11-14	US-6021126	2000/02	White et al.
	11-15	US-6026086	2000/02	Lancelot et al.
	11-16	US-6026425	2000/02	Suguri et al.
	11-17	US-6029175	2000/22/02	Chow et al.
	11-18	US-6032192	2000/29/02	Wegner et al.
	11-19	US-6041345	2000/03	Levi et al.
	11-20	US-6047292	2000/04/04	Kelly et al.
	11-21	US-6055594	2000/04	Lo et al.
	11-22	US-6061716	2000/05	Moncreiff
	11-23	US-6064975	2000/05	Moon et al.
	11-24	US-6065048	2000/16/05	Highley

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 12 of 30	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	12-1	US-6069890	2000/30/05	White et al.
	12-2	US-6085217	2000/07	Ault et al.
	12-3	US-6101182	2000/08	Sistanizadeh et al.
	12-4	US-6105053	2000/08	Kimmel et al.
	12-5	US-6122255	2000/09	Bartholomew et al.
	12-6	US-6125113	2000/09	Farris et al.
	12-7	US-6137877	2000/24/10	Robin et al.
	12-8	US-6141341	2000/31/10	Jones et al.
	12-9	US-6151643	2000/11	Cheng et al.
	12-10	US-6154445	2000/11	Farris et al.
	12-11	US-6163316	2000/12	Killian
	12-12	US-6173044	2001/01	Hortensius et al.
	12-13	US-6178453	2001/23/01	Mattaway et al.
	12-14	US-6181689	2001/01	Choung et al.
	12-15	US-6185184	2001/06/02	Mattaway et al.
	12-16	US-6188677	2001/02	Oyama et al.
	12-17	US-6195357	2001/02	Polcyn
	12-18	US-6198303	2001/03	Rangasayee
	12-19	US-6205135	2001/20/03	Chinni et al.
	12-20	US-6212625	2001/04	Russell
	12-21	US-6226678	2001/05	Mattaway et al.
	12-22	US-6226690	2001/05	Banda et al.
	12-23	US-6240444	2001/05	Fin et al.
	12-24	US-6243373	2001/06	Turock

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 13 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	13-1	US-6266539	2001/07	Pardo
	13-2	US-6275490	2001/14/08	Mattaway et al.
	13-3	US-6282272	2001/08	Noonen et al.
	13-4	US-6289369	2001/11/09	Sundaresan
	13-5	US-6300863	2001/10	Cotichini et al.
	13-6	US-6338078	2002/01	Chang et al.
	13-7	US-6343115	2002/01	Foladare et al.
	13-8	US-6343220	2002/29/01	Van Der Salm
	13-9	US-6347085	2002/02	Kelly
	13-10	US-6347342	2002/02	Marcos et al.
	13-11	US-6377568	2002/04	Kelly
	13-12	US-6385583	2002/05	Ladd et al.
	13-13	US-6393455	2002/05	Eilert et al.
	13-14	US-6427064	2002/07	Henderson, Daniel
	13-15	US-6434552	2002/08	Leong, Jin Fye
	13-16	US-6463565	2002/08/10	Keith C. Kelly
	13-17	US-6477586	2002/11	Achenson et al.
	13-18	US-6594254	2003/15/07	Kelly
	13-19	US-6687738	2004/03/02	Glenn W. Hutton
	13-20	US-6704802	2004/03	Finch et al.
	13-21	US-6728784	2004/27/04	Mattaway
	13-22	US-6772335	2004/08	Curtis et al.
	13-23	US-6829645	2004/07/12	Glenn W. Hutton
	13-24	US-6888836	2005/03/05	Cherkasova

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 14 of 30	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication/ Issue Date	Name of Patentee or Applicant of Cited Document
	14-1	US-6909708	2005/06	Krishnaswamy et al.
	14-2			
	14-3			
	14-4			
	14-5			
	14-6			
	14-7			
	14-8			
	14-9			
	14-10			
	14-11			
	14-12			
	14-13			
	14-14			
	14-15			
	14-16			
	14-17			
	14-18			
	14-19			
	14-20			
	14-21			
	14-22			
	14-23			
	14-24			

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 15 of 30	Confirmation No.	1061

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document No.	Publication Date	Name of Patentee or Applicant of Cited Document	Notes
	15-1	AU-200059377-A1	11-23-2000	Glenn W. Hutton et al.	
	15-2	AU-200059378-A1	11-30-2000	Glenn W. Hutton et al.	
	15-3	AU-200059379-A1	11-23-2000	Glenn W. Hutton et al.	
	15-4	EP-0518596	12/16/1992	Digital Equipment Corp	
	15-5	EP-0559047	09-1993	Iglehart	
	15-6	EP-0597691	05/18/1994	IBM	
	15-7	EP-0632672	01/04/1995	IBM	
	15-8	EP-0648038	04/12/1995	IBM	
	15-9	EP-1379039-A2	01-07-2004	Glenn W. Hutton	
	15-10	EP-1379050-A2	01-07-2004	Glenn W. Hutton	
	15-11	GB-2283645	05/10/1995	Digital Equipment Int.	
	15-12	JP 63-131637	3/6/1988	Y. Takehiko (w/ English abstract)	T
	15-13	JP 6-62020 (w/ SOR and EA	1994/03/04	Masatoshi et al.	
	15-14	JP-5944140	03-12-1984	Junichi Kimura et al.	T
	15-15	WO-9422087	09/29/1994	3Com Corp	
	15-16	WO-9714234	04-17-1997	Glenn W. Hutton	
	15-17	WO-9811704	03/19/1998	Don Joon Lee et al.	
	15-18				
	15-19				
	15-20				
	15-21				
	15-22				
	15-23				
	15-24				

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided. EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 16 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	16-1	"Circuit Switching", Ericsson, last published July 5, 2001, found at http://www.ericsson.com/multiservicenetworks/circuitswitching/axe/ printed on August 1, 2001, 2 pages.	
	16-2	"Data Communication Over the Telephone Network". International Telecommunication Union, CCITT The International Telegraph and Telephone Consultative Committee, Blue Book, Volume VIII - Fascicle VIII.1, IXth Plenary Assembly, Melbourne, November 14-25, 1988, pages 296-370	
	16-3	"Full Duplex Speakerphone", IBM Technical Disclosure Bulletin, Vol. 29, No. 12, May 1987, pages 5599-5602	
	16-4	"ICL OPD - One Per Desk", Issue 01 August 1990, A Comprehensive Technical Information Document (24 pages)	
	16-5	"Information Processing Techniques Program. Volume II. Wideband Integrated Voice/Data Technology" Semiannual Technical Summary Report, Massachusetts Institute of Technology Lexington, MA, 1 October 1977 - 31 March 1978, Issued 31 August 1978, pages 1-25 and 27-31, ADA067014	
	16-6	"Integrated Voice/Data PABX Communications", IBM Technical Disclosure Bulletin, September 1986, http://patents.ibm.com	
	16-7	"Level 1-5 of 65 Stories" 1990 Network World, Inc., April 16, 1990, pages 114-115	

Examiner Signature		Date Considered	

*Examiner Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 17 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	17-1	"Multi-Service Networks", Ericsson, last published June 27, 2001, found at http://www.ericsson.com/multiservicenetworks/circuitswitching/ printed on August 1, 2001, 2 pages.	
	17-2	"The History of TPC.INT", January 15, 1999, 2 pages, found at http://www.tpc.int/faq/history.html printed on August 8, 2002.	
	17-3	A. A. Kapauan, et al. "Wideband Packet Access for Workstations: Integrated Voice/Data/Image Services on the UNIX+ PC", IEEE Global Telecommunications Conference, Houston, Texas, December 1-4, 1986, Conference Record Vol. 3, pages 1439-1441	
	17-4	AHRENS, Richard L., "Frequently-Asked Questions about Internet VoiceChat 1.1 FAQ Version: 1.0", 1994, 6 pages.	
	17-5	Andy Hopper "Pandora - An Experimental System for Multimedia Applications", Operating Systems Review, January 12, 1990, pages 1-16	
	17-6	BENNETT, Geoff, "Designing TCP/IP Internetworks", Chapter 11, pages 290, 291 and 323, Van Nostrand Reinhold, 1995.	
	17-7	Bernard Gold "Digital Speech Networks", Proceedings of the IEEE, Vol. 65, No. 12, December 1977, pages 1636-1658	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 18 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	18-1	Bill Newman "An ISDN Data and Voice Terminal Based on a Personal Computer", Globecom'85, IEEE Global Telecommunications Conference, Conference Record Volume 3, New Orleans, Louisiana, December 2-5, 1985, pages 1048-1052	
	18-2	BORLAND, John, "Technology uses one number to find you on any device", May 17, 2001, 3 pages, found at http://news.cnet.com/news/0-1004-201-5939191-0.html.	
	18-3	C. MALAMUD et al., "Principles of Operation for the TPC.INT Subdomain: General Principles and Policy", RFC 1530, October 1993, pages 1-7.	
	18-4	C. MALAMUD et al., "Principles of Operation for the TPC.INT Subdomain: Remote Printing -- Administrative Policies", RFC 1529, October 1993, pages 1-5.	
	18-5	C. MALAMUD et al., "Principles of Operation for the TPC.INT Subdomain: Remote Printing -- Technical Procedures", RFC 1528, October 1993, pages 1-12.	
	18-6	C. Topolcic "Experimental Internet Stream Protocol, Version 2 (ST-II)", Request for Comments 1190, October 1990, pages 1-148	
	18-7	C. YANG, "INETPhone: Telephone Services and Servers on Internet", Request for Comments 1789, pp. 1-6, April 1995.	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 19 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	19-1	Carl A. Sunshine, et al. "Broad-Band Personal Computer LAN's", IEEE Journal on Selected Areas in Communications, Vol. SAC-3, No.3, May 1985, pages 408-415	
	19-2	Cindy MUELLER et al., "ATD Data Services", http://www.iita.ucar.edu/ws/datawkshop/Abstract-ATD.html, January 5, 1995, 2 pages.	
	19-3	Clifford J. Weinstein, et al. "Experience with Speech Communication in Packet Networks" IEEE Journal on Selected Areas in Communications, Vol SAC-1, No 6, (ISSN 0733-8716), December 1983, pages 963-980	
	19-4	D. Adolphs, et al. "Adapters for the Public ISDN", pages 72-80	
	19-5	D. Perkins "The Point-to-Point Protocol for the Transmission of Multi-Protocol Datagrams Over Point-to-Point Links", Request for Comments 1171, ftp://ftp.isi.edu/in-notes/rfc1171.txt, July 1990, pages 1-48	
	19-6	D.C. Swinehart et al., "Adding Voice to an Office Computer Network", IEEE Global Telecommunications Conference, Nov. 28-Dec 1, 1983, Conference Record Volume 1 of 3, pages 392-398	
	19-7	Dale Gulick et al., "Interface the ISDN to Your PC With a Voice/Data Board", Design Applications, 2328 Electronic Design, 35 (1987) Dec. 10, No. 29, Hashbrouck Heights, NJ, USA, pages 85-88, XP 000004313	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 20 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	20-1	Daniel C. Swinehart "Telephone Management in the Etherphone System", IEEE/IEICE Global Telecommunications Conference '87, Conference Record Volume 2 of 3, November 15-18, 1987, pages 1176-1180	
	20-2	Danny Cohen "A Network Voice Protocol NVP-II", April 1, 1981, pages 1-68	
	20-3	Danny Cohen "Packet Communication of Online Speech", AFIPS Conference Proceedings, 1981 National Computer Conference, May 4-7, 1981, Chicago, Illinois, pages 169-176	
	20-4	Danny Cohen "Specifications for the Network Voice Protocol (NVP)", Request for Comments 741, January 29, 1976, pages 1-30	
	20-5	Don H. Johnson, et al. "A Local Access Network for Packetized Digital Voice Communication", IEEE Transactions on Communications, Vol. Com. 29, No. 5, May 1981, pages 679-688	
	20-6	Douglas B. Terry and Daniel C. Swinehart, "Managing Stored Voice in the Etherphone System", 1987 ACM 089791-242-X/87/0011/0103, pages 103-104	
	20-7	Douglas B. Terry and Daniel C. Swinehart, "Managing Stored Voice in the Etherphone System", ACM Transactions on Computer Systems, Vol. 6, No. 1, February 1988, pages 3-27	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 21 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	21-1	Eve M. Schooler, et al. "A Packet-Switched Multimedia Conferencing System", SIGOIS Bulletin, pages 12-22	
	21-2	Gary C. Kessler "ISDN Concepts, Facilities, and Services", McGraw-Hill, Inc., c1990, pages 224-231, ISBN 0-07-034242-3	
	21-3	Giulio Barberis, et al. "Coded Speech in Packet-Switched Networks: Models and Experiments" IEEE Journal on Selected Areas in Communications, Vol. SAC-1, No. 6, December 1983, pages 1028-1038	
	21-4	H. Jonathan Chao, et al. "A Packet Video System Using the Dynamic Time Division Multiplexing Technique", IEEE Global Telecommunications Conference, Houston, Texas, December 1-4, 1988, Conference Record, Vol. 3, pages 0767-0772	
	21-5	H. Opderbeck "Throughput Degredations for Single Packet Messages", Request for Comments 632, ftp://ftp.isi.edu/in-notes/rfc632.txt, May 20, 1974, pages 1-6	
	21-6	Henning Schulzrinne "Voice Communication Across the Internet: A Network Voice Terminal", July 29, 1992, pages 1-34	
	21-7	Hiroshi Kobayashi and Hideaki Haruyama, "Voice, Data and Video Integrated Broadband Metropolitan Area Network", Electronics and Communications in Japan, Part 1, Vol. 73, No. 11, 1990, pages 34-42	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 22 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	22-1	Hiroyuki Ichikawa et al. "High-Speed Packet Switching Systems for Multimedia Communications", IEEE Journal on Selected Areas in Communications, October 1987, Volume SAC-5, Number 8 (ISSN 0733-8716), pages 1336-1345	
	22-2	Ian H. Merritt "Providing Telephone Line Access to a Packet Voice Network", University of Southern California, Marina Del Rey. Information Sciences Inst., February 1983, ADA126270	
	22-3	Implementation of Next-Generation Agent-Dedicated Communications, by Agatsuma et al., Tech Report of IEICE 94-216 (March 1995)	
	22-4	International Preliminary Examination Report (IPER) issued March 26, 1998 in corresponding International Application Serial No. PCT/US96/15504.	
	22-5	International Search Report issued January 27, 1998 in corresponding International Application Serial No. PCT/US96/15504.	
	22-6	Israel GITMAN, et al. "Economic Analysis of Integrated Voice and Data Networks: A Case Study" Proceedings of the IEEE, Vol. 66, No. 11, November 1978, pages 1549-1570	
	22-7	J. Huelamo, et al. "End User Premises Equipment and Terminals for Broadband Applications", Electrical Communication, Volume 64, Number 2/3, 1990	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 23 of 30	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	23-1	J. K. Reynolds et al., "Voice File Interchange Protocol (VFIP), Request for Comments 978, ftp://ftp.isi.edu/innotes/rfc978.txt, February 1986, pages 1-5	
	23-2	J. Romkey "A Nonstandard For Transmission of IP Datagrams Over Serial Lines: Slip", Request for Comments 1055, ftp://ftp.isi.edu/in-notes/std/std47.txt, June 1988, pages 1-6	
	23-3	James D. Mills, et al. "A data and voice system for the general service telephone network", Proceedings IECON '87, 1987 International Conference on IND Electronics, Control, and Instrumentation, Cambridge, Massachusetts, November 3-6, 1987	
	23-4	James W. Forgie "Speech Transmission in Packet-Switched Store-and-Forward Networks", AFIPS Conference Proceedings,1975 National Computer Conference, May 19-22, 1975, Anaheim, California, pages 137-142	
	23-5	James W. Forgie "Voice Conferencing in Packet Networks", ICC '80, Conference Record, International Conference on Communications, Seattle, WA, June 8-12, 1980, Volume 1, 80CH1505-6 CSCB, pages 21.3.1-21.3.4	
	23-6	Jane's Military Communications 1979-80, pages 452 and 453	
	23-7	Jane's Military Communications 1985, pages 585, 546, and 545	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 24 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	24-1	Jane's Military Communications 1989, Tenth Edition, Edited by John Williamson, ISBN 0710608772, pages 443, 507, and 512	
	24-2	Jane's Military Communications 1990-91, Eleventh Edition, Edited by John Williamson, ISBN 0710609000, pages [30], 264, 357, 398, 406, 450, 454, 456, 560, 572, 573, 814, 815, and 816	
	24-3	Jane's Military Communications 1992-93, Thirteenth Edition, Edited by John Williamson, ISBN 0710609809, pages 375, 376, 384, and 704	
	24-4	Jim Stevens, "Much More Idle Chatter About Reference Models", http://www-mice.cs.ucl.ac.uk/multimedia/misc/tcp_ip/8709.mm.www/0041.html, December 18, 1987, pages 1-9	
	24-5	John Bellamy, "Digital Telephony", c1982 John Wiley & Sons, Inc., pages 392-397 and 410-412	
	24-6	JP Appln. No. 2008-163825 Office Action (Translation)	
	24-7	K. Sohraby, et al. "ISDN Primary Rate Interface Impact on Performance of Integrated Voice and Data on CSMA/CD Networks – A Measurement and Simulation Study", Globecom '90 IEEE Global Telecommunication Conference & Exhibition, San Diego, California, December 2-5, 1990. Volume 2, pages 0912-0919	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 25 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	25-1	Ken Sherman "Data Communications - A User's Guide", 3rd Edition, c1981 Prentice-Hall, Inc., pages 296-307 and 404-407	
	25-2	Kevin Jeffay, et al. "Kernel Support for Live Digital Audio and Video", pages 10-21, University of North Carolina at Chapel Hill, Department of Computer Science	
	25-3	Kyuta SAITO, et al. "Voice Packet Communication System for Private Networks", Globecom '89, IEEE Global Telecommunications Conference & Exhibition, Dallas, Texas, November 27-30, 1989, Volume 3, pages 1874-1878	
	25-4	Lawrence G. Roberts "The Evolution of Packet Switching", Proceedings of the IEEE, Vol. 66, No. 11, November 1978, pages 1307-1313	
	25-5	LIN, Hwa-Chun and C.S. Raghavendra, "A Dynamic Load-Balancing Policy With a Central Job Dispatcher (LBC)," IEEE Transactions on Software Engineering, Vol. 18, No. 2, February 1992, pages 148-158.	
	25-6	M. E. Ulug, et al. "Statistical Multiplexing of Data and Encoded Voice in a Transparent Intelligent Network", Fifth Data Communications Symposium, September 27-29, 1977, Snowbird, Utah, pages 6-14 - 6-20	
	25-7	M. Gopalakrishnan, et al. "Integrating Voice and Data SALAN: An Experimtental Local Area Network", Computer Communications, Vol. 9, No. 4, August 1986, pages 186-194 and page 169	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 26 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	26-1	M.J. Ross "Alternatives for Integrating Voice and Data", 1981 International Switching Symposium, ISS' 81 CIC Montreal, September 21-25, 1981	
	26-2	Natesa Janakiraman "An Overview of Recent Developments in the Designs and Applications of Customer Premises Switches", IEEE Communications Magazine, October 1985, Vol. 23, No. 10, pages 32-45	
	26-3	P. Borgnis-Desbordes, et al. "Variable-Speed Data Transmission", IBM Technical Disclosure Bulletin, Vol. 27, No. 4A, September 1984, pages 2269-2270	
	26-4	P. Venkat Rangan and Daniel C. Swinehart, "Software Architecture for Integration of Video Services in the Etherphone System", IEEE Journal on Selected Areas in Communication, Vol. 9, No. 9, December 1991, pages 1395-1404	
	26-5	Paul Francis, "Comparison of Geographical and Provider-rooted Internet Addressing," Computer Networks and ISDN Systems 27(3)437-448, 1994 (selected paper from INET 94/JENC 5)	
	26-6	Paul Gilster, "Internet Navigator", Maruzen Kabushiki-Kaisha (1st Ed.), pgs. 473-476, Feb. 28, 1995 (with translation and SOR)	
	26-7	Paul Tsuchiya, Tony Eng, "Extending the IP Internet Through Address Reuse," ACM SIGCOMM Computer Communications Review, 23(1):16-33, Jan. 1993	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 27 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	27-1	Philip H. Reagan, "Is it the PBX or is it the LAN?", Datamation, The Telecom Manager Emerges, 03/1984, Volume 30 Number 3, pages 3-4, 147, 148, 150	
	27-2	Polle T. Zellweger et al., "An Overview of the Etherphone System and its Applications", 2nd IEEE Conference on Computer Workstations, March 7-10, 1988, pages 160-168	
	27-3	R. BRAUDES et al., "Requirements for Multicast Protocols", Request for Comments 1458, Network Working Group, May 1993, pp. 1-19.	
	27-4	R. W. Meba, et al. "Experiments in Wideband Packet Technology", Digital Communications - New Directions in Switching and Networks, Proceedings of the International Seminar, Zurich, Switzerland, March 11-13, 1986, pages 135-139	
	27-5	R.P. McNamara, "Some Considerations of the Voice-Data Capabilities of Broadband Cable Networks", IEEE Digest of Papers Spring CompCon 82, February 22-25, 1982, pages 312-314	
	27-6	Randy Cole "Packet Voice: When It Makes Sense", Speech Technology, September/October 1982, pages 52-61.	
	27-7	Scott FLINN, "Coordinating Heterogeneous Time-Based Media Between Independent Applications" ACM Multimedia 95 - Electronic Proceedings November 5-9, 1995, pages 1-16.	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided. EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 28 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	28-1	Shimmi Hattori et al., "Integrated Digital Switching System with Queueing Storage Facility", IEEE Transactions on Communications, Vol. Com-30, No. 8, August 1982, pages 1900-1905, (ISSN 0090-6778)	
	28-2	Steve Oltmanns, et al. "A Voice and Communications System for the IBM PC", Speech Technology, March/April 1986, pages 94-99	
	28-3	Stuart CHESHIRE et al., "Internet Mobility 4x4", www.acm.org, 1996, pages 1-12.	
	28-4	Susan Angebranndt et al., "Integrating Audio and Telephony in a Distributed Workstation Environment", Proceedings of the Summer 1991 USENIX Conference, June 10-14, 1991, Nashville, Tennessee, pages 419-435	
	28-5	T. Kamae "Visual Terminals and User Interfaces", FGCS North-Holland, pages 257-278	
	28-6	T. Kamae "Voice/Data Integration in the INS Model System and Local Area Networks" IEEE Communications Magazine, December 1986, Vol. 24, No. 12, pages 7-15	
	28-7	T7540 Digital Telephone Codec, AT&T Microelectronics, January 1991, pages 1-62 and Data Sheet Addendum, July 1991, 4 pages	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified) Sheet 29 of 30		
	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	29-1	Takashi Yamada, et al. "New Technologies - Multimedia High-throughput X.25 Packet Switching System", NTT Review, Vol. 1, No. 2, July 1989, pages 82-88	
	29-2	talk (software) description from WikiPedia	
	29-3	Tamohiro Kawai, Nikkei Communications, No. 202, pgs 29-30, Nikkei BP, July 17, 1995 ("Communication software appears on the Internet") (w/ SOR)	
	29-4	Theodore Bially, et al. "Voice Communication in Integrated Digital Voice and Data Networks", IEEE Transactions on Communications, Vol. Com-28, No. 9, September 1980, pages 1478-1490	
	29-5	Toru Tsuda, et al. "An Approach to Multi-Service Subscriber Loop System Using Packetized Voice/Data Terminals" ISSLS '78, The International Symposium on Subscriber Loops and Services, March 20-24, 1978, Atlanta, Georgia, Conference Record, pages 161-165	
	29-6	Translation of Japanese Kokai H07-129488 (published May 19, 1995)	
	29-7	U.S. Reexam Control No. 90/010,421 - 2009-08-14 PTO Office Action	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT FORM PTO-1449 (modified)	Reexam number	90/010,416
	First Named Inventor	Hutton
	Patent Under Re-Exam	6108704
	Issue Date	2000/08/22
	Group Art Unit	3992
	Examiner Name	KOSOWSKI, ALEXANDER J
	Attorney Docket No.	2655-0188
Sheet 30 of 30	Confirmation No.	1061

NON-PATENT REFERENCES

Examiner Initials*	Cite No.	Non-patent Reference bibliographic information, where available	Notes
	30-1	U.S. Reexam Control No. 90/010,422 - 2009-08-25 PTO Office Action	
	30-2	U.S. Reexam Control No. 90/010,424 - 2009-08-25 PTO Office Action	
	30-3	V. Jacobson, et al. "TCP Extension for High-Speed Paths", Request for Comments 1185, ftp://ftp.isi.edu/in-notes/rfc1185.txt, October 1990, pages 1-21	
	30-4	V. Jacobson, et al. "TCP Extensions for High Performance", Request for Comments 1323, ftp://ftp.isi.edu/in-notes/rfc1323.txt, May 1992, pages 1-37	
	30-5	Vinton G. Cerf, "Packet Satellite Technology Reference Sources", Request for Comments 829, November 1982, http://www.cis.ohio-state.edu/htbin/rfc/rfc829.html, pages 1-5	
	30-6	VocalTec Internet Phone (TM) Version 2.5 Readme, VocalTec Ltd., 02/1995, 5 pages.	
	30-7	Written Opinion issued February 12, 1998 in corresponding International Application Serial No. PCT/US96/15504.	

Examiner Signature		Date Considered	

*Examiner: Initial if reference was considered, whether or not citation is in conformance with MPEP 609. Draw a line through citation if not in conformance and not considered. Include a copy of this form with next communication to applicant. Notes: If identified, the following is provided: EA = English Abtract, T = Translation, PF = Patent Family.

(12) PATENT APPLICATION (11) Application No. AU 200059377 A1
(19) AUSTRALIAN PATENT OFFICE

(54) Title
Point-to-point Internet protocol

(51)[7] International Patent Classification(s)
H04L 029/00

(21) Application No: 200059377 (22) Application Date: 2000.09.13

(43) Publication Date : 2000.11.23
(43) Publication Journal Date : 2000.11.23

(62) Divisional of:
199672476

(71) Applicant(s)
NetSpeak Corporation

(72) Inventor(s)
Glenn W Hutton; Shane D Mattaway; Craig B Strickland

(74) Agent/Attorney
DAVIES COLLISON CAVE,1 Little Collins Street,MELBOURNE VIC 3000

Abstract

An apparatus for establishing a point-to-point communication link, the apparatus operating in a computer system operatively coupled to another computer system and a server over a computer network, the apparatus comprising means for transmitting an E-mail signal containing a network protocol address from a first process to a second process over the computer network, means for receiving a second network protocol address from the second process over the computer network, and means, responsive to the second network protocol address, for establishing a point-to-point communication link between the first process and the second process over the computer network.

13 09 00 59377

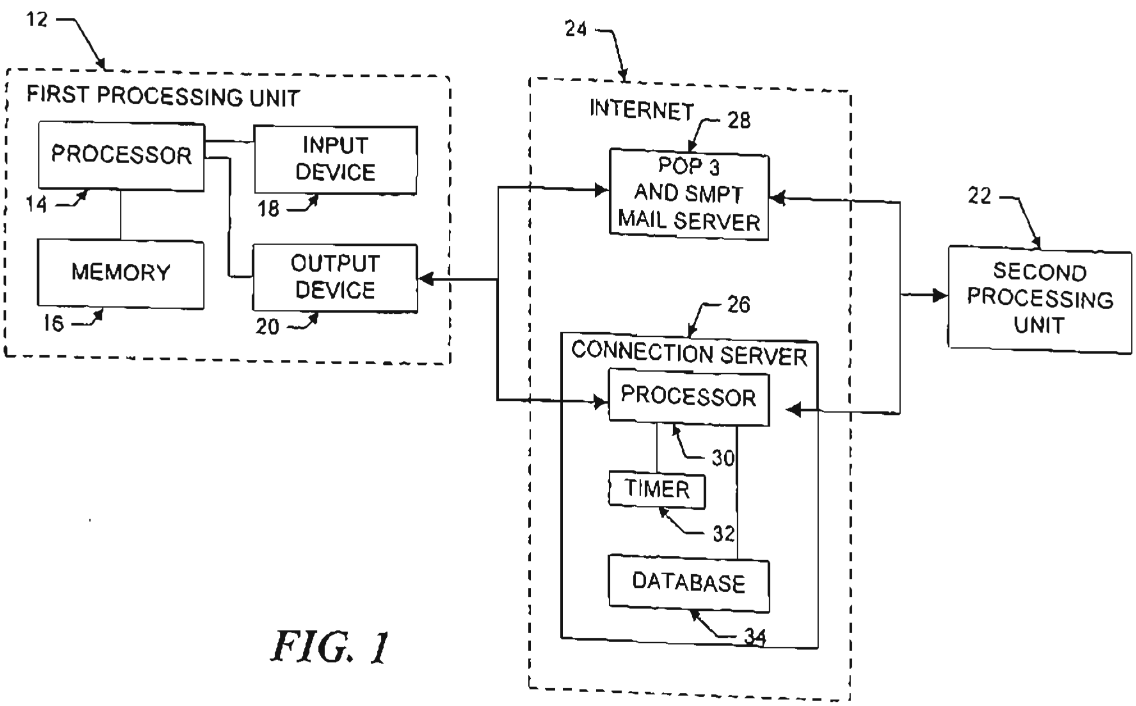


FIG. 1

AUSTRALIA

Patents Act 1990

COMPLETE SPECIFICATION

STANDARD PATENT

(ORIGINAL)

Name of Applicant: **NetSpeak Corporation**, of 902 Clint Moore Road, Suite 104, Boca Raton, Florida 33487, United States of America.

Actual Inventors: HUTTON, Glenn W
MATTAWAY, Shane D
STRICKLAND, Craig B

Address for Service: **DAVIES COLLISON CAVE**, Patent Attorneys, of 1 Little Collins Street, Melbourne, Victoria 3000, Australia.

Invention Title: **"Point-to-Point Internet Protocol"**

The following statement is a full description of this invention, including the best method of performing it known to us:

POINT-TO-POINT INTERNET PROTOCOL

The present invention relates, in general, to data processing systems, and more specifically, to a method and apparatus for facilitating audio communications over computer networks.

The increased popularity of on-line services such as AMERICA ONLINE™, COMPUSERVE®, and other services such as Internet gateways have spurred applications to provide multimedia, including video and voice clips, to online users. An example of an online voice clip application is VOICE E-MAIL FOR WINCIM and VOICE E-MAIL FOR AMERICA ONLINE™, available from Bonzi Software, as described in "Simple Utilities Send Voice E-Mail Online", MULTIMEDIA WORLD, VOL. 2, NO. 9, August 1995, p. 52. Using such Voice E-Mail software, a user may create an audio message to be sent to a predetermined E-mail address specified by the user.

Generally, devices interfacing to the Internet and other online services may communicate with each other upon establishing respective device addresses. One type of device address is the Internet Protocol (IP) address, which acts as a pointer to the device associated with the IP address. A typical device may have a Serial Line Internet Protocol or Point-to-Point Protocol (SLIP/PPP) account with a permanent IP address for receiving E-mail, voicemail, and the like over the Internet. E-mail and voicemail is generally intended to convey text, audio, etc., with any routing information such as an IP address and routing headers generally

being considered an artifact of the communication, or even gibberish to the recipient.

Devices such as a host computer or server of a company may include multiple modems for connection of users to the Internet, with a temporary IP address allocated to each user. For example, the host computer may have a general IP address "XXX.XXX.XXX," and each user may be allocated a successive IP address of XXX.XXX.XXX.10, XXX.XXX.XXX.11, XXX.XXX.XXX.12, etc. Such temporary IP addresses may be reassigned or recycled to the users, for example, as each user is successively connected to an outside party. For example, a host computer of a company may support a maximum of 254 IP addresses which are pooled and shared between devices connected to the host computer.

Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet. For example, realtime video teleconferencing has been implemented using dedicated IP addresses and mechanisms known as reflectors.

A technique for matching domain names to Internet Protocol addresses is described in the text entitled "Internetworking With TCP/IP", 2nd Edition, by Douglas E. Comer, November 1992, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A. Comer describes a domain name system and cooperative systems of name servers for matching domain names to network addresses. Each name server is a server program that supplies mapping of domain names to IP addresses. The system described in Comer, however, is not designed for use with network nodes whose network names or name to address bindings change frequently.

International Publication WO 92/19054 discloses a network

monitoring system including an address tracking module which uses passive monitoring of all packet communications over a local area network to maintain a name table of IP address mappings. The disclosed address tracking module is capable of monitoring only a small number of nodes on a local area network and is not suitable for use with a multitude of nodes over a wide area network.

None of the above-described systems are suitable for use with processes which have dynamically assigned network protocol addresses and which are communicating over wide area or global networks.

Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain.

In accordance with the present invention, there is provided an apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system operatively coupled to another computer system and a server over a computer network, said apparatus comprising:

a. means for transmitting an E-mail signal containing a network protocol address from a first process to a second process over the computer network;

b. means for receiving a second network protocol address from the second process over the computer network; and

c. means, responsive to the second network protocol address, for establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides a method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and a mail server process, said method comprising:

a. transmitting an E-mail signal to the mail server process over the computer network, the E-mail signal containing a first network protocol address assigned to the first process upon connection to the computer network;

b. receiving a second network protocol address from the second process over the computer network, the second network protocol address assigned to the second process upon connection to the computer network; and

c. establishing a point-to-point communication link between the first process and the second process over the computer network, in response to receiving the second network protocol address.

The present invention also provides an apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system operatively connectable to other processes and a server process over a computer network, said apparatus comprising:

a. program logic configured to transmit an E-mail signal containing a network protocol address from a first process to a second process over the computer network;

b. program logic configured to receive a second network protocol address from the second process over the computer network; and

c. program logic, responsive to the second network protocol address, and configured to establish a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides a computer program product for use with a computer system, the computer system capable of executing a first process and operatively connectable to a second process and a server over a computer network, the computer program product comprising a computer useable medium having program code embodied in the medium, the program code further comprising:

program code for transmitting an E-mail signal comprising a network protocol address of the first process to the second processor over the computer network;

program code for receiving a second network protocol address from the second process over the computer network; and

program code, responsive to the second network protocol address, for establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides a method of establishing a point-to-point communication between a first process and second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an E-mail server, said method comprising the steps of:

A. transmitting to the second process over the computer network an E-mail signal comprising a network protocol address of the first process;

B. receiving from the second process over the computer network a second network protocol address; and

C. in response to the second network protocol address, establishing a point-to-point communication link between the first process and the second process over the computer network.

Preferred embodiments of the present invention are hereinafter described, by way of example only, with reference to the following drawings, wherein:

FIG 1 illustrates, in block diagram format, a system for the disclosed point-to-point Internet protocol:

FIG 2 illustrates, in block diagram format, the system using a secondary point-to-point Internet protocol;

FIG 3 illustrates, in block diagram format; the system of FIGS 1-2 with the point-to-point Internet protocol established;

FIG 4 is another block diagram of the system of FIGS 1-2 with audio communications being conducted;

FIG. 5 illustrates a display screen for a processing unit;

FIG. 6 illustrates another display screen for a processing unit;

FIG. 7 illustrates a flowchart of the initiation of the point-to-point Internet protocols;

FIG. 8 illustrates a flowchart of the performance of the primary point-to-point Internet protocols; and

FIG. 9 illustrates a flowchart of the performance of the secondary point-to-point Internet protocol.

Referring now in specific detail to the drawings, with like reference numerals identifying similar or identical elements, as shown in FIG. 1, the present disclosure describes a point-to-point network protocol and system 10 for using such a protocol.

In an exemplary embodiment, the system 10 includes a first processing unit 12 for sending at least a voice signal from a first user to a second user. The first processing unit 12 includes a processor 14, a memory 16, an input device 18, and an output device 20. The output device 20 includes at least one modem capable of, for example, 14.4 kbaud communications and operatively connected via wired and/or wireless communication connections to the Internet or other computer networks such as an Intranet, i.e., a private computer network. One skilled in the art would understand that the input device 18 may be implemented at least in part by the modem of the output device 20 to allow input signals from the communication connections to be received. The second processing unit 22 may have a processor, memory, and input and output devices, including at least one modem and associated communication connections, as described above for the first processing unit 12. In an exemplary embodiment, each of the processing units 12, 22 may execute the WEBPHONE™ Internet telephony application available from NetSpeak Corporation, Boca Raton, FL, which is capable of performing the disclosed point-to-point Internet protocol and system 10, as described herein.

The first processing unit 12 and the second processing unit 22 are operatively connected to the Internet 24 by communication devices and software known in the art, such as an Internet Service Provider (ISP) or an Internet gateway. The processing units 12, 22 may be operatively interconnected through the Internet 24 to a connection server 26, and

may also be operatively connected to a mail server 28 associated with the Internet 24.

The connection server 26 includes a processor 30, a timer 32 for generating time stamps, and a memory such as a database 34 for storing, for example, E-mail and Internet Protocol (IP) addresses of logged-in units. In an exemplary embodiment, the connection server 26 may be a SPARC 5 server or a SPARC 20 server, available from SUN MICROSYSTEMS, INC., Mountain View, CA, having a central processing unit (CPU) as processor 30, an operating system (OS) such as UNIX, for providing timing operations such as maintaining the timer 32, a hard drive or fixed drive, as well as dynamic random access memory (DRAM) for storing the database 34, and a keyboard and display and/or other input and output devices (not shown in FIG. 1). The database 34 may be an SQL database available from ORACLE or INFORMIX.

In an exemplary embodiment, the mail server 28 may be a Post Office Protocol (POP) Version 3 mail server including a processor, memory, and stored programs operating in a UNIX environment, or, alternatively, another OS, to process E-mail capabilities between processing units and devices over the Internet 24.

The first processing unit 12 may operate the disclosed point-to-point Internet protocol by a computer program described hereinbelow in conjunction with FIG. 6, which may be implemented from compiled and /or interpreted source code in the C++ programming language and which may be downloaded to the first processing unit 12 from an external computer. The operating computer program may be stored in the memory 16, which may include about 8 MB RAM and/or a hard or fixed drive having about 8 MB. Alternatively, the source code may be implemented in the first processing unit 12 as firmware, as an erasable read only memory (EPROM), etc. It is understood that one skilled in the

art would be able to use programming languages other than C++ to implement the disclosed point-to-point network protocol and system 10.

The processor 14 receives input commands and data from a first user associated with the first processing unit 12 though the input device 18, which may be an input port connected by a wired, optical, or a wireless connection for electromagnetic transmissions, or alternatively may be transferable storage media, such as floppy disks, magnetic tapes, compact disks, or other storage media including the input data from the first user.

The input device 18 may include a user interface (not shown) having, for example, at least one button actuated by the user to input commands to select from a plurality of operating modes to operate the first processing unit 12. In alternative embodiments, the input device 18 may include a keyboard, a mouse, a touch screen, and/or a data reading device such as a disk drive for receiving the input data from input data files stored in storage media such as a floppy disk or, for example, an 8 mm storage tape. The input device 18 may alternatively include connections to other computer systems to receive the input commands and data therefrom.

The first processing unit 12 may include a visual interface for use in conjunction with the input device 18 and output device 20 similar to those screens illustrated in FIGS. 5-6, discussed below. It is also understood that alternative devices may be used to receive commands and data from the user, such as keyboards, mouse devices, and graphical user interfaces (GUI) such as WINDOWS™ 3.1 available form MICROSOFT Corporation, Redmond, WA., and other operating systems and GUIs, such as OS/2 and OS/2 WARP, available from IBM CORPORATION, Boca Raton, FL. Processing unit 12 may also include microphones and/or telephone handsets for receiving audio voice data

and commands, speech or voice recognition devices, dual tone multi-frequency (DTMF) based devices, and/or software known in the art to accept voice data and commands and to operate the first processing unit 12.

In addition, either of the first processing unit 12 and the second processing unit 22 may be implemented in a personal digital assistant (PDA) providing modem and E-mail capabilities and Internet access, with the PDA providing the input/output screens for mouse interactions or for touchscreen activation as shown, for example, in FIGS. 5-6, as a combination of the input device 18 and output device 20.

For clarity of explanation, the illustrative embodiment of the disclosed point-to-point Internet protocol and system 10 is presented as having individual functional blocks, which may include functional blocks labeled as "processor" and "processing unit". The functions represented by these blocks may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. For example, the functions of each of the processors and processing units presented herein may be provided by a shared processor or by a plurality of individual processors. Moreover, the use of the functional blocks with accompanying labels herein is not to be construed to refer exclusively to hardware capable of executing software. Illustrative embodiments may include digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, read-only memory (ROM) for storing software performing the operations discussed below, and random access memory (RAM) for storing DSP results. Very large scale integration (VLSI) hardware embodiments, as well as custom VLSI circuitry in combination with a general purpose DSP circuit, may also be provided. Any and all of these embodiments may be deemed to fail within the meaning of the labels for the functional blocks as used herein.

The processing units 12, 22 are capable of placing calls and connecting to other processing units connected to the Internet 24, for example, via dialup SLIP/PPP lines. In an exemplary embodiment, each processing unit assigns an unsigned long session number, for example, a 32- bit long sequence in a *.ini file for each call. Each call may be assigned a successive session number in sequence, which may be used by the respective processing unit to associate the call with one of the SLIP/PPP lines, to associate a <ConnectOK> response signal with a <Connect Request> signal, and to allow for multiplexing and demultiplexing of inbound and outbound conversations on conference lines, as explained hereinafter.

For callee (or called) processing units with fixed IP addresses, the caller (or calling) processing unit may open a "socket", i.e. a file handle or address indicating where data is to be sent, and transmit a <Call> command to establish communication with the callee utilizing, for example, datagram services such as Internet Standard network layering as well as transport layering, which may include a Transport Control Protocol (TCP) or a User Datagram Protocol (UDP) on top of the IP. Typically, a processing unit having a fixed IP address may maintain at least one open socket and a called processing unit waits for a <Call> command to assign the open socket to the incoming signal. If all lines are in use, the callee processing unit sends a BUSY signal or message to the callee processing unit. As shown in FIG. 1, the disclosed point-to-point Internet protocol and system 10 operate when a callee processing unit does not have a fixed or predetermined IP address. In the exemplary embodiment and without loss of generality, the first processing unit 12 is the caller processing unit and the second processing unit 22 is the called processing unit. When either of processing units 12, 22 logs on to the Internet via a dial-up connection, the respective unit is provided a

dynamically allocated IP address by the a connection service provider.

Upon the first user initiating the point-to-point Internet protocol when the first user is logged on to the Internet 24, the first processing unit 12 automatically transmits its associated E-mail address and its dynamically allocated IP address to the connection server 26. The connection server 26 then stores these addresses in the database 34 and time stamps the stored addresses using timer 32. The first user operating the first processing unit 12 is thus established in the database 34 as an active on-line party available for communication using the disclosed point-to-point Internet protocol. Similarly, a second user operating the second processing unit 22, upon connection to the Internet 24 through the a connection service provider, is processed by the connection server 26 to be established in the database 34 as an active on-line party.

The connection server 26 may use the time stamps to update the status of each processing unit; for example, after 2 hours, so that the on-line status information stored in the database 34 is relatively current. Other predetermined time periods, such as a default value of 24 hours, may be configured by a systems operator.

The first user with the first processing unit 12 initiates a call using, for example, a Send command and/or a command to speeddial an N^{TH} stored number, which may be labeled [SND] and [SPD] [N], respectively, by the input device 18 and/or the output device 20, such as shown in FIGS. 5-6. In response to either the Send or speeddial commands, the first processing unit 12 retrieves from memory 16 a stored E-mail address of the callee corresponding to the N^{TH} stored number. Alternatively, the first user may directly enter the E-mail address of the callee.

The first processing unit 12 then sends a query, including the E-mail address of the callee, to the connection server 26. The connection

server 26 then searches the database 34 to determine whether the callee is logged-in by finding any stored information corresponding to the callee's E-mail address indicating that the callee is active and on-line. If the callee is active and on-line, the connection server 26 then performs the primary point-to-point Internet protocol; i.e. the IP address of the callee is retrieved from the database 34 and sent to the first processing unit 12. The first processing unit 12 may then directly establish the point-to-point Internet communications with the callee using the IP address of the callee.

If the callee is not on-line when the connection server 26 determines the callee's status, the connection server 26 sends an OFF-LINE signal or message to the first processing unit 12. The first processing unit 12 may also display a message such as "Called Party Off-Line" to the first user.

When a user logs off or goes off-line from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user's information, or by flagging the user as being off-line. The connection server 26 may be instructed to update the user's information in the database 34 by an off-line message, such as a data packet, sent automatically from the processing unit of the user prior to being disconnected from the connection server 26. Accordingly, an off-line user is effectively disabled from making and/or receiving point-to-point Internet communications.

As shown in FIGS. 2-4, the disclosed secondary point-to-point Internet protocol may be used as an alternative to the primary point-to-point Internet protocol described above, for example, if the connection server 26 is non-responsive, inoperative, and/or unable to perform the primary point-to-point Internet protocol, as a non-responsive condition. Alternatively, the disclosed secondary point-to-point Internet protocol may

be used independent of the primary point-to-point Internet protocol. In the disclosed secondary point-to-point Internet protocol, the first processing unit 12 sends a <ConnectRequest> message via E-mail over the Internet 24 to the mail server 28. The E-mail including the <ConnectRequest> message may have, for example, the subject

[*wp#XXXXXXXX#nnn.nnn.nnn.#emailAddr]

where nnn.nnn.nnn.nnn. is the current (i.e. temporary or permanent) IP address of the first user, and XXXXXXXX is a session number, which may be unique and associated with the request of the first user to initiate point-to-point communication with the second user.

As described above, the first processing unit 12 may send the <ConnectRequest> message in response to an unsuccessful attempt to perform the primary point-to-point Internet protocol. Alternatively, the first processing unit 12 may send the <ConnectRequest> message in response to the first user initiating a SEND command or the like.

After the <ConnectRequest> message via E-mail is sent, the first processing unit 12 opens a socket and waits to detect a response from the second processing unit 22. A timeout timer, such as timer 32, may be set by the first processing unit 12, in a manner known in the art, to wait for a predetermined duration to receive a <ConnectOK> signal. The processor 14 of the first processing unit 12 may cause the output device 20 to output a Ring signal to the user, such as an audible ringing sound, about every 3 seconds . For example, the processor 14 may output a *.wav file, which may be labeled RING.WAV, which is processed by the output device 20 to output an audible ringing sound.

The mail server 28 then polls the second processing unit 22, for example, every 3-5 seconds, to deliver the E-mail. Generally, the second processing unit 22 checks the incoming lines, for example, at regular intervals to wait for and to detect incoming E-mail from the mail server 28

through the Internet 24.

Typically, for sending E-mail to users having associated processing units operatively connected to a host computer or server operating an Internet gateway, E-mail for a specific user may be sent over the Internet 24 and directed to the permanent IP address or the SLIP/PPP account designation of the host computer, which then assigns a temporary IP address to the processing unit of the specified user for properly routing the E-mail. The E-mail signal may include a name or other designation such as a user name which identifies the specific user regardless of the processing unit assigned to the user; that is, the host computer may track and store the specific device where a specific user is assigned or logged on, independent of the IP address system, and so the host computer may switch the E-mail signal to the device of the specific user. At that time, a temporary IP address may be generated or assigned to the specific user and device.

Upon detecting and/or receiving the incoming E-mail signal from the first processing unit 12, the second processing unit 22 may assign or may be assigned a temporary IP address. Therefore, the delivery of the E-mail through the Internet 24 provides the second processing unit 22 with a session number as well as IP addresses of both the first processing unit 12 and the second processing unit 22.

Point-to-point communication may then be established by the processing unit 22 processing the E-mail signal to extract the <ConnectRequest> message, including the IP address of the first processing unit 12 and the session number. The second processing unit 22 may then open a socket and generate a <ConnectOK> response signal, which includes the temporary IP address of the second processing unit 22 as well as the session number of the first processing unit.

The second processing unit 22 sends the <ConnectOK> signal

directly over the Internet 24 to the IP address of the first processing unit 12 without processing by the mail server 28, and a timeout timer of the second processing unit 22 may be set to wait and detect a <Call> signal expected from the first processing unit 12.

Realtime point-to-point communication of audio signals over the Internet 24, as well as video and voicemail, may thus be established and supported without requiring permanent IP addresses to be assigned to either of the users or processing units 12, 22. For the duration of the realtime point-to-point link, the relative permanence of the current IP addresses of the processing units 12, 22 is sufficient, whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).

In the exemplary embodiment, a first user operating the first processing unit 12 is not required to be notified by the first processing unit 12 that an E-mail is being generated and sent to establish the point-to-point link with the second user at the second processing unit 22. Similarly, the second user is not required to be notified by the second processing unit 22 that an E-mail has been received and/or a temporary IP address is associated with the second processing unit 22. The processing units 12, 22 may perform the disclosed point-to-point Internet protocol automatically upon initiation of the point-to-point communication command by the first user without displaying the E-mail interactions to either user. Accordingly, the disclosed point-to-point Internet protocol may be transparent to the users. Alternatively, either of the first and second users may receive, for example, a brief message of "CONNECTION IN PROGRESS" or the like on a display of the respective output device of the processing units 12, 22.

After the initiation of either the primary or the secondary point-to-

point Internet protocols described above in conjunction with FIGS. 1-2, the point-to-point communication link over the Internet 24 may be established as shown in FIGS. 3-4 in a manner known in the art. For example, referring to FIG. 3, upon receiving the <ConnectorOK> signal from the second processing unit 22, the first processing unit 12 extracts the IP address of the second processing unit 22 and the session number, and the session number sent from the second processing unit 22 is then checked with the session number originally sent from the first processing unit 12 in the <ConnectRequest> message as E-mail. If the session numbers sent and received by the processing unit 12 match, then the first processing unit 12 sends a <Call> signal directly over the Internet 24 to the second processing unit 22; i.e. using the IP address of the second processing unit 22 provided to the first processing unit 12 in the <ConnectOK> signal.

Upon receiving the <Call> signal, the second processing unit 22 may then begin a ring sequence, for example, by indicating or annunciating to the second user that an incoming call is being received. For example, the word "CALL" may be displayed on the output device of the second processing unit 22. The second user may then activate the second processing unit 22 to receive the incoming call.

Referring to FIG. 4, after the second processing unit 22 receives the incoming call, realtime audio and/or video conversations may be conducted in a manner known in the art between the first and second users through the Internet 24, for example, by compressed digital audio signals. Each of the processing units 12, 22 also display to each respective user the words "IN USE" to indicate that the point-to-point communication link is established and audio or video signals are being transmitted.

In addition, either user may terminate the point-to-point

communication link by, for example, activating a termination command, such as by activating an [END] button or icon on a respective processing unit, causing the respective processing unit to send an <End> signal which causes both processing units to terminate the respective sockets, as well as to perform other cleanup commands and functions known in the art.

FIGS. 5-6 illustrate examples of display screens 36 which may be output by a respective output device of each processing unit 12, 22 of FIGS. 1-4 for providing the disclosed point-to-point Internet protocol and system 10. Such display screens may be displayed on a display of a personal computer (PC) or a PDA in a manner known in the art.

As shown in FIG. 5, a first display screen 36 includes a status area 38 for indicating, for example, a called user by name and/or by IP address or telephone number; a current function such as C2; a current time; a current operating status such as "IN USE", and other control icons such as a down arrow icon 40 for scrolling down a list of parties on a current conference line. The operating status may include such annunciators as "IN USE," "IDLE," "BUSY," "NO ANSWER," "OFFLINE," "CALL," "DIALING," "MESSAGES," and "SPEEDDIAL."

Other areas of the display screen 36 may include activation areas or icons for actuating commands or entering data. For example, the display screen 36 may include a set of icons 42 arranged in columns and rows including digits 0-9 and commands such as END, SND, HLD, etc. For example, the END and SND commands may be initiated as described above, and the HLD icon 44 may be actuated to place a current line on hold. Such icons may also be configured to substantially simulate a telephone handset or a cellular telephone interface to facilitate ease of use, as well as to simulate function keys of a keyboard. For example, icons labeled L1-L4 may be mapped to function keys F1-F4 on standard

PC keyboards, and icons C1-C3 may be mapped to perform as combinations of function keys, such as CTRL-F1, CTRL-F2, and CTRL-F3, respectively. In addition, the icons labeled L1-L4 and C1-C3 may include circular regions which may simulate light emitting diodes (LEDs) which indicate that the function or element represented by the respective icon is active or being performed.

Icons L1-L4 may represent each of 4 lines available to the caller, and icons C1-C3 may represent conference calls using at least one line to connect, for example, two or more parties in a conference call. The icons L1-L4 and C1-C3 may indicate the activity of each respective line or conference line. For example, as illustrated in FIG. 5, icons L1-L2 may have lightly shaded or colored circles, such as a green circle, indicating that each of lines 1 and 2 are in use, while icons L3-L4 may have darkly shaded or color circles, such as a red or black circle, indicating that each of lines 3 and 4 are not in use. Similarly, the lightly shaded circle of the icon labeled C2 indicates that the function corresponding to C2 is active, as additionally indicated in the status are 38, while darkly shaded circles of icons labeled C1 and C3 indicate that such corresponding functions are not active.

The icons 42 are used in conjunction with the status area 38. For example, using a mouse for input, a line that is in use, as indicated by the lightly colored circle of the icon, may be activated to indicate a party's name by clicking a right mouse button for 5 seconds until another mouse click is actuated or the [ESC] key or icon is actuated. Thus, the user may switch between multiple calls in progress on respective lines.

Using the icons as well as an input device such as a mouse, a user may enter the name or alias or IP address, if known, of a party to be called by either manually entering the name, by using the speeddial feature, or by double clicking on an entry in a directory stored in the

memory, such as the memory 16 of the first processing unit 12, where the directory entries may be scrolled using the status area 38 and the down arrow icon 40.

Once a called party is listed in the status area 38 as being active on a line, the user may transfer the called party to another line or a conference line by clicking and dragging the status area 38, which is represented by a reduced icon 46. Dragging the reduced icon 46 to any one of line icons L1-L4 transfers the called party in use to the selected line, and dragging the reduced icon 46 to any one of conference line icons C1-C3 adds the called party to the selected conference call.

Other features may be supported, such as icons 48-52, where icon 48 corresponds to, for example, an ALT-X command to exit the communication facility of a processing unit, and icon 50 corresponds to, for example, an ALT-M command to minimize or maximize the display screen 36 by the output device of the processing unit. Icon 52 corresponds to an OPEN command, which may, for example, correspond to pressing the O key on a keyboard, to expand or contract the display screen 36 to represent the opening and closing of a cellular telephone. An "opened" configuration is shown in FIG. 5, and a "closed" configuration is shown in FIG. 6. In the "opened" configuration, additional features such as output volume (VOL) controls, input microphone (MIC) controls, waveform (WAV) sound controls, etc.

The use of display screens such as those shown in FIGS. 5-6 provided flexibility in implementing various features available to the user. It is to be understood that additional features such as those known in the art may be supported by the processing units 12, 22.

Alternatively, it is to be understood that one skilled in the art may implement the processing units 12, 22 to have the features of the display screens in FIGS. 5-6 in hardware; i.e. a wired telephone or wireless

cellular telephone may include various keys, LEDs, liquid crystal displays (LCDs), and touchscreen actuators corresponding to the icons and features shown in FIGS. 5-6. In addition, a PC may have the keys of a keyboard and mouse mapped to the icons and features shown in FIGS. 5-6.

Referring to FIG. 7, the disclosed point-to-point Internet protocol and system 10 is illustrated. First processing unit 12 initiates the point-to-point Internet protocol in step 56 by sending a query from the first processing unit 12 to the connection server 26. If connection server 26 is operative to perform the point-to-point Internet protocol, in step 58, first processing unit 12 receives an on-line status signal from the connection server 26, such signal may include the IP address of the callee or a "Callee Off-Line" message. Next, first processing unit 12 performs the primary point-to-point Internet protocol in step 60, which may include receiving, at the first processing unit 12, the IP address of the callee if the callee is active and on-line. Alternatively, processing unit 60 may initiate and perform the secondary point-to-point Internet protocol in step 62, if the called party is not active and/or on-line.

Referring to FIG. 8, in conjunction with FIGS. 1 and 3-4, the disclosed point-to-point Internet protocol and system 10 is illustrated. Connection server 26 starts the point-to-point Internet protocol, in step 64, and timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66. Connection server 26 receives a query from a first processing unit 12 in step 68 to determine whether a second user or second processing unit 22 is logged-in to the Internet 24, with the second user being specified, for example, by an E-mail address. Connection server 26 retrieves the IP address of the specified user from the database 34 in step 70, if the specified user is logged-in to the Internet, and sends the retrieved IP address to the first

processing unit 12 in step 72 to enable first processing unit 12 to establish point-to-point communications with the specified second user.

The disclosed secondary point-to-point Internet protocol operates as shown in FIG. 9. First processing unit 12 generates an E-mail signal, including a session number and a first IP address corresponding to a first processing unit in step 76. First processing unit 12 transmits the E-mail signal as a <ConnectRequest> signal to the Internet 24 in step 78. The E-mail signal is delivered through the Internet 24 using a mail server 28 to the second processing unit 22 in step 80. Second processing unit 22 extracts the session number and the first IP address from the E-mail signal in step 82 and transmits or sends the session number and a second IP address corresponding to the second processing unit 22, back to the first processing unit 12 through the Internet 24, in step 84. First processing unit 12 verifies the session number received from the second processing unit 22 in step 86, and establishes a point-to-point Internet communication link between the first processing unit 12 and second processing unit 22 using the first and second IP addresses in step 88.

While the disclosed point-to-point Internet protocols and system have been particularly shown and described with reference to the preferred embodiments, it is understood by those skilled in the art that various modifications in form and detail may be made therein without departing from the scope and spirit of the invention. Accordingly, modifications such as those suggested above, but not limited thereto, are to be considered within the scope of the invention.

Throughout this specification and the claims which follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

The reference to any prior art in this specification is not, and should not be taken as, an acknowledgment or any form of suggestion that that prior art forms part of the common general knowledge in Australia.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. Apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system operatively coupled to another computer system and a server over a computer network, said apparatus comprising:

a. means for transmitting an E-mail signal containing a network protocol address from a first process to a second process over the computer network;

b. means for receiving a second network protocol address from the second process over the computer network; and

c. means, responsive to the second network protocol address, for establishing a point-to-point communication link between the first process and the second process over the computer network.

2. A method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and a mail server process, said method comprising:

a. transmitting an E-mail signal to the mail server process over the computer network, the E-mail signal containing a first network protocol address assigned to the first process upon connection to the computer network;

b. receiving a second network protocol address from the second process over the computer network, the second network protocol address assigned to the second process upon connection to the computer network; and

c. establishing a point-to-point communication link between the first process and the second process over the computer network, in response to receiving the second network protocol address.

3. Apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system operatively connectable to other

processes and a server process over a computer network, said apparatus comprising:

a. program logic configured to transmit an E-mail signal containing a network protocol address from a first process to a second process over the computer network;

b. program logic configured to receive a second network protocol address from the second process over the computer network; and

c. program logic, responsive to the second network protocol address, and configured to establish a point-to-point communication link between the first process and the second process over the computer network.

4. A computer program product for use with a computer system, the computer system capable of executing a first process and operatively connectable to a second process and a server over a computer network, the computer program product comprising a computer useable medium having program code embodied in the medium, the program code further comprising:

program code for transmitting an E-mail signal comprising a network protocol address of the first process to the second processor over the computer network;

program code for receiving a second network protocol address from the second process over the computer network; and

program code, responsive to the second network protocol address, for establishing a point-to-point communication link between the first process and the second process over the computer network.

5. A method of establishing a point-to-point communication between a first process and second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an E-mail server, said method comprising the steps of:

A. transmitting to the second process over the computer network an E-mail signal comprising a network protocol address of the first process;

B. receiving from the second process over the computer network a second network protocol address; and

C. in response to the second network protocol address, establishing a point-to-point communication link between the first process and the second process over the computer network.

6. Apparatus substantially as hereinbefore described with reference to the accompanying drawings.

7. A method substantially as hereinbefore described with reference to the accompanying drawings.

8. A computer program product substantially as hereinbefore described with reference to the accompanying drawings.

DATED this 8th day of September 2000

NetSpeak Corporation
By its Patent Attorneys
DAVIES COLLISON CAVE

13 09 00 59377

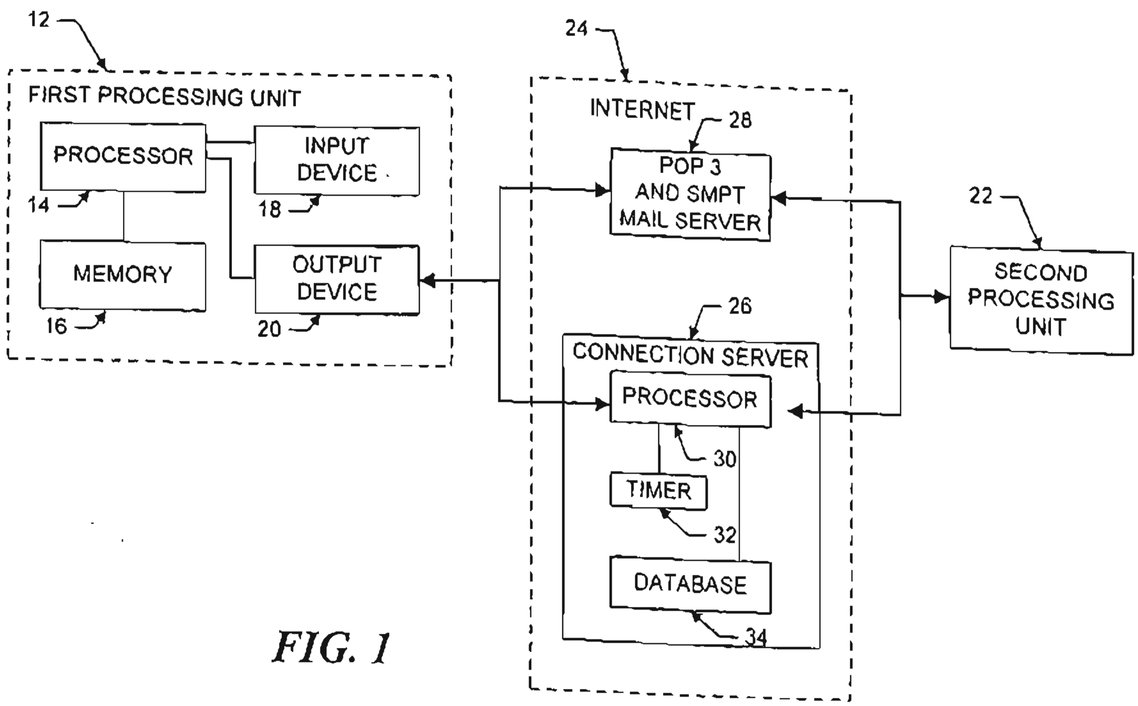


FIG. 1

13 09 00 59377

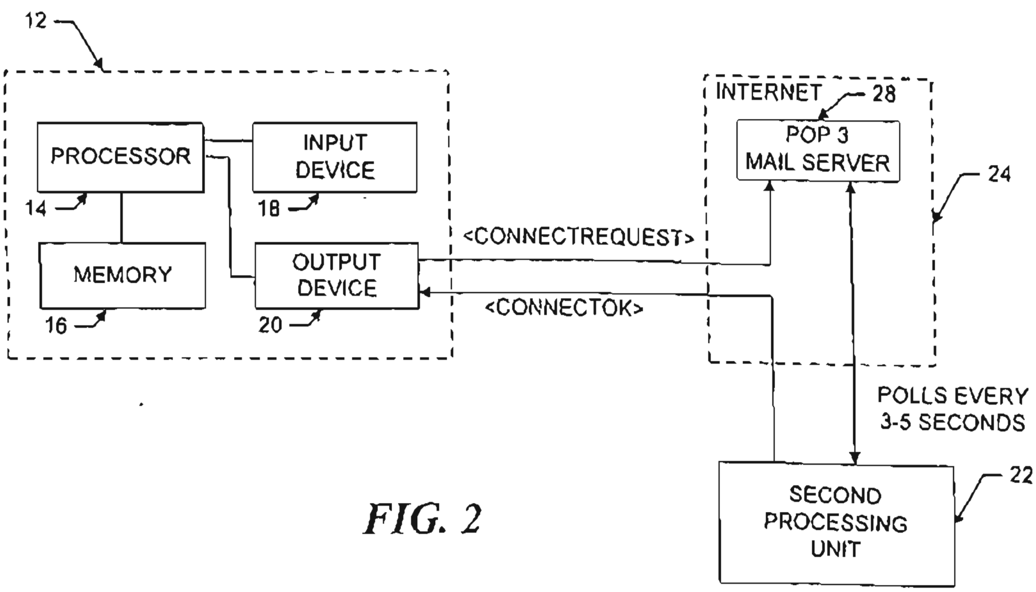


FIG. 2

13 09 00 59377

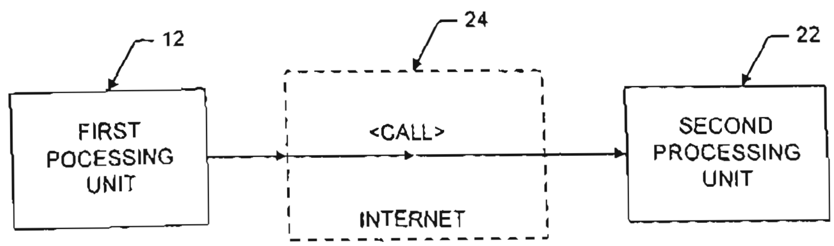


FIG. 3

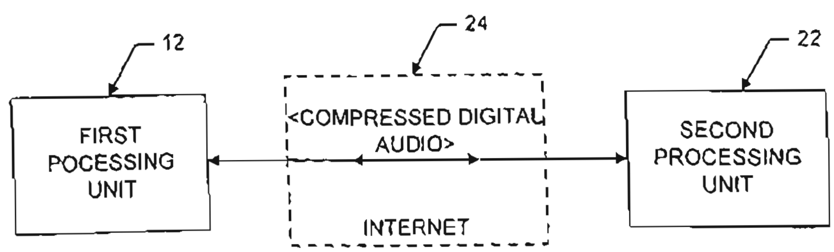


FIG. 4

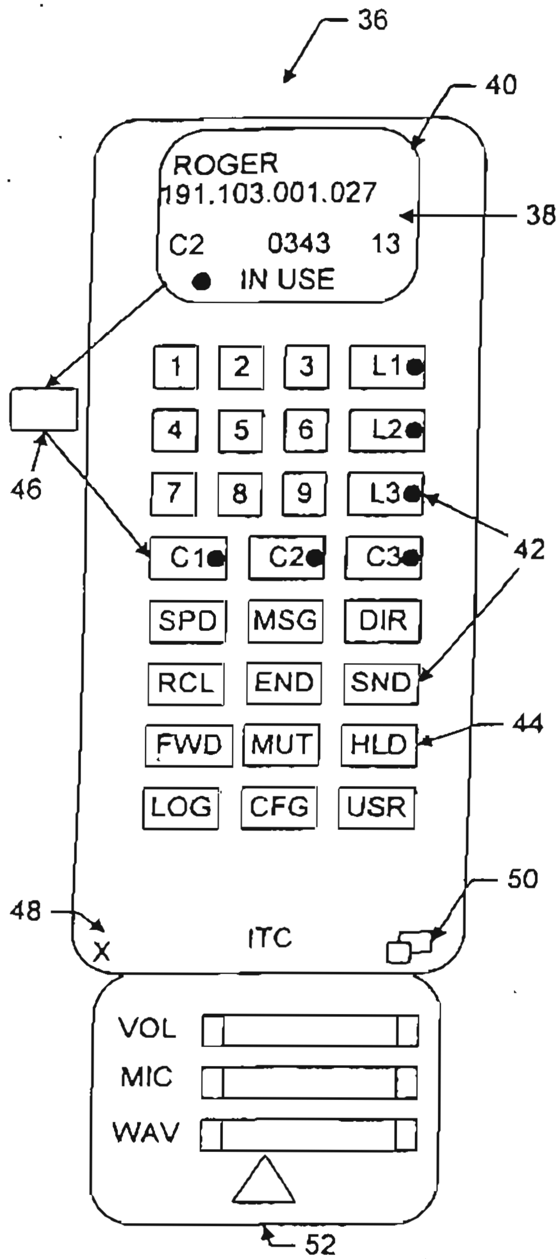

36
40
ROGER
191.103.001.027
38
C2 0343 13
IN USE
1 2 3 L1
4 5 6 L2
46
7 8 9 L3
C1 C2 C3
42
SPD MSG DIR
RCL END SND
FWD MUT HLD
44
LOG CFG USR
48
50
X
ITC
VOL
MIC
WAV
52

FIG. 5

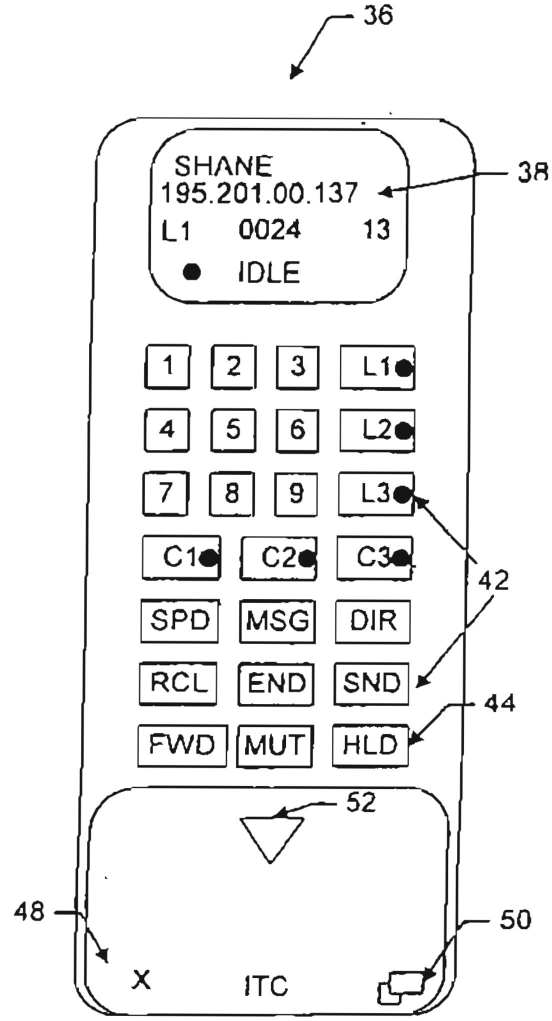

36
SHANE
195.201.00.137
38
L1 0024 13
IDLE
1 2 3 L1
4 5 6 L2
7 8 9 L3
C1 C2 C3
42
SPD MSG DIR
RCL END SND
44
FWD MUT HLD
52
48
50
X
ITC

FIG. 6

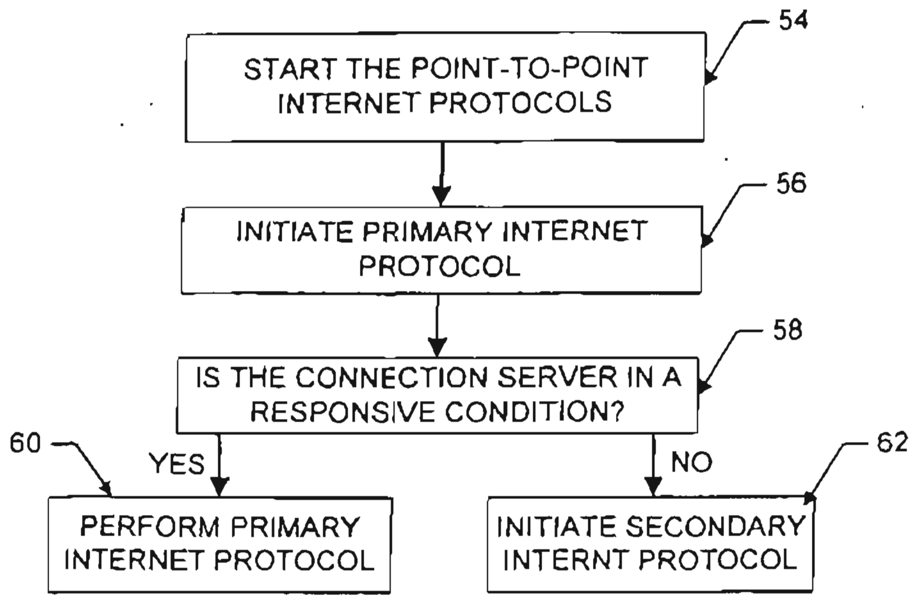

54
START THE POINT-TO-POINT INTERNET PROTOCOLS
56
INITIATE PRIMARY INTERNET PROTOCOL
58
IS THE CONNECTION SERVER IN A RESPONSIVE CONDITION?
60
YES
NO
62
PERFORM PRIMARY INTERNET PROTOCOL
INITIATE SECONDARY INTERNT PROTOCOL

FIG. 7

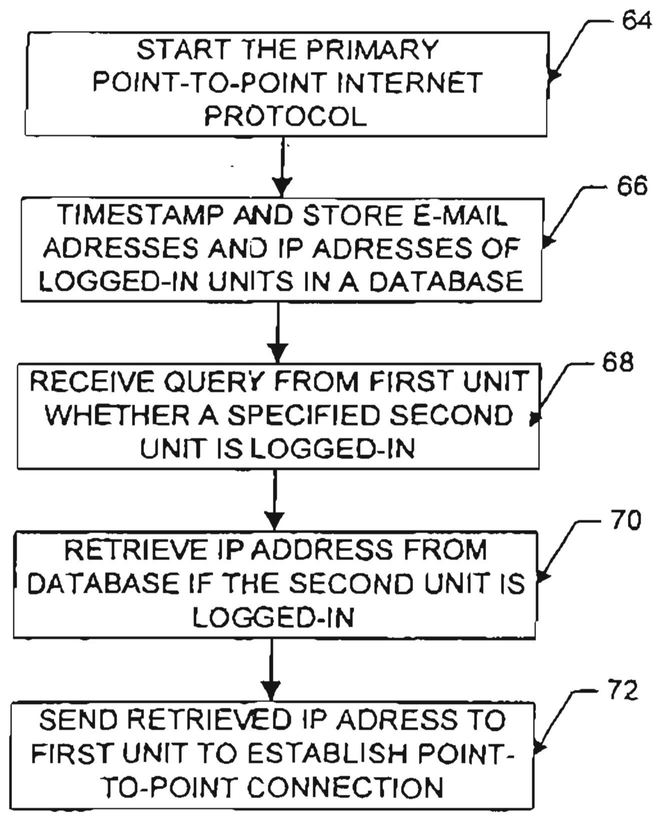

64
START THE PRIMARY POINT-TO-POINT INTERNET PROTOCOL
66
TIMESTAMP AND STORE E-MAIL ADRESSES AND IP ADRESSES OF LOGGED-IN UNITS IN A DATABASE
68
RECEIVE QUERY FROM FIRST UNIT WHETHER A SPECIFIED SECOND UNIT IS LOGGED-IN
70
RETRIEVE IP ADDRESS FROM DATABASE IF THE SECOND UNIT IS LOGGED-IN
72
SEND RETRIEVED IP ADRESS TO FIRST UNIT TO ESTABLISH POINT-TO-POINT CONNECTION

FIG. 8

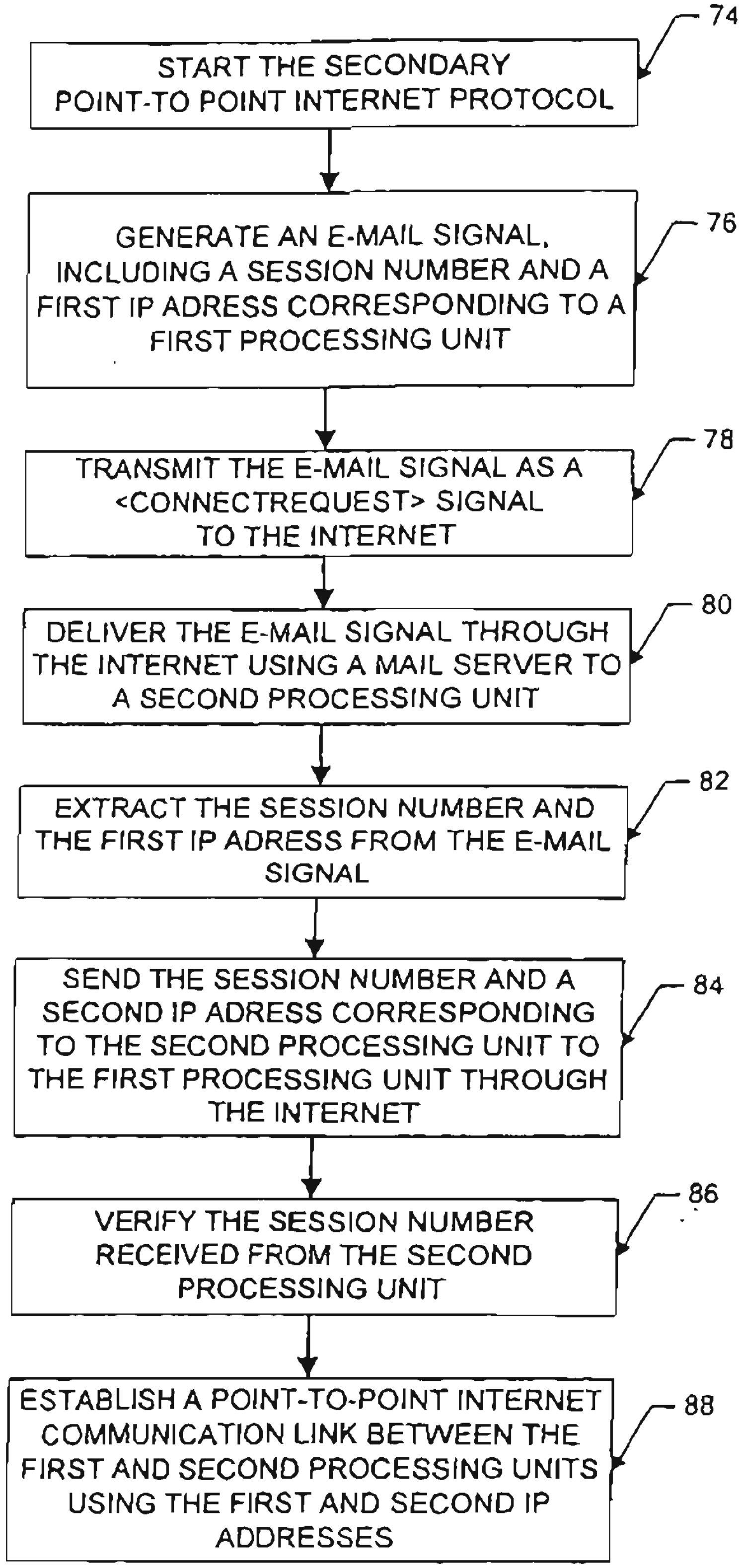

74
START THE SECONDARY POINT-TO POINT INTERNET PROTOCOL
76
GENERATE AN E-MAIL SIGNAL, INCLUDING A SESSION NUMBER AND A FIRST IP ADRESS CORRESPONDING TO A FIRST PROCESSING UNIT
78
TRANSMIT THE E-MAIL SIGNAL AS A <CONNECTREQUEST> SIGNAL TO THE INTERNET
80
DELIVER THE E-MAIL SIGNAL THROUGH THE INTERNET USING A MAIL SERVER TO A SECOND PROCESSING UNIT
82
EXTRACT THE SESSION NUMBER AND THE FIRST IP ADRESS FROM THE E-MAIL SIGNAL
84
SEND THE SESSION NUMBER AND A SECOND IP ADRESS CORRESPONDING TO THE SECOND PROCESSING UNIT TO THE FIRST PROCESSING UNIT THROUGH THE INTERNET
86
VERIFY THE SESSION NUMBER RECEIVED FROM THE SECOND PROCESSING UNIT
88
ESTABLISH A POINT-TO-POINT INTERNET COMMUNICATION LINK BETWEEN THE FIRST AND SECOND PROCESSING UNITS USING THE FIRST AND SECOND IP ADDRESSES

FIG. 9

(12) PATENT APPLICATION (11) Application No. AU 200059378 A1
(19) AUSTRALIAN PATENT OFFICE

(54) Title
Point-to-point internet protocol

(51)[7] International Patent Classification(s)
H04L 029/00

(21) Application No: 200059378 (22) Application Date: 2000.09.13

(43) Publication Date : 2000.11.30
(43) Publication Journal Date : 2000.11.30

(62) Divisional of:
199672476

(71) Applicant(s)
NetSpeak Corporation

(72) Inventor(s)
Glenn W Hutton; Shane D Mattaway; Craig B Strickland

(74) Agent/Attorney
DAVIES COLLISON CAVE,1 Little Collins Street,MELBOURNE VIC 3000

Abstract

A method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, said method for use in a computer system, the method comprising, providing a user interface element representing a first communication line, providing a user interface element representing a first callee process, and establishing a point-to-point communication link from the caller process to the first callee process, in response to a user associating the element representing the first callee process with the element representing the first communication line.

9. The method of claim 7 wherein the element provided in step D represents a communication line on mute status.

10. The method of claim 1 wherein the caller process further comprises a visual display and the user interface comprises a graphic user interface.

11. The method of claim 10 wherein the steps of establishing a point-to-point link as described in step C is performed in response to manipulation of the graphic elements on the graphic user interface.

12. A computer program product for use with a computer system comprising:

a computer useable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the medium further comprising:

program code for generating an element representing a first communication line;

program code for generating an element representing a first callee process;

program code, responsive to a user associating the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process.

13. The computer program product of claim 12 wherein the program code for establishing a point-to-point communication link further comprises:

program code for querying the server as to the on-line status of the first callee process; and

program code for receiving a network protocol address of the first callee process over the computer network from the server.

13 09 00 59378

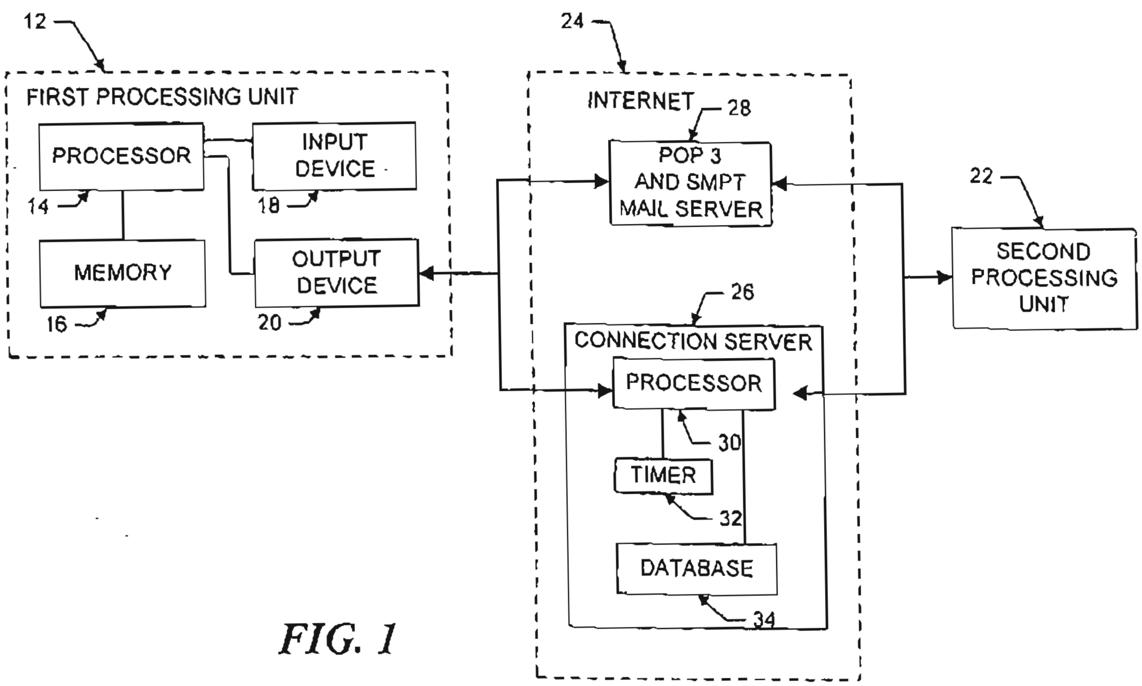


FIG. 1

P/00/011 28/5/91
Regulation 3.2

AUSTRALIA

Patents Act 1990

COMPLETE SPECIFICATION

STANDARD PATENT

(ORIGINAL)

Name of Applicant: **NetSpeak Corporation**, of 902 Clint Moore Road, Suite 104, Boca Raton, Florida 33487, United States of America.

Actual Inventors: HUTTON, Glenn W
MATTAWAY, Shane D
STRICKLAND, Craig B

Address for Service: **DAVIES COLLISON CAVE**, Patent Attorneys, of 1 Little Collins Street, Melbourne, Victoria 3000, Australia.

Invention Title: **"Point-to-Point Internet Protocol"**

The following statement is a full description of this invention, including the best method of performing it known to us:

POINT-TO-POINT INTERNET PROTOCOL

The present invention relates, in general, to data processing systems, and more specifically, to a method and apparatus for facilitating audio communications over computer networks.

The increased popularity of on-line services such as AMERICA ONLINE™, COMPUSERVE®, and other services such as Internet gateways have spurred applications to provide multimedia, including video and voice clips, to online users. An example of an online voice clip application is VOICE E-MAIL FOR WINCIM and VOICE E-MAIL FOR AMERICA ONLINE™, available from Bonzi Software, as described in "Simple Utilities Send Voice E-Mail Online", MULTIMEDIA WORLD, VOL. 2, NO. 9, August 1995, p. 52. Using such Voice E-Mail software, a user may create an audio message to be sent to a predetermined E-mail address specified by the user.

Generally, devices interfacing to the Internet and other online services may communicate with each other upon establishing respective device addresses. One type of device address is the Internet Protocol (IP) address, which acts as a pointer to the device associated with the IP address. A typical device may have a Serial Line Internet Protocol or Point-to-Point Protocol (SLIP/PPP) account with a permanent IP address for receiving E-mail, voicemail, and the like over the Internet. E-mail and voicemail is generally intended to convey text, audio, etc., with any routing information such as an IP address and routing headers generally

being considered an artifact of the communication, or even gibberish to the recipient.

Devices such as a host computer or server of a company may include multiple modems for connection of users to the Internet, with a temporary IP address allocated to each user. For example, the host computer may have a general IP address "XXX.XXX.XXX," and each user may be allocated a successive IP address of XXX.XXX.XXX.10, XXX.XXX.XXX.11, XXX.XXX.XXX.12, etc. Such temporary IP addresses may be reassigned or recycled to the users, for example, as each user is successively connected to an outside party. For example, a host computer of a company may support a maximum of 254 IP addresses which are pooled and shared between devices connected to the host computer.

Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet. For example, realtime video teleconferencing has been implemented using dedicated IP addresses and mechanisms known as reflectors.

A technique for matching domain names to Internet Protocol addresses is described in the text entitled "Internetworking With TCP/IP", 2nd Edition, by Douglas E. Comer, November 1992, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A. Comer describes a domain name system and cooperative systems of name servers for matching domain names to network addresses. Each name server is a server program that supplies mapping of domain names to IP addresses. The system described in Comer, however, is not designed for use with network nodes whose network names or name to address bindings change frequently.

International Publication WO 92/19054 discloses a network

monitoring system including an address tracking module which uses passive monitoring of all packet communications over a local area network to maintain a name table of IP address mappings. The disclosed address tracking module is capable of monitoring only a small number of nodes on a local area network and is not suitable for use with a multitude of nodes over a wide area network.

None of the above-described systems are suitable for use with processes which have dynamically assigned network protocol addresses and which are communicating over wide area or global networks.

Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain.

In accordance with the present invention, there is provided a method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, said method for use in a computer system, said method comprising:

A. providing a user interface element representing a first communication line;

B. providing a user interface element representing a first callee process; and

C. establishing a point-to-point communication link from the caller process to the first callee process, in response to a user associating the element representing the first callee process with the element representing the first communication line.

The present invention also provides a computer program product for use with a computer system comprising:

a computer useable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the medium further comprising:

program code for generating an element representing a first communication line;

program code for generating an element representing a first callee process; program code, responsive to a user associating the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process.

The present invention also provides a computer data signal embodied in a carrier wave comprising:

program code for generating a element representing a first communication line;

program code for generating an element representing a first callee process;

program code, responsive to association of the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process over a computer network.

Preferred embodiments of the present invention are hereinafter described, by way of example only, with reference to the following drawings, wherein:

FIG 1 illustrates, in block diagram format, a system for the disclosed point-to-point Internet protocol:

FIG 2 illustrates, in block diagram format, the system using a secondary point-to-point Internet protocol;

FIG 3 illustrates, in block diagram format; the system of FIGS 1-2 with the point-to-point Internet protocol established;

FIG 4 is another block diagram of the system of FIGS 1-2 with audio communications being conducted;

FIG. 5 illustrates a display screen for a processing unit;

FIG. 6 illustrates another display screen for a processing unit;

FIG. 7 illustrates a flowchart of the initiation of the point-to-point Internet protocols;

FIG. 8 illustrates a flowchart of the performance of the primary point-to-point Internet protocols; and

FIG. 9 illustrates a flowchart of the performance of the secondary point-to-point Internet protocol.

Referring now in specific detail to the drawings, with like reference numerals identifying similar or identical elements, as shown in FIG. 1, the present disclosure describes a point-to-point network protocol and system 10 for using such a protocol.

In an exemplary embodiment, the system 10 includes a first processing unit 12 for sending at least a voice signal from a first user to a second user. The first processing unit 12 includes a processor 14, a memory 16, an input device 18, and an output device 20. The output device 20 includes at least one modem capable of, for example, 14.4 kbaud communications and operatively connected via wired and/or wireless communication connections to the Internet or other computer networks such as an Intranet, i.e., a private computer network. One skilled in the art would understand that the input device 18 may be implemented at least in part by the modem of the output device 20 to allow input signals from the communication connections to be received. The second processing unit 22 may have a processor, memory, and input and output devices, including at least one modem and associated communication connections, as described above for the first processing unit 12. In an exemplary embodiment, each of the processing units 12, 22 may execute the WEBPHONE™ Internet telephony application available from NetSpeak Corporation, Boca Raton, FL, which is capable of performing the disclosed point-to-point Internet protocol and system 10, as described herein.

The first processing unit 12 and the second processing unit 22 are operatively connected to the Internet 24 by communication devices and software known in the art, such as an Internet Service Provider (ISP) or an Internet gateway. The processing units 12, 22 may be operatively interconnected through the Internet 24 to a connection server 26, and

may also be operatively connected to a mail server 28 associated with the Internet 24.

The connection server 26 includes a processor 30, a timer 32 for generating time stamps, and a memory such as a database 34 for storing, for example, E-mail and Internet Protocol (IP) addresses of logged-in units. In an exemplary embodiment, the connection server 26 may be a SPARC 5 server or a SPARC 20 server, available from SUN MICROSYSTEMS, INC., Mountain View, CA, having a central processing unit (CPU) as processor 30, an operating system (OS) such as UNIX, for providing timing operations such as maintaining the timer 32, a hard drive or fixed drive, as well as dynamic random access memory (DRAM) for storing the database 34, and a keyboard and display and/or other input and output devices (not shown in FIG. 1). The database 34 may be an SQL database available from ORACLE or INFORMIX.

In an exemplary embodiment, the mail server 28 may be a Post Office Protocol (POP) Version 3 mail server including a processor, memory, and stored programs operating in a UNIX environment, or, alternatively, another OS, to process E-mail capabilities between processing units and devices over the Internet 24.

The first processing unit 12 may operate the disclosed point-to-point Internet protocol by a computer program described hereinbelow in conjunction with FIG. 6, which may be implemented from compiled and /or interpreted source code in the C++ programming language and which may be downloaded to the first processing unit 12 from an external computer. The operating computer program may be stored in the memory 16, which may include about 8 MB RAM and/or a hard or fixed drive having about 8 MB. Alternatively, the source code may be implemented in the first processing unit 12 as firmware, as an erasable read only memory (EPROM), etc. It is understood that one skilled in the

art would be able to use programming languages other than C++ to implement the disclosed point-to-point network protocol and system 10.

The processor 14 receives input commands and data from a first user associated with the first processing unit 12 though the input device 18, which may be an input port connected by a wired, optical, or a wireless connection for electromagnetic transmissions, or alternatively may be transferable storage media, such as floppy disks, magnetic tapes, compact disks, or other storage media including the input data from the first user.

The input device 18 may include a user interface (not shown) having, for example, at least one button actuated by the user to input commands to select from a plurality of operating modes to operate the first processing unit 12. In alternative embodiments, the input device 18 may include a keyboard, a mouse, a touch screen, and/or a data reading device such as a disk drive for receiving the input data from input data files stored in storage media such as a floppy disk or, for example, an 8 mm storage tape. The input device 18 may alternatively include connections to other computer systems to receive the input commands and data therefrom.

The first processing unit 12 may include a visual interface for use in conjunction with the input device 18 and output device 20 similar to those screens illustrated in FIGS. 5-6, discussed below. It is also understood that alternative devices may be used to receive commands and data from the user, such as keyboards, mouse devices, and graphical user interfaces (GUI) such as WINDOWS™ 3.1 available form MICROSOFT Corporation, Redmond, WA., and other operating systems and GUIs, such as OS/2 and OS/2 WARP, available from IBM CORPORATION, Boca Raton, FL. Processing unit 12 may also include microphones and/or telephone handsets for receiving audio voice data

and commands, speech or voice recognition devices, dual tone multi-frequency (DTMF) based devices, and/or software known in the art to accept voice data and commands and to operate the first processing unit 12.

In addition, either of the first processing unit 12 and the second processing unit 22 may be implemented in a personal digital assistant (PDA) providing modem and E-mail capabilities and Internet access, with the PDA providing the input/output screens for mouse interactions or for touchscreen activation as shown, for example, in FIGS. 5-6, as a combination of the input device 18 and output device 20.

For clarity of explanation, the illustrative embodiment of the disclosed point-to-point Internet protocol and system 10 is presented as having individual functional blocks, which may include functional blocks labeled as "processor" and "processing unit". The functions represented by these blocks may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. For example, the functions of each of the processors and processing units presented herein may be provided by a shared processor or by a plurality of individual processors. Moreover, the use of the functional blocks with accompanying labels herein is not to be construed to refer exclusively to hardware capable of executing software. Illustrative embodiments may include digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, read-only memory (ROM) for storing software performing the operations discussed below, and random access memory (RAM) for storing DSP results. Very large scale integration (VLSI) hardware embodiments, as well as custom VLSI circuitry in combination with a general purpose DSP circuit, may also be provided. Any and all of these embodiments may be deemed to fall within the meaning of the labels for the functional blocks as used herein.

The processing units 12, 22 are capable of placing calls and connecting to other processing units connected to the Internet 24, for example, via dialup SLIP/PPP lines. In an exemplary embodiment, each processing unit assigns an unsigned long session number, for example, a 32- bit long sequence in a *.ini file for each call. Each call may be assigned a successive session number in sequence, which may be used by the respective processing unit to associate the call with one of the SLIP/PPP lines, to associate a <ConnectOK> response signal with a <Connect Request> signal, and to allow for multiplexing and demultiplexing of inbound and outbound conversations on conference lines, as explained hereinafter.

For callee (or called) processing units with fixed IP addresses, the caller (or calling) processing unit may open a "socket", i.e. a file handle or address indicating where data is to be sent, and transmit a <Call> command to establish communication with the callee utilizing, for example, datagram services such as Internet Standard network layering as well as transport layering, which may include a Transport Control Protocol (TCP) or a User Datagram Protocol (UDP) on top of the IP. Typically, a processing unit having a fixed IP address may maintain at least one open socket and a called processing unit waits for a <Call> command to assign the open socket to the incoming signal. If all lines are in use, the callee processing unit sends a BUSY signal or message to the callee processing unit. As shown in FIG. 1, the disclosed point-to-point Internet protocol and system 10 operate when a callee processing unit does not have a fixed or predetermined IP address. In the exemplary embodiment and without loss of generality, the first processing unit 12 is the caller processing unit and the second processing unit 22 is the called processing unit. When either of processing units 12, 22 logs on to the Internet via a dial-up connection, the respective unit is provided a

dynamically allocated IP address by the a connection service provider.

Upon the first user initiating the point-to-point Internet protocol when the first user is logged on to the Internet 24, the first processing unit 12 automatically transmits its associated E-mail address and its dynamically allocated IP address to the connection server 26. The connection server 26 then stores these addresses in the database 34 and time stamps the stored addresses using timer 32. The first user operating the first processing unit 12 is thus established in the database 34 as an active on-line party available for communication using the disclosed point-to-point Internet protocol. Similarly, a second user operating the second processing unit 22, upon connection to the Internet 24 through the a connection service provider, is processed by the connection server 26 to be established in the database 34 as an active on-line party.

The connection server 26 may use the time stamps to update the status of each processing unit; for example, after 2 hours, so that the on-line status information stored in the database 34 is relatively current. Other predetermined time periods, such as a default value of 24 hours, may be configured by a systems operator.

The first user with the first processing unit 12 initiates a call using, for example, a Send command and/or a command to speeddial an N^{TH} stored number, which may be labeled [SND] and [SPD] [N], respectively, by the input device 18 and/or the output device 20, such as shown in FIGS. 5-6. In response to either the Send or speeddial commands, the first processing unit 12 retrieves from memory 16 a stored E-mail address of the callee corresponding to the N^{TH} stored number. Alternatively, the first user may directly enter the E-mail address of the callee.

The first processing unit 12 then sends a query, including the E-mail address of the callee, to the connection server 26. The connection

server 26 then searches the database 34 to determine whether the callee is logged-in by finding any stored information corresponding to the callee's E-mail address indicating that the callee is active and on-line. If the callee is active and on-line, the connection server 26 then performs the primary point-to-point Internet protocol; i.e. the IP address of the callee is retrieved from the database 34 and sent to the first processing unit 12. The first processing unit 12 may then directly establish the point-to-point Internet communications with the callee using the IP address of the callee.

If the callee is not on-line when the connection server 26 determines the callee's status, the connection server 26 sends an OFF-LINE signal or message to the first processing unit 12. The first processing unit 12 may also display a message such as "Called Party Off-Line" to the first user.

When a user logs off or goes off-line from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user's information, or by flagging the user as being off-line. The connection server 26 may be instructed to update the user's information in the database 34 by an off-line message, such as a data packet, sent automatically from the processing unit of the user prior to being disconnected from the connection server 26. Accordingly, an off-line user is effectively disabled from making and/or receiving point-to-point Internet communications.

As shown in FIGS. 2-4, the disclosed secondary point-to-point Internet protocol may be used as an alternative to the primary point-to-point Internet protocol described above, for example, if the connection server 26 is non-responsive, inoperative, and/or unable to perform the primary point-to-point Internet protocol, as a non-responsive condition. Alternatively, the disclosed secondary point-to-point Internet protocol may

be used independent of the primary point-to-point Internet protocol. In the disclosed secondary point-to-point Internet protocol, the first processing unit 12 sends a <ConnectRequest> message via E-mail over the Internet 24 to the mail server 28. The E-mail including the <ConnectRequest> message may have, for example, the subject

[*wp#XXXXXXXX#nnn.nnn.nnn.#emailAddr]

where nnn.nnn.nnn.nnn. is the current (i.e. temporary or permanent) IP address of the first user, and XXXXXXXX is a session number, which may be unique and associated with the request of the first user to initiate point-to-point communication with the second user.

As described above, the first processing unit 12 may send the <ConnectRequest> message in response to an unsuccessful attempt to perform the primary point-to-point Internet protocol. Alternatively, the first processing unit 12 may send the <ConnectRequest> message in response to the first user initiating a SEND command or the like.

After the <ConnectRequest> message via E-mail is sent, the first processing unit 12 opens a socket and waits to detect a response from the second processing unit 22. A timeout timer, such as timer 32, may be set by the first processing unit 12, in a manner known in the art, to wait for a predetermined duration to receive a <ConnectOK> signal. The processor 14 of the first processing unit 12 may cause the output device 20 to output a Ring signal to the user, such as an audible ringing sound, about every 3 seconds . For example, the processor 14 may output a *.wav file, which may be labeled RING.WAV, which is processed by the output device 20 to output an audible ringing sound.

The mail server 28 then polls the second processing unit 22, for example, every 3-5 seconds, to deliver the E-mail. Generally, the second processing unit 22 checks the incoming lines, for example, at regular intervals to wait for and to detect incoming E-mail from the mail server 28

through the Internet 24.

Typically, for sending E-mail to users having associated processing units operatively connected to a host computer or server operating an Internet gateway, E-mail for a specific user may be sent over the Internet 24 and directed to the permanent IP address or the SLIP/PPP account designation of the host computer, which then assigns a temporary IP address to the processing unit of the specified user for properly routing the E-mail. The E-mail signal may include a name or other designation such as a user name which identifies the specific user regardless of the processing unit assigned to the user; that is, the host computer may track and store the specific device where a specific user is assigned or logged on, independent of the IP address system, and so the host computer may switch the E-mail signal to the device of the specific user. At that time, a temporary IP address may be generated or assigned to the specific user and device.

Upon detecting and/or receiving the incoming E-mail signal from the first processing unit 12, the second processing unit 22 may assign or may be assigned a temporary IP address. Therefore, the delivery of the E-mail through the Internet 24 provides the second processing unit 22 with a session number as well as IP addresses of both the first processing unit 12 and the second processing unit 22.

Point-to-point communication may then be established by the processing unit 22 processing the E-mail signal to extract the <ConnectRequest> message, including the IP address of the first processing unit 12 and the session number. The second processing unit 22 may then open a socket and generate a <ConnectOK> response signal, which includes the temporary IP address of the second processing unit 22 as well as the session number of the first processing unit.

The second processing unit 22 sends the <ConnectOK> signal

directly over the Internet 24 to the IP address of the first processing unit 12 without processing by the mail server 28, and a timeout timer of the second processing unit 22 may be set to wait and detect a <Call> signal expected from the first processing unit 12.

Realtime point-to-point communication of audio signals over the Internet 24, as well as video and voicemail, may thus be established and supported without requiring permanent IP addresses to be assigned to either of the users or processing units 12, 22. For the duration of the realtime point-to-point link, the relative permanence of the current IP addresses of the processing units 12, 22 is sufficient, whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).

In the exemplary embodiment, a first user operating the first processing unit 12 is not required to be notified by the first processing unit 12 that an E-mail is being generated and sent to establish the point-to-point link with the second user at the second processing unit 22. Similarly, the second user is not required to be notified by the second processing unit 22 that an E-mail has been received and/or a temporary IP address is associated with the second processing unit 22. The processing units 12, 22 may perform the disclosed point-to-point Internet protocol automatically upon initiation of the point-to-point communication command by the first user without displaying the E-mail interactions to either user. Accordingly, the disclosed point-to-point Internet protocol may be transparent to the users. Alternatively, either of the first and second users may receive, for example, a brief message of "CONNECTION IN PROGRESS" or the like on a display of the respective output device of the processing units 12, 22.

After the initiation of either the primary or the secondary point-to-

point Internet protocols described above in conjunction with FIGS. 1-2, the point-to-point communication link over the Internet 24 may be established as shown in FIGS. 3-4 in a manner known in the art. For example, referring to FIG. 3, upon receiving the <ConnectorOK> signal from the second processing unit 22, the first processing unit 12 extracts the IP address of the second processing unit 22 and the session number, and the session number sent from the second processing unit 22 is then checked with the session number originally sent from the first processing unit 12 in the <ConnectRequest> message as E-mail. If the session numbers sent and received by the processing unit 12 match, then the first processing unit 12 sends a <Call> signal directly over the Internet 24 to the second processing unit 22; i.e. using the IP address of the second processing unit 22 provided to the first processing unit 12 in the <ConnectOK> signal.

Upon receiving the <Call> signal, the second processing unit 22 may then begin a ring sequence, for example, by indicating or annunciating to the second user that an incoming call is being received. For example, the word "CALL" may be displayed on the output device of the second processing unit 22. The second user may then activate the second processing unit 22 to receive the incoming call.

Referring to FIG. 4, after the second processing unit 22 receives the incoming call, realtime audio and/or video conversations may be conducted in a manner known in the art between the first and second users through the Internet 24, for example, by compressed digital audio signals. Each of the processing units 12, 22 also display to each respective user the words "IN USE" to indicate that the point-to-point communication link is established and audio or video signals are being transmitted.

In addition, either user may terminate the point-to-point

communication link by, for example, activating a termination command, such as by activating an [END] button or icon on a respective processing unit, causing the respective processing unit to send an <End> signal which causes both processing units to terminate the respective sockets, as well as to perform other cleanup commands and functions known in the art.

FIGS. 5-6 illustrate examples of display screens 36 which may be output by a respective output device of each processing unit 12, 22 of FIGS. 1-4 for providing the disclosed point-to-point Internet protocol and system 10. Such display screens may be displayed on a display of a personal computer (PC) or a PDA in a manner known in the art.

As shown in FIG. 5, a first display screen 36 includes a status area 38 for indicating, for example, a called user by name and/or by IP address or telephone number; a current function such as C2; a current time; a current operating status such as "IN USE", and other control icons such as a down arrow icon 40 for scrolling down a list of parties on a current conference line. The operating status may include such annunciators as "IN USE," "IDLE," "BUSY," "NO ANSWER," "OFFLINE," "CALL," "DIALING," "MESSAGES," and "SPEEDDIAL."

Other areas of the display screen 36 may include activation areas or icons for actuating commands or entering data. For example, the display screen 36 may include a set of icons 42 arranged in columns and rows including digits 0-9 and commands such as END, SND, HLD, etc. For example, the END and SND commands may be initiated as described above, and the HLD icon 44 may be actuated to place a current line on hold. Such icons may also be configured to substantially simulate a telephone handset or a cellular telephone interface to facilitate ease of use, as well as to simulate function keys of a keyboard. For example, icons labeled L1-L4 may be mapped to function keys F1-F4 on standard

PC keyboards, and icons C1-C3 may be mapped to perform as combinations of function keys, such as CTRL-F1, CTRL-F2, and CTRL-F3, respectively. In addition, the icons labeled L1-L4 and C1-C3 may include circular regions which may simulate light emitting diodes (LEDs) which indicate that the function or element represented by the respective icon is active or being performed.

Icons L1-L4 may represent each of 4 lines available to the caller, and icons C1-C3 may represent conference calls using at least one line to connect, for example, two or more parties in a conference call. The icons L1-L4 and C1-C3 may indicate the activity of each respective line or conference line. For example, as illustrated in FIG. 5, icons L1-L2 may have lightly shaded or colored circles, such as a green circle, indicating that each of lines 1 and 2 are in use, while icons L3-L4 may have darkly shaded or color circles, such as a red or black circle, indicating that each of lines 3 and 4 are not in use. Similarly, the lightly shaded circle of the icon labeled C2 indicates that the function corresponding to C2 is active, as additionally indicated in the status are 38, while darkly shaded circles of icons labeled C1 and C3 indicate that such corresponding functions are not active.

The icons 42 are used in conjunction with the status area 38. For example, using a mouse for input, a line that is in use, as indicated by the lightly colored circle of the icon, may be activated to indicate a party's name by clicking a right mouse button for 5 seconds until another mouse click is actuated or the [ESC] key or icon is actuated. Thus, the user may switch between multiple calls in progress on respective lines.

Using the icons as well as an input device such as a mouse, a user may enter the name or alias or IP address, if known, of a party to be called by either manually entering the name, by using the speeddial feature, or by double clicking on an entry in a directory stored in the

memory, such as the memory 16 of the first processing unit 12, where the directory entries may be scrolled using the status area 38 and the down arrow icon 40.

Once a called party is listed in the status area 38 as being active on a line, the user may transfer the called party to another line or a conference line by clicking and dragging the status area 38, which is represented by a reduced icon 46. Dragging the reduced icon 46 to any one of line icons L1-L4 transfers the called party in use to the selected line, and dragging the reduced icon 46 to any one of conference line icons C1-C3 adds the called party to the selected conference call.

Other features may be supported, such as icons 48-52, where icon 48 corresponds to, for example, an ALT-X command to exit the communication facility of a processing unit, and icon 50 corresponds to, for example, an ALT-M command to minimize or maximize the display screen 36 by the output device of the processing unit. Icon 52 corresponds to an OPEN command, which may, for example, correspond to pressing the O key on a keyboard, to expand or contract the display screen 36 to represent the opening and closing of a cellular telephone. An "opened" configuration is shown in FIG. 5, and a "closed" configuration is shown in FIG. 6. In the "opened" configuration, additional features such as output volume (VOL) controls, input microphone (MIC) controls, waveform (WAV) sound controls, etc.

The use of display screens such as those shown in FIGS. 5-6 provided flexibility in implementing various features available to the user. It is to be understood that additional features such as those known in the art may be supported by the processing units 12, 22.

Alternatively, it is to be understood that one skilled in the art may implement the processing units 12, 22 to have the features of the display screens in FIGS. 5-6 in hardware; i.e. a wired telephone or wireless

cellular telephone may include various keys, LEDs, liquid crystal displays (LCDs), and touchscreen actuators corresponding to the icons and features shown in FIGS. 5-6. In addition, a PC may have the keys of a keyboard and mouse mapped to the icons and features shown in FIGS. 5-6.

Referring to FIG. 7, the disclosed point-to-point Internet protocol and system 10 is illustrated. First processing unit 12 initiates the point-to-point Internet protocol in step 56 by sending a query from the first processing unit 12 to the connection server 26. If connection server 26 is operative to perform the point-to-point Internet protocol, in step 58, first processing unit 12 receives an on-line status signal from the connection server 26, such signal may include the IP address of the callee or a "Callee Off-Line" message. Next, first processing unit 12 performs the primary point-to-point Internet protocol in step 60, which may include receiving, at the first processing unit 12, the IP address of the callee if the callee is active and on-line. Alternatively, processing unit 60 may initiate and perform the secondary point-to-point Internet protocol in step 62, if the called party is not active and/or on-line.

Referring to FIG. 8, in conjunction with FIGS. 1 and 3-4, the disclosed point-to-point Internet protocol and system 10 is illustrated. Connection server 26 starts the point-to-point Internet protocol, in step 64, and timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66. Connection server 26 receives a query from a first processing unit 12 in step 68 to determine whether a second user or second processing unit 22 is logged-in to the Internet 24, with the second user being specified, for example, by an E-mail address. Connection server 26 retrieves the IP address of the specified user from the database 34 in step 70, if the specified user is logged-in to the Internet, and sends the retrieved IP address to the first

processing unit 12 in step 72 to enable first processing unit 12 to establish point-to-point communications with the specified second user.

The disclosed secondary point-to-point Internet protocol operates as shown in FIG. 9. First processing unit 12 generates an E-mail signal, including a session number and a first IP address corresponding to a first processing unit in step 76. First processing unit 12 transmits the E-mail signal as a <ConnectRequest> signal to the Internet 24 in step 78. The E-mail signal is delivered through the Internet 24 using a mail server 28 to the second processing unit 22 in step 80. Second processing unit 22 extracts the session number and the first IP address from the E-mail signal in step 82 and transmits or sends the session number and a second IP address corresponding to the second processing unit 22, back to the first processing unit 12 through the Internet 24, in step 84. First processing unit 12 verifies the session number received from the second processing unit 22 in step 86, and establishes a point-to-point Internet communication link between the first processing unit 12 and second processing unit 22 using the first and second IP addresses in step 88.

While the disclosed point-to-point Internet protocols and system have been particularly shown and described with reference to the preferred embodiments, it is understood by those skilled in the art that various modifications in form and detail may be made therein without departing from the scope and spirit of the invention. Accordingly, modifications such as those suggested above, but not limited thereto, are to be considered within the scope of the invention.

Throughout this specification and the claims which follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

The reference to any prior art in this specification is not, and should not be taken as, an acknowledgment or any form of suggestion that that prior art forms part of the common general knowledge in Australia.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. A method for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, said method for use in a computer system, said method comprising:

A. providing a user interface element representing a first communication line;

B. providing a user interface element representing a first callee process; and

C. establishing a point-to-point communication link from the caller process to the first callee process, in response to a user associating the element representing the first callee process with the element representing the first communication line.

2. The method of claim 1 wherein step C further comprises the steps of:

c.1 querying the server as to the on-line status of the first callee process and

c.2 receiving a network protocol address of the first callee process over the computer network from the server.

3. The method of claim 1 further comprising the step of:

D. providing an element representing a second communication line.

4. The method of claim 3 further comprising the step of:

E. terminating the point-to-point communication link from the caller process to the first callee process, in response to the user disassociating the element representing the first callee process from the element representing the first communication line; and

F. establishing a different point-to-point communication link from the caller process to the first callee process, in response to the user associating the element representing the first callee process with the element representing the second communication line.

5. The method of claim 1 further comprising the steps of:

D. providing a user interface element representing a second callee process; and

E. establishing a conference point-to-point communication link between the caller process and the first and second callee process, in response to the user associating the element representing the second callee process with the element representing the first communication line.

6. The method of claim 5 further comprising the step of:

F. removing the second callee process from the conference point-to-point communication link in response to the user disassociating the element representing the second callee process from the element representing the first communication line.

7. The method of claim 1 further comprising the steps of:

D. providing a user interface element representing a communication line having a temporarily disabled status; and

E. temporarily disabling a point-to-point communication link between the caller process and the first callee process, in response to the user associating the element representing the first callee process with the element representing the communication line having a temporarily disabled status.

8. The method of claim 7 wherein the element provided in step D represents a communication line on hold status.

9. The method of claim 7 wherein the element provided in step D represents a communication line on mute status.

10. The method of claim 1 wherein the caller process further comprises a visual display and the user interface comprises a graphic user interface.

11. The method of claim 10 wherein the steps of establishing a point-to-point link as described in step C is performed in response to manipulation of the graphic elements on the graphic user interface.

12. A computer program product for use with a computer system comprising:

a computer useable medium having program code embodied in the medium for establishing a point-to-point communication link from a caller process to a callee process over a computer network, the caller process having a user interface and being operatively connectable to the callee process and a server over the computer network, the medium further comprising:

program code for generating an element representing a first communication line;

program code for generating an element representing a first callee process;

program code, responsive to a user associating the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process.

13. The computer program product of claim 12 wherein the program code for establishing a point-to-point communication link further comprises:

program code for querying the server as to the on-line status of the first callee process; and

program code for receiving a network protocol address of the first callee process over the computer network from the server.

14. A computer program product of claim 12 further comprising:

program code for generating an element representing a second communication line.

15. The computer program product of claim 14 further comprising:

program code, responsive to the user disassociating the element representing the first callee process from the element representing the first communication line, for terminating the point-to-point communication link from the caller process to the first callee process; and

program code, responsive to the user associating the element representing the first callee process with the element presenting the second communication line, for establishing a different point-to-point communication link from the caller process to the first callee process.

16. The computer program product of claim 12 further comprising:

program code for generating an element representing a second callee process; and

program code means, responsive to the user associating the element representing the second callee process with the element representing the first communication line, for establishing a conference communication link between the caller process and the first and second callee process.

17. The computer program product of claim 16 further comprising:

program code, responsive to the user disassociating the element representing the second callee process from the element representing the first communication line, for removing the second callee process from the conference communication link.

18. The computer program product of claim 12 further comprising:

program code for generating an element representing a communication line having a temporarily disabled status; and program code, responsive to the association of the element representing the first callee process with the element representing the communication line having a temporarily disabled status, for temporarily disabling the point-to-point communication link between the caller process and the first callee process.

19. The computer program product of claim 18 wherein the communication line having a temporarily disabled status comprises a communication line on hold status.

20. The computer program product of claim 18 wherein the communication line having a temporarily disabled status comprises a communication line on mute status.

21. A computer program product of claim 12 wherein the computer system further comprises a visual display and the user interface comprises a graphic user interface.

22. The computer program product of claim 21 wherein the element representing the first communication line and the element representing the first callee process are graphic elements and wherein the program code for establishing a point-to-point communication link from the caller process to the first callee process further comprises:

program code, responsive to manipulation of the graphic elements on the graphic user interface, for establishing the point-to-point communication link from the caller process to the first callee process.

23. A computer data signal embodied in a carrier wave comprising:

program code for generating a element representing a first communication line;

program code for generating an element representing a first callee process;

program code, responsive to association of the element representing the first callee process with the element representing the first communication line, for establishing a point-to-point communication link from the caller process to the first callee process over a computer network.

24. A method substantially as hereinbefore described with reference to the accompanying drawings.

25. A computer program product substantially as hereinbefore described with reference to the accompanying drawings.

26. A computer data signal substantially as hereinbefore described with reference to the accompanying drawings.

DATED this 8th day of September 2000

NetSpeak Corporation
By its Patent Attorneys
DAVIES COLLISON CAVE

13 09 00 59376

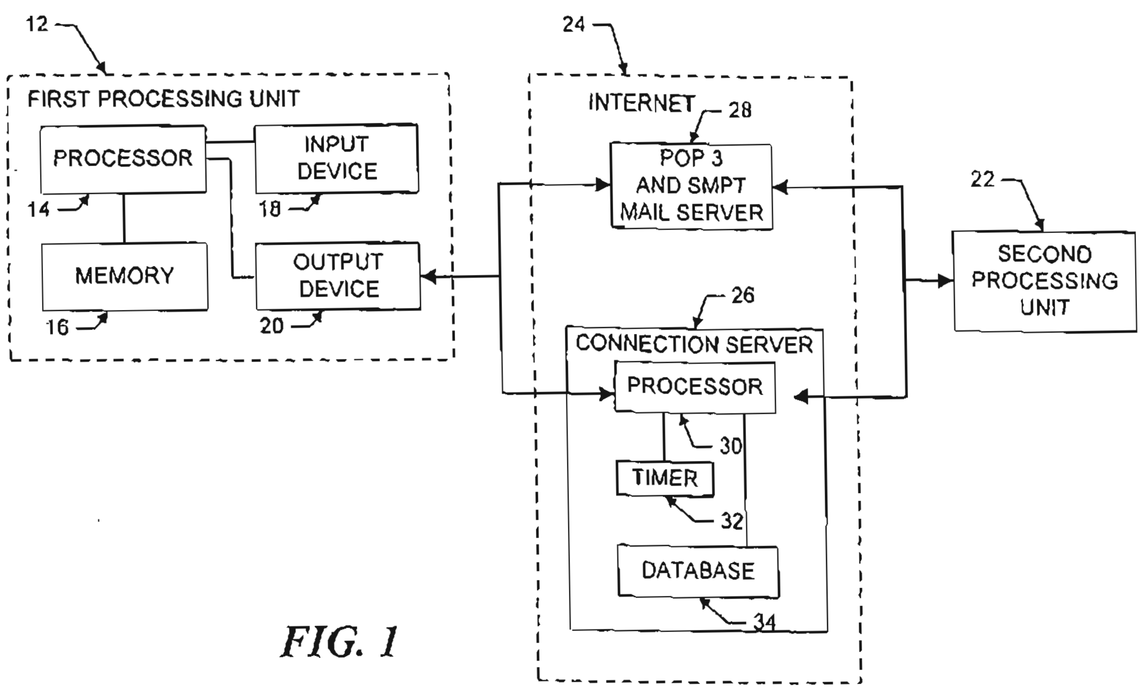


FIG. 1

13 09 00 59378

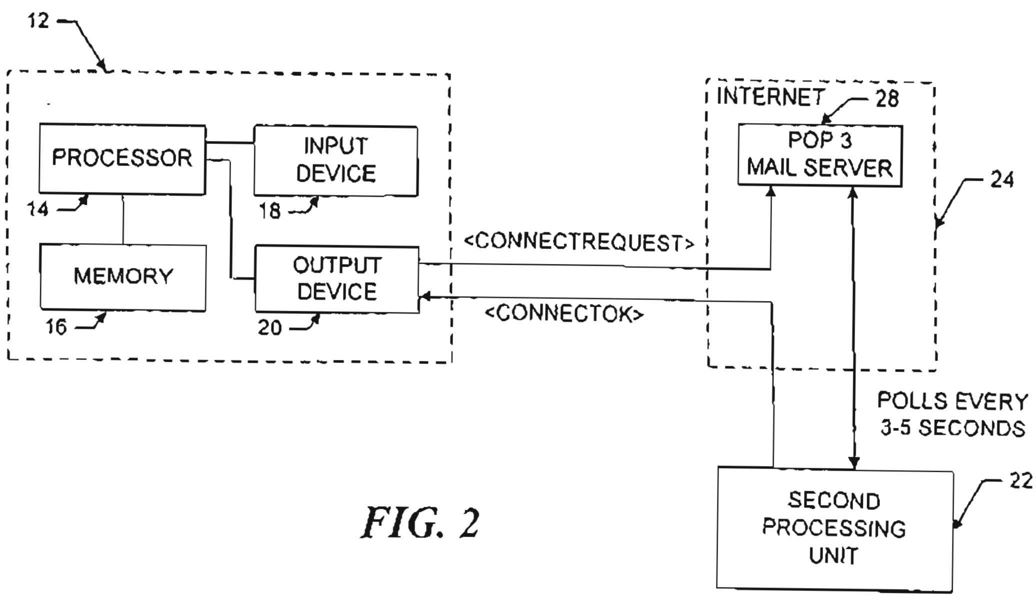


FIG. 2

13 09 00 59378

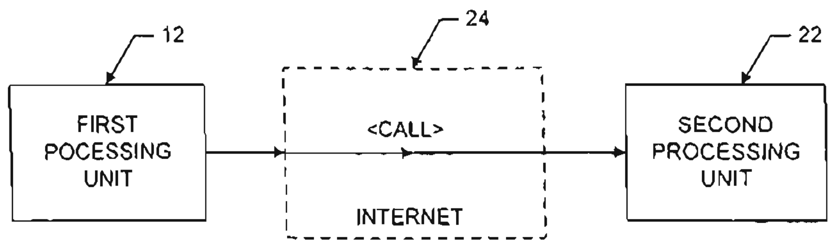


FIG. 3

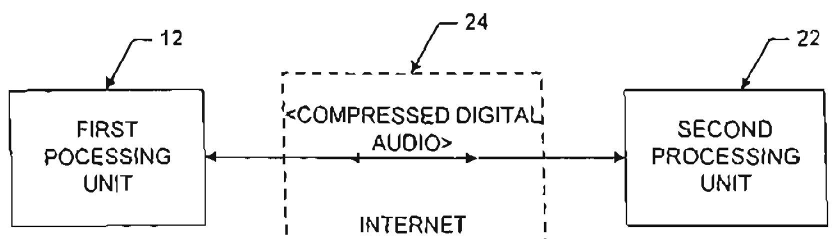


FIG. 4

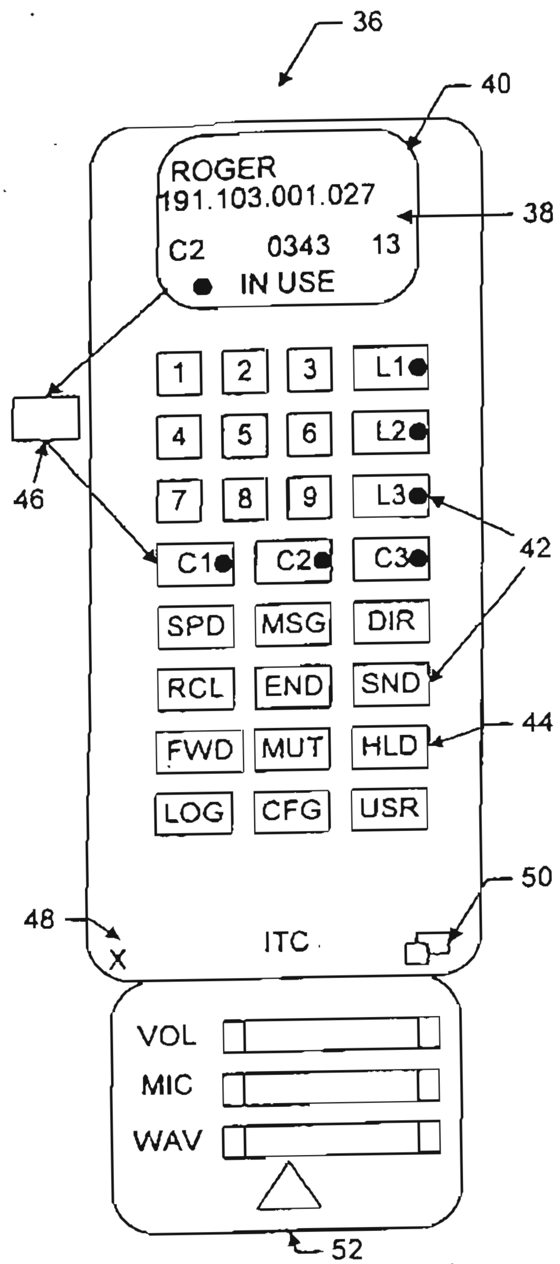

36
40
ROGER
191.103.001.027
C2 0343 13
IN USE
38
1 2 3 L1
4 5 6 L2
7 8 9 L3
C1 C2 C3
SPD MSG DIR
RCL END SND
FWD MUT HLD
LOG CFG USR
46
42
44
48
X
ITC
50
VOL
MIC
WAV
52

FIG. 5

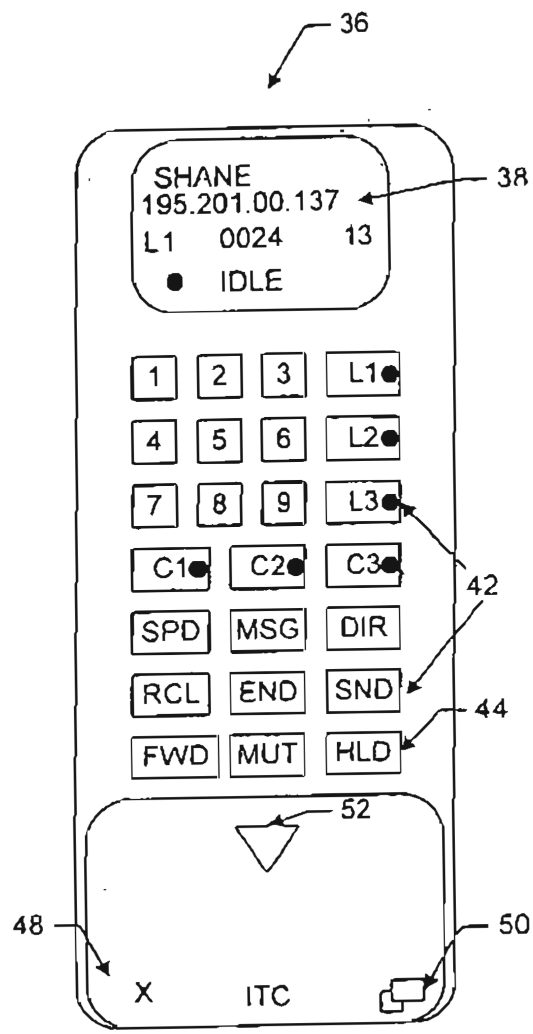

36
SHANE
195.201.00.137
L1 0024 13
IDLE
38
1 2 3 L1
4 5 6 L2
7 8 9 L3
C1 C2 C3
SPD MSG DIR
RCL END SND
FWD MUT HLD
42
44
52
48
X
ITC
50

FIG. 6

4/6

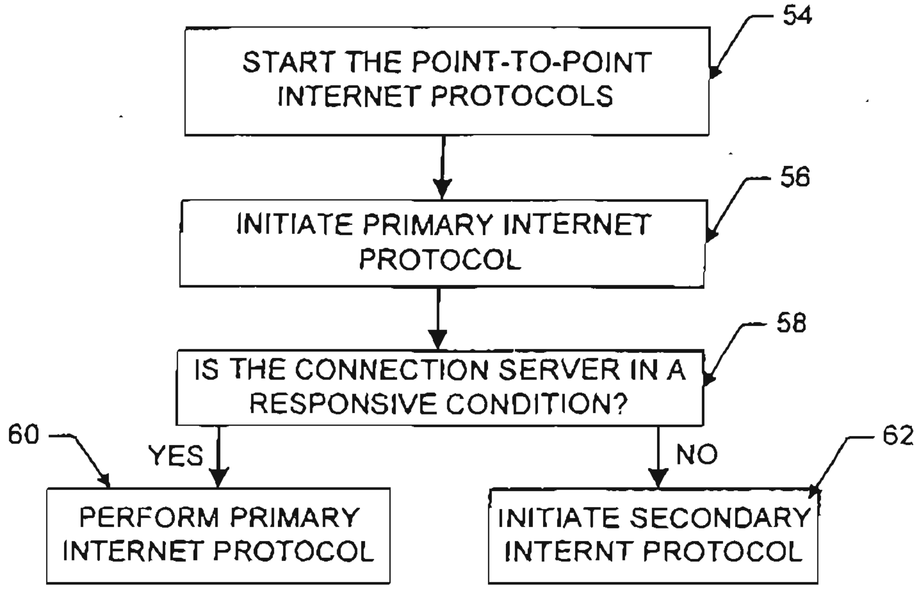

54
START THE POINT-TO-POINT INTERNET PROTOCOLS
56
INITIATE PRIMARY INTERNET PROTOCOL
58
IS THE CONNECTION SERVER IN A RESPONSIVE CONDITION?
60
YES
NO
62
PERFORM PRIMARY INTERNET PROTOCOL
INITIATE SECONDARY INTERNT PROTOCOL

FIG. 7

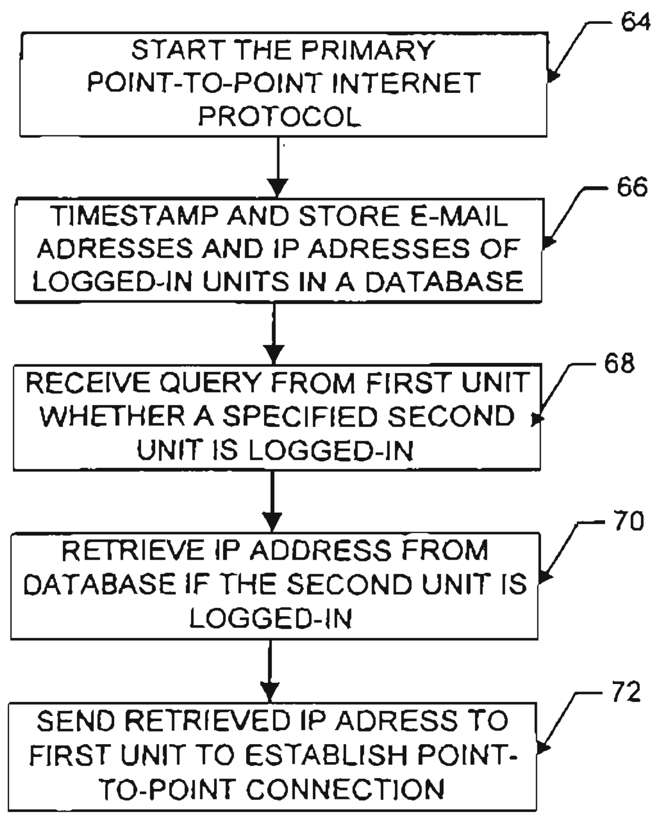

64
START THE PRIMARY POINT-TO-POINT INTERNET PROTOCOL
66
TIMESTAMP AND STORE E-MAIL ADRESSES AND IP ADRESSES OF LOGGED-IN UNITS IN A DATABASE
68
RECEIVE QUERY FROM FIRST UNIT WHETHER A SPECIFIED SECOND UNIT IS LOGGED-IN
70
RETRIEVE IP ADDRESS FROM DATABASE IF THE SECOND UNIT IS LOGGED-IN
72
SEND RETRIEVED IP ADRESS TO FIRST UNIT TO ESTABLISH POINT-TO-POINT CONNECTION

FIG. 8

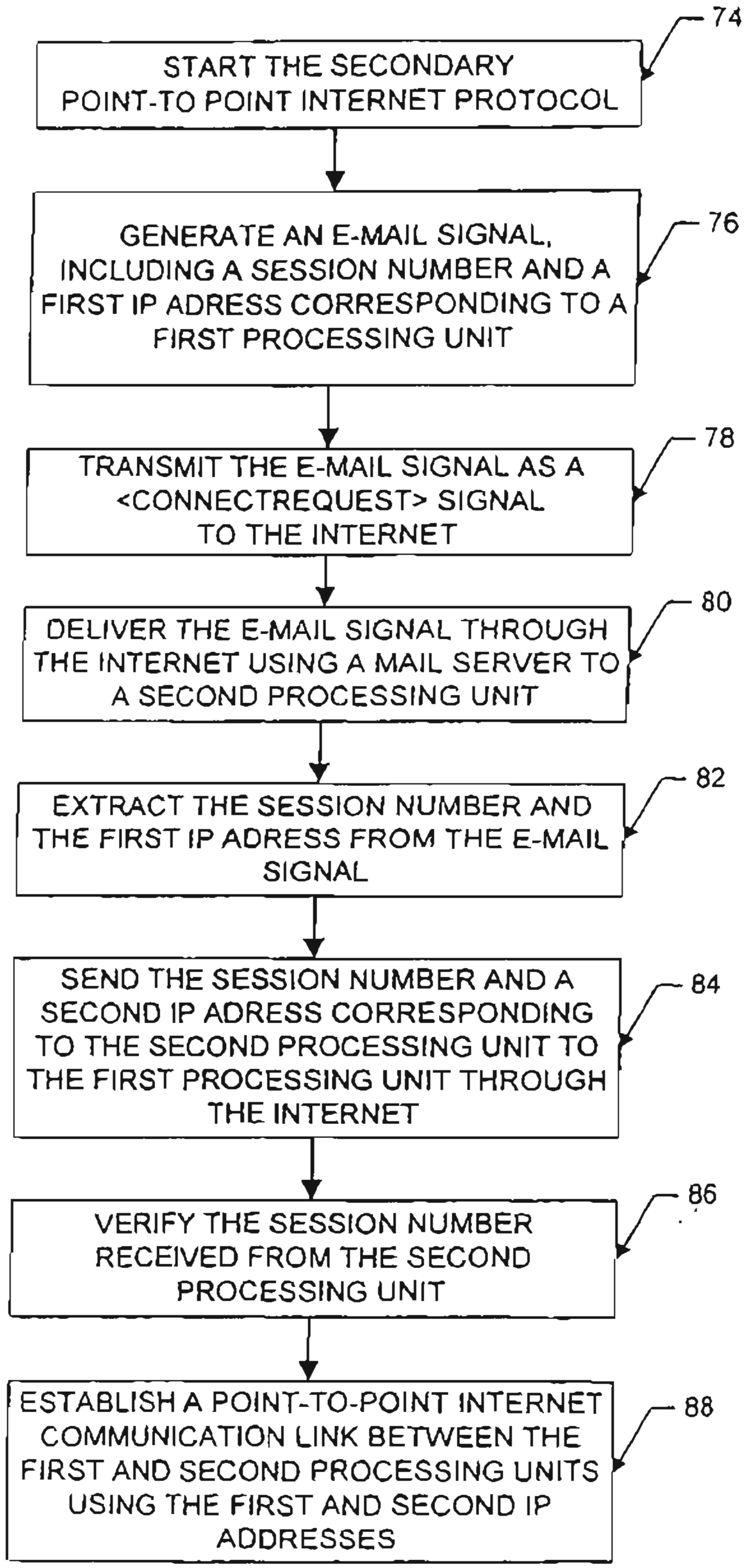

74
START THE SECONDARY POINT-TO POINT INTERNET PROTOCOL
76
GENERATE AN E-MAIL SIGNAL, INCLUDING A SESSION NUMBER AND A FIRST IP ADRESS CORRESPONDING TO A FIRST PROCESSING UNIT
78
TRANSMIT THE E-MAIL SIGNAL AS A <CONNECTREQUEST> SIGNAL TO THE INTERNET
80
DELIVER THE E-MAIL SIGNAL THROUGH THE INTERNET USING A MAIL SERVER TO A SECOND PROCESSING UNIT
82
EXTRACT THE SESSION NUMBER AND THE FIRST IP ADRESS FROM THE E-MAIL SIGNAL
84
SEND THE SESSION NUMBER AND A SECOND IP ADRESS CORRESPONDING TO THE SECOND PROCESSING UNIT TO THE FIRST PROCESSING UNIT THROUGH THE INTERNET
86
VERIFY THE SESSION NUMBER RECEIVED FROM THE SECOND PROCESSING UNIT
88
ESTABLISH A POINT-TO-POINT INTERNET COMMUNICATION LINK BETWEEN THE FIRST AND SECOND PROCESSING UNITS USING THE FIRST AND SECOND IP ADDRESSES

FIG. 9

(12) **PATENT APPLICATION** (11) Application No. **AU 200059379 A1**
(19) **AUSTRALIAN PATENT OFFICE**

(54) Title
Point-to-point internet protocol

(51)[7] International Patent Classification(s)
H04L 029/00 G06F 019/00

(21) Application No: **200059379** (22) Application Date: **2000.09.13**

(43) Publication Date : **2000.11.23**
(43) Publication Journal Date : **2000.11.23**

(62) Divisional of:
199672476

(71) Applicant(s)
NetSpeak Corporation

(72) Inventor(s)
Glenn W Hutton; Shane D Mattaway; Craig B Strickland

(74) Agent/Attorney
DAVIES COLLISON CAVE,1 Little Collins Street,MELBOURNE VIC 3000

Abstract

An apparatus for establishing a point-to-point communication link said apparatus operating in a computer system operatively coupled to other computers and a server over a computer network, the apparatus comprising means for transmitting, from the first process to a server a query as to whether a second process is connected to the computer network, means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network, and means, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

13 09 00 59379

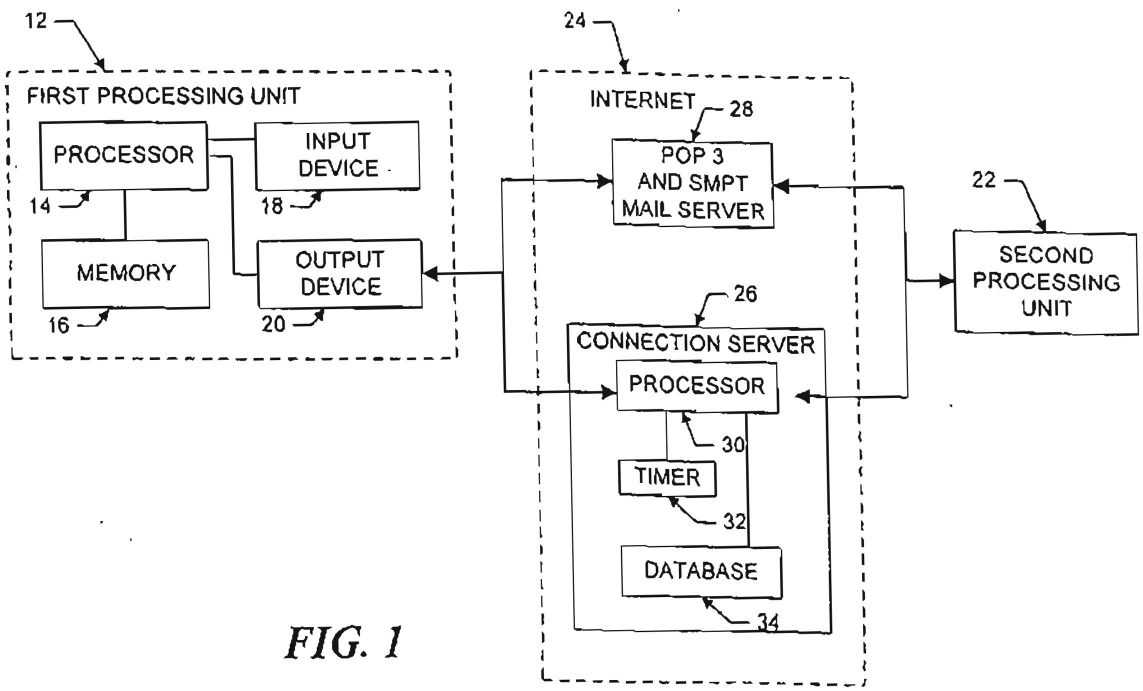


FIG. 1

P/00/011 28/5/91
Regulation 3.2

AUSTRALIA

Patents Act 1990

COMPLETE SPECIFICATION

STANDARD PATENT

(ORIGINAL)

Name of Applicant: **NetSpeak Corporation**, of 902 Clint Moore Road, Suite 104, Boca Raton, Florida 33487, United States of America.

Actual Inventors: HUTTON, Glenn W
MATTAWAY, Shane D
STRICKLAND, Craig B

Address for Service: **DAVIES COLLISON CAVE**, Patent Attorneys, of 1 Little Collins Street, Melbourne, Victoria 3000, Australia.

Invention Title: **"Point-to-Point Internet Protocol"**

The following statement is a full description of this invention, including the best method of performing it known to us:

POINT-TO-POINT INTERNET PROTOCOL

The present invention relates, in general, to data processing systems, and more specifically, to a method and apparatus for facilitating audio communications over computer networks.

The increased popularity of on-line services such as AMERICA ONLINE™, COMPUSERVE®, and other services such as Internet gateways have spurred applications to provide multimedia, including video and voice clips, to online users. An example of an online voice clip application is VOICE E-MAIL FOR WINCIM and VOICE E-MAIL FOR AMERICA ONLINE™, available from Bonzi Software, as described in "Simple Utilities Send Voice E-Mail Online", MULTIMEDIA WORLD, VOL. 2, NO. 9, August 1995, p. 52. Using such Voice E-Mail software, a user may create an audio message to be sent to a predetermined E-mail address specified by the user.

Generally, devices interfacing to the Internet and other online services may communicate with each other upon establishing respective device addresses. One type of device address is the Internet Protocol (IP) address, which acts as a pointer to the device associated with the IP address. A typical device may have a Serial Line Internet Protocol or Point-to-Point Protocol (SLIP/PPP) account with a permanent IP address for receiving E-mail, voicemail, and the like over the Internet. E-mail and voicemail is generally intended to convey text, audio, etc., with any routing information such as an IP address and routing headers generally

being considered an artifact of the communication, or even gibberish to the recipient.

Devices such as a host computer or server of a company may include multiple modems for connection of users to the Internet, with a temporary IP address allocated to each user. For example, the host computer may have a general IP address "XXX.XXX.XXX," and each user may be allocated a successive IP address of XXX.XXX.XXX.10, XXX.XXX.XXX.11, XXX.XXX.XXX.12, etc. Such temporary IP addresses may be reassigned or recycled to the users, for example, as each user is successively connected to an outside party. For example, a host computer of a company may support a maximum of 254 IP addresses which are pooled and shared between devices connected to the host computer.

Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet. For example, realtime video teleconferencing has been implemented using dedicated IP addresses and mechanisms known as reflectors.

A technique for matching domain names to Internet Protocol addresses is described in the text entitled "Internetworking With TCP/IP", 2nd Edition, by Douglas E. Comer, November 1992, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A. Comer describes a domain name system and cooperative systems of name servers for matching domain names to network addresses. Each name server is a server program that supplies mapping of domain names to IP addresses. The system described in Comer, however, is not designed for use with network nodes whose network names or name to address bindings change frequently.

International Publication WO 92/19054 discloses a network

monitoring system including an address tracking module which uses passive monitoring of all packet communications over a local area network to maintain a name table of IP address mappings. The disclosed address tracking module is capable of monitoring only a small number of nodes on a local area network and is not suitable for use with a multitude of nodes over a wide area network.

None of the above-described systems are suitable for use with processes which have dynamically assigned network protocol addresses and which are communicating over wide area or global networks.

Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain.

In accordance with the present invention, there is provided an apparatus for establishing a point-to-point communication link said apparatus operating in a computer system operatively coupled to other computers and a server over a computer network, said apparatus comprising:

a. means for transmitting, from the first process to a server a query as to whether a second process is connected to the computer network;

b. means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network; and

c. means, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides an apparatus for use with a computer system capable of executing a first process and communicating with other processes, a directory server process and a mail server process over a computer network, the apparatus comprising:

A. program logic configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;

B. program logic configured to establish a communication connection with the directory server process once the assigned network protocol of the first process is known;

C. program logic configured to forward the assigned network protocol address of the first process to the directory server process upon establishing a communication connection with the directory server process; and

D. program logic configured to establish a point-to-point communication with another process over the computer network.

The present invention also provides an apparatus for use with a computer system, the computer system capable of executing a first process connectable over a computer network to a second process and a directory database server process, the apparatus comprising:

a. program logic configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. program logic responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.

The present invention also provides a computer program product for use with a computer system, the computer system executing a first process operatively connectable over a computer network to a second process and a server process, the computer program product comprising a computer useable medium having computer readable program code embodied therein, the program code comprising:

a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. program code responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.

The present invention also provides a method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

A. following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network;

B. querying the address server as to whether the second process is connected to the computer network;

C. receiving a network protocol address of the second process from the address server, when the second process is connected to the computer network; and

D. in response to the network protocol address of the second process, establishing a point-to-point communication link with the second process over the computer network.

9 The present invention also provides a method for establishing point-to-point communications with other processes, said method for use in a computer system capable of executing a first process and communicating with other processes and a server process over a computer network, said method comprising:

A. determining the currently assigned network protocol address of the first process upon connection to the computer network;

B. establishing a communication connection with the server process once the assigned network protocol of the first process is known;

C. forwarding the assigned network protocol address of the first process to the server process upon establishing a communication connection with the server process; and

D. establishing a point-to-point communication with another process over the computer network.

The present invention also provides A method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

a. accessing a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. in response to one of the network protocol addresses, establishing a point-to-point communication link from the first process to the second process over the computer network.

The present invention also provides a method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

a. transmitting to the server a network protocol address received by the first process following connection to the computer network;

b. transmitting, to the server, a query as to whether the second process is connected to the computer network;

c. receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

d. in response to the network protocol address of the second process, establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides an apparatus capable of executing a first process and connecting to other processes and a server process over a computer network, the apparatus comprising:

a. program logic configured to generate a user-interface enabling control of a first process executing on the computer system;

b. program logic configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;

c. program logic, responsive to the currently assigned network protocol address of the first process, and configured to establish a communication connection with the server process and to forward the assigned network protocol address of the first process to the server process upon establishing a communication connection with the server process; and

d. program logic, responsive to user input commands, and configured to establish a point-to-point communications with another process over the computer network.

The present invention also provides an apparatus for use with a computer system, the computer system executing a first process operatively coupled over a computer network to a second process and a directory database server process, the apparatus comprising:

A. program logic configured to, following connection of the first process to the computer network, forward to the address server a network protocol address at which the first process is connected to the computer network;

B. program logic configured to query the address server as to whether the second process is connected to the computer network;

C. program logic configured to receive a network protocol address of the second process from the address server, when the second process is connected to the computer network; and

D. program logic configured to, in response to the network protocol address of the second process, establish a point-to-point communication link with the second process over the computer network.

The present invention also provides Apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system capable of executing a first process and operatively connectable to other processes and a server over a computer network, said apparatus comprising:

a. means for transmitting, from the first process to the server, a query as to whether a second process is connected to the computer network;

b. means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network; and

c. means, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides a computer data signal embodied in a carrier wave comprising:

program code for transmitting to a server a network protocol address received by a first process following connection to a computer network;

program code for transmitting, to the server, a query as to whether a second process is connected to the computer network;

program code for receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

The present invention also provides a computer program product for use with a computer system, the computer system executing a first process and operatively connectable to a second process and a server over a computer network, the computer program product comprising a computer useable medium having program code embodied in the medium, the program code comprising:

program code for transmitting to the server a network protocol address received by the first process following connection to the computer network;

program code for transmitting, to the server, a query as to whether the second process is connected to the computer network;

program code for receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

Preferred embodiments of the present invention are hereinafter described, by way of example only, with reference to the following drawings, wherein:

FIG 1 illustrates, in block diagram format, a system for the disclosed point-to-point Internet protocol;

FIG 2 illustrates, in block diagram format, the system using a secondary point-to-point Internet protocol;

FIG 3 illustrates, in block diagram format, the system of FIGS 1-2 with the point-to-point Internet protocol established;

FIG 4 is another block diagram of the system of FIGS 1-2 with audio communications being conducted;

FIG. 5 illustrates a display screen for a processing unit;

FIG. 6 illustrates another display screen for a processing unit;

FIG. 7 illustrates a flowchart of the initiation of the point-to-point Internet protocols;

FIG. 8 illustrates a flowchart of the performance of the primary point-to-point Internet protocols; and

FIG. 9 illustrates a flowchart of the performance of the secondary point-to-point Internet protocol.

Referring now in specific detail to the drawings, with like reference numerals identifying similar or identical elements, as shown in FIG. 1, the present disclosure describes a point-to-point network protocol and system 10 for using such a protocol.

In an exemplary embodiment, the system 10 includes a first processing unit 12 for sending at least a voice signal from a first user to a second user. The first processing unit 12 includes a processor 14, a memory 16, an input device 18, and an output device 20. The output device 20 includes at least one modem capable of, for example, 14.4 kbaud communications and operatively connected via wired and/or wireless communication connections to the Internet or other computer networks such as an Intranet, i.e., a private computer network. One skilled in the art would understand that the input device 18 may be implemented at least in part by the modem of the output device 20 to allow input signals from the communication connections to be received. The second processing unit 22 may have a processor, memory, and input and output devices, including at least one modem and associated communication connections, as described above for the first processing unit 12. In an exemplary embodiment, each of the processing units 12, 22 may execute the WEBPHONE™ Internet telephony application available from NetSpeak Corporation, Boca Raton, FL, which is capable of performing the disclosed point-to-point Internet protocol and system 10, as described herein.

The first processing unit 12 and the second processing unit 22 are operatively connected to the Internet 24 by communication devices and software known in the art, such as an Internet Service Provider (ISP) or an Internet gateway. The processing units 12, 22 may be operatively interconnected through the Internet 24 to a connection server 26, and

may also be operatively connected to a mail server 28 associated with the Internet 24.

The connection server 26 includes a processor 30, a timer 32 for generating time stamps, and a memory such as a database 34 for storing, for example, E-mail and Internet Protocol (IP) addresses of logged-in units. In an exemplary embodiment, the connection server 26 may be a SPARC 5 server or a SPARC 20 server, available from SUN MICROSYSTEMS, INC., Mountain View, CA, having a central processing unit (CPU) as processor 30, an operating system (OS) such as UNIX, for providing timing operations such as maintaining the timer 32, a hard drive or fixed drive, as well as dynamic random access memory (DRAM) for storing the database 34, and a keyboard and display and/or other input and output devices (not shown in FIG. 1). The database 34 may be an SQL database available from ORACLE or INFORMIX.

In an exemplary embodiment, the mail server 28 may be a Post Office Protocol (POP) Version 3 mail server including a processor, memory, and stored programs operating in a UNIX environment, or, alternatively, another OS, to process E-mail capabilities between processing units and devices over the Internet 24.

The first processing unit 12 may operate the disclosed point-to-point Internet protocol by a computer program described hereinbelow in conjunction with FIG. 6, which may be implemented from compiled and /or interpreted source code in the C++ programming language and which may be downloaded to the first processing unit 12 from an external computer. The operating computer program may be stored in the memory 16, which may include about 8 MB RAM and/or a hard or fixed drive having about 8 MB. Alternatively, the source code may be implemented in the first processing unit 12 as firmware, as an erasable read only memory (EPROM), etc. It is understood that one skilled in the

art would be able to use programming languages other than C++ to implement the disclosed point-to-point network protocol and system 10.

The processor 14 receives input commands and data from a first user associated with the first processing unit 12 though the input device 18, which may be an input port connected by a wired, optical, or a wireless connection for electromagnetic transmissions, or alternatively may be transferable storage media, such as floppy disks, magnetic tapes, compact disks, or other storage media including the input data from the first user.

The input device 18 may include a user interface (not shown) having, for example, at least one button actuated by the user to input commands to select from a plurality of operating modes to operate the first processing unit 12. In alternative embodiments, the input device 18 may include a keyboard, a mouse, a touch screen, and/or a data reading device such as a disk drive for receiving the input data from input data files stored in storage media such as a floppy disk or, for example, an 8 mm storage tape. The input device 18 may alternatively include connections to other computer systems to receive the input commands and data therefrom.

The first processing unit 12 may include a visual interface for use in conjunction with the input device 18 and output device 20 similar to those screens illustrated in FIGS. 5-6, discussed below. It is also understood that alternative devices may be used to receive commands and data from the user, such as keyboards, mouse devices, and graphical user interfaces (GUI) such as WINDOWS™ 3.1 available form MICROSOFT Corporation, Redmond, WA., and other operating systems and GUIs, such as OS/2 and OS/2 WARP, available from IBM CORPORATION, Boca Raton, FL. Processing unit 12 may also include microphones and/or telephone handsets for receiving audio voice data

and commands, speech or voice recognition devices, dual tone multi-frequency (DTMF) based devices, and/or software known in the art to accept voice data and commands and to operate the first processing unit 12.

In addition, either of the first processing unit 12 and the second processing unit 22 may be implemented in a personal digital assistant (PDA) providing modem and E-mail capabilities and Internet access, with the PDA providing the input/output screens for mouse interactions or for touchscreen activation as shown, for example, in FIGS. 5-6, as a combination of the input device 18 and output device 20.

For clarity of explanation, the illustrative embodiment of the disclosed point-to-point Internet protocol and system 10 is presented as having individual functional blocks, which may include functional blocks labeled as "processor" and "processing unit". The functions represented by these blocks may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. For example, the functions of each of the processors and processing units presented herein may be provided by a shared processor or by a plurality of individual processors. Moreover, the use of the functional blocks with accompanying labels herein is not to be construed to refer exclusively to hardware capable of executing software. Illustrative embodiments may include digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, read-only memory (ROM) for storing software performing the operations discussed below, and random access memory (RAM) for storing DSP results. Very large scale integration (VLSI) hardware embodiments, as well as custom VLSI circuitry in combination with a general purpose DSP circuit, may also be provided. Any and all of these embodiments may be deemed to fail within the meaning of the labels for the functional blocks as used herein.

The processing units 12, 22 are capable of placing calls and connecting to other processing units connected to the Internet 24, for example, via dialup SLIP/PPP lines. In an exemplary embodiment, each processing unit assigns an unsigned long session number, for example, a 32- bit long sequence in a *.ini file for each call. Each call may be assigned a successive session number in sequence, which may be used by the respective processing unit to associate the call with one of the SLIP/PPP lines, to associate a <ConnectOK> response signal with a <Connect Request> signal, and to allow for multiplexing and demultiplexing of inbound and outbound conversations on conference lines, as explained hereinafter.

For callee (or called) processing units with fixed IP addresses, the caller (or calling) processing unit may open a "socket", i.e. a file handle or address indicating where data is to be sent, and transmit a <Call> command to establish communication with the callee utilizing, for example, datagram services such as Internet Standard network layering as well as transport layering, which may include a Transport Control Protocol (TCP) or a User Datagram Protocol (UDP) on top of the IP. Typically, a processing unit having a fixed IP address may maintain at least one open socket and a called processing unit waits for a <Call> command to assign the open socket to the incoming signal. If all lines are in use, the callee processing unit sends a BUSY signal or message to the callee processing unit. As shown in FIG. 1, the disclosed point-to-point Internet protocol and system 10 operate when a callee processing unit does not have a fixed or predetermined IP address. In the exemplary embodiment and without loss of generality, the first processing unit 12 is the caller processing unit and the second processing unit 22 is the called processing unit. When either of processing units 12, 22 logs on to the Internet via a dial-up connection, the respective unit is provided a

dynamically allocated IP address by the a connection service provider.

Upon the first user initiating the point-to-point Internet protocol when the first user is logged on to the Internet 24, the first processing unit 12 automatically transmits its associated E-mail address and its dynamically allocated IP address to the connection server 26. The connection server 26 then stores these addresses in the database 34 and time stamps the stored addresses using timer 32. The first user operating the first processing unit 12 is thus established in the database 34 as an active on-line party available for communication using the disclosed point-to-point Internet protocol. Similarly, a second user operating the second processing unit 22, upon connection to the Internet 24 through the a connection service provider, is processed by the connection server 26 to be established in the database 34 as an active on-line party.

The connection server 26 may use the time stamps to update the status of each processing unit; for example, after 2 hours, so that the on-line status information stored in the database 34 is relatively current. Other predetermined time periods, such as a default value of 24 hours, may be configured by a systems operator.

The first user with the first processing unit 12 initiates a call using, for example, a Send command and/or a command to speeddial an N^{TH} stored number, which may be labeled [SND] and [SPD] [N], respectively, by the input device 18 and/or the output device 20, such as shown in FIGS. 5-6. In response to either the Send or speeddial commands, the first processing unit 12 retrieves from memory 16 a stored E-mail address of the callee corresponding to the N^{TH} stored number. Alternatively, the first user may directly enter the E-mail address of the callee.

The first processing unit 12 then sends a query, including the E-mail address of the callee, to the connection server 26. The connection

server 26 then searches the database 34 to determine whether the callee is logged-in by finding any stored information corresponding to the callee's E-mail address indicating that the callee is active and on-line. If the callee is active and on-line, the connection server 26 then performs the primary point-to-point Internet protocol; i.e. the IP address of the callee is retrieved from the database 34 and sent to the first processing unit 12. The first processing unit 12 may then directly establish the point-to-point Internet communications with the callee using the IP address of the callee.

If the callee is not on-line when the connection server 26 determines the callee's status, the connection server 26 sends an OFF-LINE signal or message to the first processing unit 12. The first processing unit 12 may also display a message such as "Called Party Off-Line" to the first user.

When a user logs off or goes off-line from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user's information, or by flagging the user as being off-line. The connection server 26 may be instructed to update the user's information in the database 34 by an off-line message, such as a data packet, sent automatically from the processing unit of the user prior to being disconnected from the connection server 26. Accordingly, an off-line user is effectively disabled from making and/or receiving point-to-point Internet communications.

As shown in FIGS. 2-4, the disclosed secondary point-to-point Internet protocol may be used as an alternative to the primary point-to-point Internet protocol described above, for example, if the connection server 26 is non-responsive, inoperative, and/or unable to perform the primary point-to-point Internet protocol, as a non-responsive condition. Alternatively, the disclosed secondary point-to-point Internet protocol may

be used independent of the primary point-to-point Internet protocol. In the disclosed secondary point-to-point Internet protocol, the first processing unit 12 sends a <ConnectRequest> message via E-mail over the Internet 24 to the mail server 28. The E-mail including the <ConnectRequest> message may have, for example, the subject

[*wp#XXXXXXXX#nnn.nnn.nnn.#emailAddr]

where nnn.nnn.nnn.nnn. is the current (i.e. temporary or permanent) IP address of the first user, and XXXXXXXX is a session number, which may be unique and associated with the request of the first user to initiate point-to-point communication with the second user.

As described above, the first processing unit 12 may send the <ConnectRequest> message in response to an unsuccessful attempt to perform the primary point-to-point Internet protocol. Alternatively, the first processing unit 12 may send the <ConnectRequest> message in response to the first user initiating a SEND command or the like.

After the <ConnectRequest> message via E-mail is sent, the first processing unit 12 opens a socket and waits to detect a response from the second processing unit 22. A timeout timer, such as timer 32, may be set by the first processing unit 12, in a manner known in the art, to wait for a predetermined duration to receive a <ConnectOK> signal. The processor 14 of the first processing unit 12 may cause the output device 20 to output a Ring signal to the user, such as an audible ringing sound, about every 3 seconds . For example, the processor 14 may output a *.wav file, which may be labeled RING.WAV, which is processed by the output device 20 to output an audible ringing sound.

The mail server 28 then polls the second processing unit 22, for example, every 3-5 seconds, to deliver the E-mail. Generally, the second processing unit 22 checks the incoming lines, for example, at regular intervals to wait for and to detect incoming E-mail from the mail server 28

through the Internet 24.

Typically, for sending E-mail to users having associated processing units operatively connected to a host computer or server operating an Internet gateway, E-mail for a specific user may be sent over the Internet 24 and directed to the permanent IP address or the SLIP/PPP account designation of the host computer, which then assigns a temporary IP address to the processing unit of the specified user for properly routing the E-mail. The E-mail signal may include a name or other designation such as a user name which identifies the specific user regardless of the processing unit assigned to the user; that is, the host computer may track and store the specific device where a specific user is assigned or logged on, independent of the IP address system, and so the host computer may switch the E-mail signal to the device of the specific user. At that time, a temporary IP address may be generated or assigned to the specific user and device.

Upon detecting and/or receiving the incoming E-mail signal from the first processing unit 12, the second processing unit 22 may assign or may be assigned a temporary IP address. Therefore, the delivery of the E-mail through the Internet 24 provides the second processing unit 22 with a session number as well as IP addresses of both the first processing unit 12 and the second processing unit 22.

Point-to-point communication may then be established by the processing unit 22 processing the E-mail signal to extract the <ConnectRequest> message, including the IP address of the first processing unit 12 and the session number. The second processing unit 22 may then open a socket and generate a <ConnectOK> response signal, which includes the temporary IP address of the second processing unit 22 as well as the session number of the first processing unit.

The second processing unit 22 sends the <ConnectOK> signal

directly over the Internet 24 to the IP address of the first processing unit 12 without processing by the mail server 28, and a timeout timer of the second processing unit 22 may be set to wait and detect a <Call> signal expected from the first processing unit 12.

Realtime point-to-point communication of audio signals over the Internet 24, as well as video and voicemail, may thus be established and supported without requiring permanent IP addresses to be assigned to either of the users or processing units 12, 22. For the duration of the realtime point-to-point link, the relative permanence of the current IP addresses of the processing units 12, 22 is sufficient, whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).

In the exemplary embodiment, a first user operating the first processing unit 12 is not required to be notified by the first processing unit 12 that an E-mail is being generated and sent to establish the point-to-point link with the second user at the second processing unit 22. Similarly, the second user is not required to be notified by the second processing unit 22 that an E-mail has been received and/or a temporary IP address is associated with the second processing unit 22. The processing units 12, 22 may perform the disclosed point-to-point Internet protocol automatically upon initiation of the point-to-point communication command by the first user without displaying the E-mail interactions to either user. Accordingly, the disclosed point-to-point Internet protocol may be transparent to the users. Alternatively, either of the first and second users may receive, for example, a brief message of "CONNECTION IN PROGRESS" or the like on a display of the respective output device of the processing units 12, 22.

After the initiation of either the primary or the secondary point-to-

point Internet protocols described above in conjunction with FIGS. 1-2, the point-to-point communication link over the Internet 24 may be established as shown in FIGS. 3-4 in a manner known in the art. For example, referring to FIG. 3, upon receiving the <ConnectorOK> signal from the second processing unit 22, the first processing unit 12 extracts the IP address of the second processing unit 22 and the session number, and the session number sent from the second processing unit 22 is then checked with the session number originally sent from the first processing unit 12 in the <ConnectRequest> message as E-mail. If the session numbers sent and received by the processing unit 12 match, then the first processing unit 12 sends a <Call> signal directly over the Internet 24 to the second processing unit 22; i.e. using the IP address of the second processing unit 22 provided to the first processing unit 12 in the <ConnectOK> signal.

Upon receiving the <Call> signal, the second processing unit 22 may then begin a ring sequence, for example, by indicating or annunciating to the second user that an incoming call is being received. For example, the word "CALL" may be displayed on the output device of the second processing unit 22. The second user may then activate the second processing unit 22 to receive the incoming call.

Referring to FIG. 4, after the second processing unit 22 receives the incoming call, realtime audio and/or video conversations may be conducted in a manner known in the art between the first and second users through the Internet 24, for example, by compressed digital audio signals. Each of the processing units 12, 22 also display to each respective user the words "IN USE" to indicate that the point-to-point communication link is established and audio or video signals are being transmitted.

In addition, either user may terminate the point-to-point

communication link by, for example, activating a termination command, such as by activating an [END] button or icon on a respective processing unit, causing the respective processing unit to send an <End> signal which causes both processing units to terminate the respective sockets, as well as to perform other cleanup commands and functions known in the art.

FIGS. 5-6 illustrate examples of display screens 36 which may be output by a respective output device of each processing unit 12, 22 of FIGS. 1-4 for providing the disclosed point-to-point Internet protocol and system 10. Such display screens may be displayed on a display of a personal computer (PC) or a PDA in a manner known in the art.

As shown in FIG. 5, a first display screen 36 includes a status area 38 for indicating, for example, a called user by name and/or by IP address or telephone number; a current function such as C2; a current time; a current operating status such as "IN USE", and other control icons such as a down arrow icon 40 for scrolling down a list of parties on a current conference line. The operating status may include such annunciators as "IN USE," "IDLE," "BUSY," "NO ANSWER," "OFFLINE," "CALL," "DIALING," "MESSAGES," and "SPEEDDIAL."

Other areas of the display screen 36 may include activation areas or icons for actuating commands or entering data. For example, the display screen 36 may include a set of icons 42 arranged in columns and rows including digits 0-9 and commands such as END, SND, HLD, etc. For example, the END and SND commands may be initiated as described above, and the HLD icon 44 may be actuated to place a current line on hold. Such icons may also be configured to substantially simulate a telephone handset or a cellular telephone interface to facilitate ease of use, as well as to simulate function keys of a keyboard. For example, icons labeled L1-L4 may be mapped to function keys F1-F4 on standard

PC keyboards, and icons C1-C3 may be mapped to perform as combinations of function keys, such as CTRL-F1, CTRL-F2, and CTRL-F3, respectively. In addition, the icons labeled L1-L4 and C1-C3 may include circular regions which may simulate light emitting diodes (LEDs) which indicate that the function or element represented by the respective icon is active or being performed.

Icons L1-L4 may represent each of 4 lines available to the caller, and icons C1-C3 may represent conference calls using at least one line to connect, for example, two or more parties in a conference call. The icons L1-L4 and C1-C3 may indicate the activity of each respective line or conference line. For example, as illustrated in FIG. 5, icons L1-L2 may have lightly shaded or colored circles, such as a green circle, indicating that each of lines 1 and 2 are in use, while icons L3-L4 may have darkly shaded or color circles, such as a red or black circle, indicating that each of lines 3 and 4 are not in use. Similarly, the lightly shaded circle of the icon labeled C2 indicates that the function corresponding to C2 is active, as additionally indicated in the status are 38, while darkly shaded circles of icons labeled C1 and C3 indicate that such corresponding functions are not active.

The icons 42 are used in conjunction with the status area 38. For example, using a mouse for input, a line that is in use, as indicated by the lightly colored circle of the icon, may be activated to indicate a party's name by clicking a right mouse button for 5 seconds until another mouse click is actuated or the [ESC] key or icon is actuated. Thus, the user may switch between multiple calls in progress on respective lines.

Using the icons as well as an input device such as a mouse, a user may enter the name or alias or IP address, if known, of a party to be called by either manually entering the name, by using the speeddial feature, or by double clicking on an entry in a directory stored in the

memory, such as the memory 16 of the first processing unit 12, where the directory entries may be scrolled using the status area 38 and the down arrow icon 40.

Once a called party is listed in the status area 38 as being active on a line, the user may transfer the called party to another line or a conference line by clicking and dragging the status area 38, which is represented by a reduced icon 46. Dragging the reduced icon 46 to any one of line icons L1-L4 transfers the called party in use to the selected line, and dragging the reduced icon 46 to any one of conference line icons C1-C3 adds the called party to the selected conference call.

Other features may be supported, such as icons 48-52, where icon 48 corresponds to, for example, an ALT-X command to exit the communication facility of a processing unit, and icon 50 corresponds to, for example, an ALT-M command to minimize or maximize the display screen 36 by the output device of the processing unit. Icon 52 corresponds to an OPEN command, which may, for example, correspond to pressing the O key on a keyboard, to expand or contract the display screen 36 to represent the opening and closing of a cellular telephone. An "opened" configuration is shown in FIG. 5, and a "closed" configuration is shown in FIG. 6. In the "opened" configuration, additional features such as output volume (VOL) controls, input microphone (MIC) controls, waveform (WAV) sound controls, etc.

The use of display screens such as those shown in FIGS. 5-6 provided flexibility in implementing various features available to the user. It is to be understood that additional features such as those known in the art may be supported by the processing units 12, 22.

Alternatively, it is to be understood that one skilled in the art may implement the processing units 12, 22 to have the features of the display screens in FIGS. 5-6 in hardware; i.e. a wired telephone or wireless

cellular telephone may include various keys, LEDs, liquid crystal displays (LCDs), and touchscreen actuators corresponding to the icons and features shown in FIGS. 5-6. In addition, a PC may have the keys of a keyboard and mouse mapped to the icons and features shown in FIGS. 5-6.

Referring to FIG. 7, the disclosed point-to-point Internet protocol and system 10 is illustrated. First processing unit 12 initiates the point-to-point Internet protocol in step 56 by sending a query from the first processing unit 12 to the connection server 26. If connection server 26 is operative to perform the point-to-point Internet protocol, in step 58, first processing unit 12 receives an on-line status signal from the connection server 26, such signal may include the IP address of the callee or a "Callee Off-Line" message. Next, first processing unit 12 performs the primary point-to-point Internet protocol in step 60, which may include receiving, at the first processing unit 12, the IP address of the callee if the callee is active and on-line. Alternatively, processing unit 60 may initiate and perform the secondary point-to-point Internet protocol in step 62, if the called party is not active and/or on-line.

Referring to FIG. 8, in conjunction with FIGS. 1 and 3-4, the disclosed point-to-point Internet protocol and system 10 is illustrated. Connection server 26 starts the point-to-point Internet protocol, in step 64, and timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66. Connection server 26 receives a query from a first processing unit 12 in step 68 to determine whether a second user or second processing unit 22 is logged-in to the Internet 24, with the second user being specified, for example, by an E-mail address. Connection server 26 retrieves the IP address of the specified user from the database 34 in step 70, if the specified user is logged-in to the Internet, and sends the retrieved IP address to the first

processing unit 12 in step 72 to enable first processing unit 12 to establish point-to-point communications with the specified second user.

The disclosed secondary point-to-point Internet protocol operates as shown in FIG. 9. First processing unit 12 generates an E-mail signal, including a session number and a first IP address corresponding to a first processing unit in step 76. First processing unit 12 transmits the E-mail signal as a <ConnectRequest> signal to the Internet 24 in step 78. The E-mail signal is delivered through the Internet 24 using a mail server 28 to the second processing unit 22 in step 80. Second processing unit 22 extracts the session number and the first IP address from the E-mail signal in step 82 and transmits or sends the session number and a second IP address corresponding to the second processing unit 22, back to the first processing unit 12 through the Internet 24, in step 84. First processing unit 12 verifies the session number received from the second processing unit 22 in step 86, and establishes a point-to-point Internet communication link between the first processing unit 12 and second processing unit 22 using the first and second IP addresses in step 88.

While the disclosed point-to-point Internet protocols and system have been particularly shown and described with reference to the preferred embodiments, it is understood by those skilled in the art that various modifications in form and detail may be made therein without departing from the scope and spirit of the invention. Accordingly, modifications such as those suggested above, but not limited thereto, are to be considered within the scope of the invention.

Throughout this specification and the claims which follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

The reference to any prior art in this specification is not, and should not be taken as, an acknowledgment or any form of suggestion that that prior art forms part of the common general knowledge in Australia.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. Apparatus for establishing a point-to-point communication link said apparatus operating in a computer system operatively coupled to other computers and a server over a computer network, said apparatus comprising:

a. means for transmitting, from the first process to a server a query as to whether a second process is connected to the computer network;

b. means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network; and

c. means, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

2. The computer apparatus of claim 1 further comprising:

d. means for receiving audio data and transmitting the audio data to the second processor over the established point-to-point communication link.

3. An apparatus for use with a computer system capable of executing a first process and communicating with other processes, a directory server process and a mail server process over a computer network, the apparatus comprising:

A. program logic configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;

B. program logic configured to establish a communication connection with the directory server process once the assigned network protocol of the first process is known;

C. program logic configured to forward the assigned network protocol address of the first process to the directory server process upon establishing a communication connection with the directory server process; and

D. program logic configured to establish a point-to-point communication with another process over the computer network.

4. The apparatus of claim 3 wherein the program logic D further comprises:

D.1 program logic configured to transmit, from the first process to the directory server process, a query as to whether a second process is connected to the computer network; and

D.2 program logic configured to receive a network protocol address of the second process from the directory server process, when the second process is connected to the computer network.

5. The apparatus of claim 3 wherein the program logic D further comprises:

D.1 program logic configured to transmit an E-mail message containing a network protocol address from the first process to the mail server process over the computer network;

D.2 program logic configured to receive a second network protocol address from a second process over the computer network.

6. An apparatus for use with a computer system, the computer system capable of executing a first process connectable over a computer network to a second process and a directory database server process, the apparatus comprising:

a. program logic configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. program logic responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.

7. A computer program product for use with a computer system, the computer system executing a first process operatively connectable over a computer network to a second process and a server process, the computer program product comprising a computer useable medium having computer readable program code embodied therein, the program code comprising:

a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. program code responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.

8. A method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

A. following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network;

B. querying the address server as to whether the second process is connected to the computer network;

C. receiving a network protocol address of the second process from the address server, when the second process is connected to the computer network; and

D. in response to the network protocol address of the second process, establishing a point-to-point communication link with the second process over the computer network.

9. A method for establishing point-to-point communications with other processes, said method for use in a computer system capable of executing a first process and communicating with other processes and a server process over a computer network, said method comprising:

A. determining the currently assigned network protocol address of the first process upon connection to the computer network;

B. establishing a communication connection with the server process once the assigned network protocol of the first process is known;

C. forwarding the assigned network protocol address of the first process to the server process upon establishing a communication connection with the server process; and

D. establishing a point-tc-point communication with another process over the computer network.

10. The method of claim 9 wherein the program step D comprises:

D.1 transmitting, from the first process to the server process, a query as to whether a second process is connected to the computer network; and

D.2 receiving a network protocol address of the second process from the server process, when the second process is connected to the computer network.

11. A method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

a. accessing a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. in response to one of the network protocol addresses, establishing a point-to-point communication link from the first process to the second process over the computer network.

12. A method of establishing a point-to-point communication between a first process and a second process, said method for use in a first computer process operatively coupled over a computer network to a second process and an address server, said method comprising:

a. transmitting to the server a network protocol address received by the first process following connection to the computer network;

b. transmitting, to the server, a query as to whether the second process is connected to the computer network;

c. receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

d. in response to the network protocol address of the second process, establishing a point-to-point communication link between the first process and the second process over the computer network.

13. An apparatus capable of executing a first process and connecting to other processes and a server process over a computer network, the apparatus comprising:

a. program logic configured to generate a user-interface enabling control of a first process executing on the computer system;

b. program logic configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;

c. program logic, responsive to the currently assigned network protocol address of the first process, and configured to establish a communication connection with the server process and to forward the assigned network protocol address of the first process to the server process upon establishing a communication connection with the server process; and

d. program logic, responsive to user input commands, and configured to establish a point-to-point communications with another process over the computer network.

14. The apparatus of claim 13 wherein the program logic configured to establish a point-to-point communication link further comprises:

d.1 program logic, responsive to the network protocol address of a second process, and configured to establish a point-to-point communication link between the first process and the second process over the computer network.

15. The apparatus of claim 14 wherein the program logic configured to establish a point-to-point communication link further comprise:

d.2 program logic configured to transmit, from the first process to the server process, a query as to whether the second process is connected to the computer network; and

d.3 program logic configured to receive a network protocol address of the second process from the server process, when the second process is connected to the computer network.

16. An apparatus for use with a computer system, the computer system executing a first process operatively coupled over a computer network to a second process and a directory database server process, the apparatus comprising:

A. program logic configured to, following connection of the first process to the computer network, forward to the address server a network protocol address at which the first process is connected to the computer network;

B. program logic configured to query the address server as to whether the second process is connected to the computer network;

C. program logic configured to receive a network protocol address of the second process from the address server, when the second process is connected to the computer network; and

D. program logic configured to, in response to the network protocol address of the second process, establish a point-to-point communication link with the second process over the computer network.

17. Apparatus for establishing a point-to-point communication link, said apparatus operating in a computer system capable of executing a first process and operatively connectable to other processes and a server over a computer network, said apparatus comprising:

a. means for transmitting, from the first process to the server, a query as to whether a second process is connected to the computer network;

b. means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network; and

c. means, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

18. A computer data signal embodied in a carrier wave comprising:

program code for transmitting to a server a network protocol address received by a first process following connection to a computer network;

program code for transmitting, to the server, a query as to whether a second process is connected to the computer network;

program code for receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

19. A computer program product for use with a computer system, the computer system executing a first process and operatively connectable to a second process and a server over a computer network, the computer program product comprising a computer useable medium having program code embodied in the medium, the program code comprising:

program code for transmitting to the server a network protocol address received by the first process following connection to the computer network;

program code for transmitting, to the server, a query as to whether the second process is connected to the computer network;

program code for receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and

program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

20. Apparatus substantially as hereinbefore described with reference to the accompanying drawings.

21. A computer program product substantially as hereinbefore described with reference to the accompanying drawings.

22. A method substantially as hereinbefore described with reference to the accompanying drawings.

23. A computer data signal substantially as hereinbefore described with reference to the accompanying drawings.

DATED this 8th day of September 2000

NetSpeak Corporation
By its Patent Attorneys
DAVIES COLLISON CAVE

13 09 00 59379

1/6

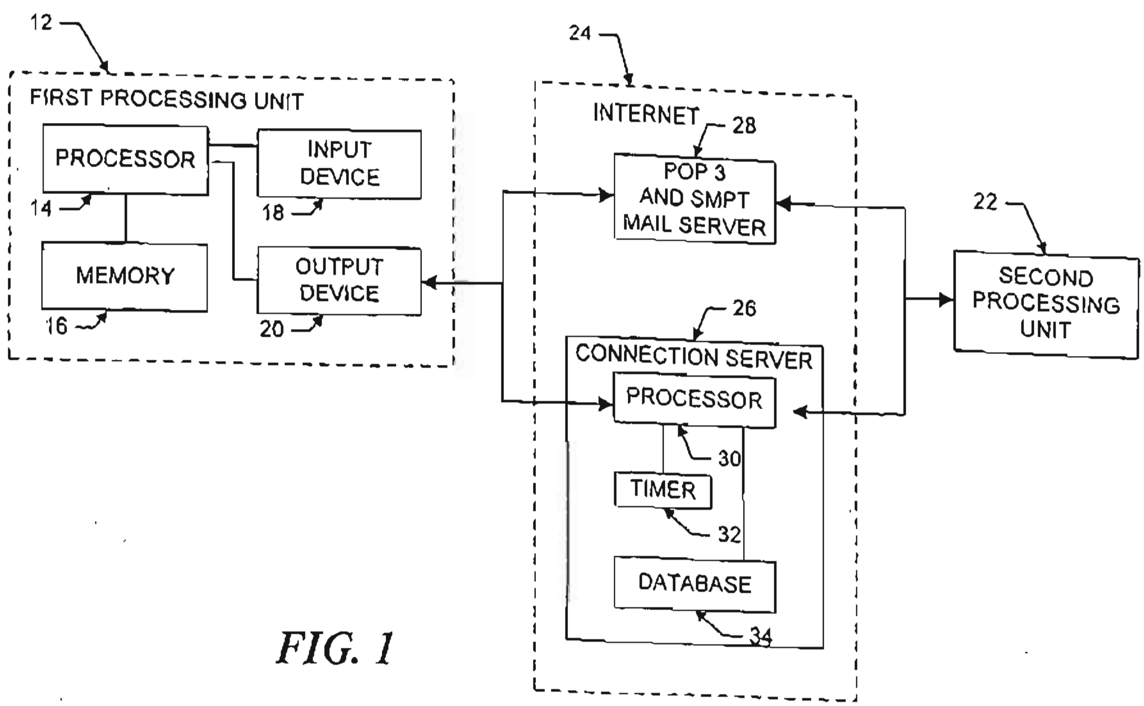


FIG. 1

13 09 00 59379

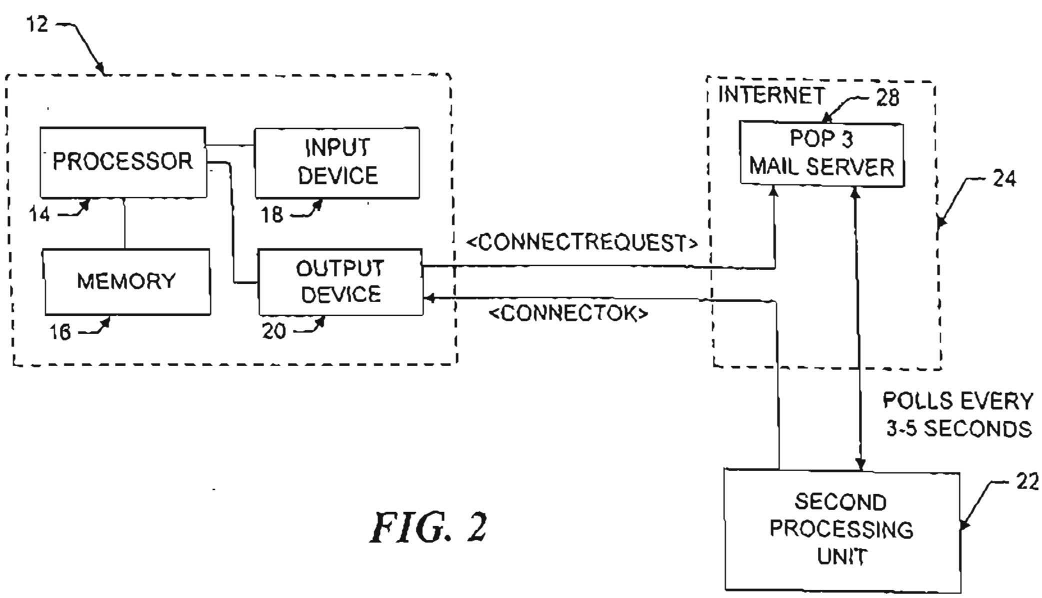


FIG. 2

13 09 00 59379

3/6

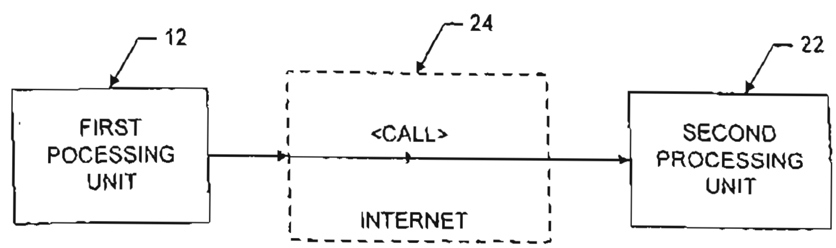


FIG. 3

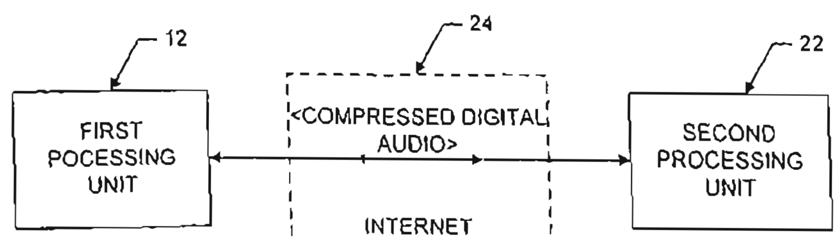


FIG. 4

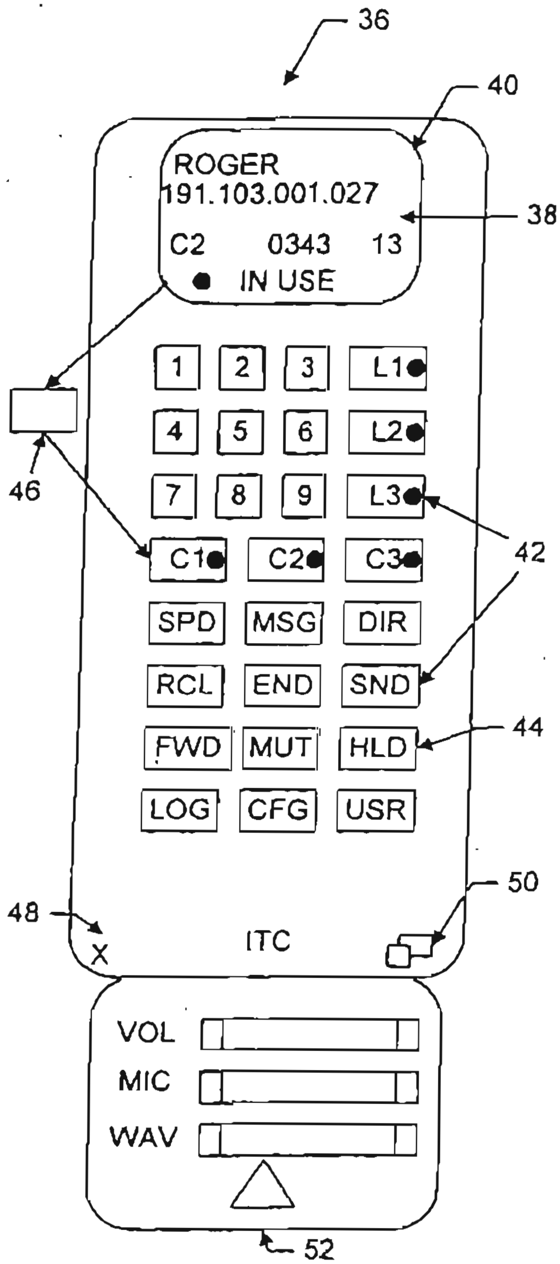

36
40
ROGER
191.103.001.027
38
C2 0343 13
IN USE
1 2 3 L1
4 5 6 L2
7 8 9 L3
46
C1 C2 C3
42
SPD MSG DIR
RCL END SND
FWD MUT HLD
44
LOG CFG USR
50
48
X
ITC
VOL
MIC
WAV
52

FIG. 5

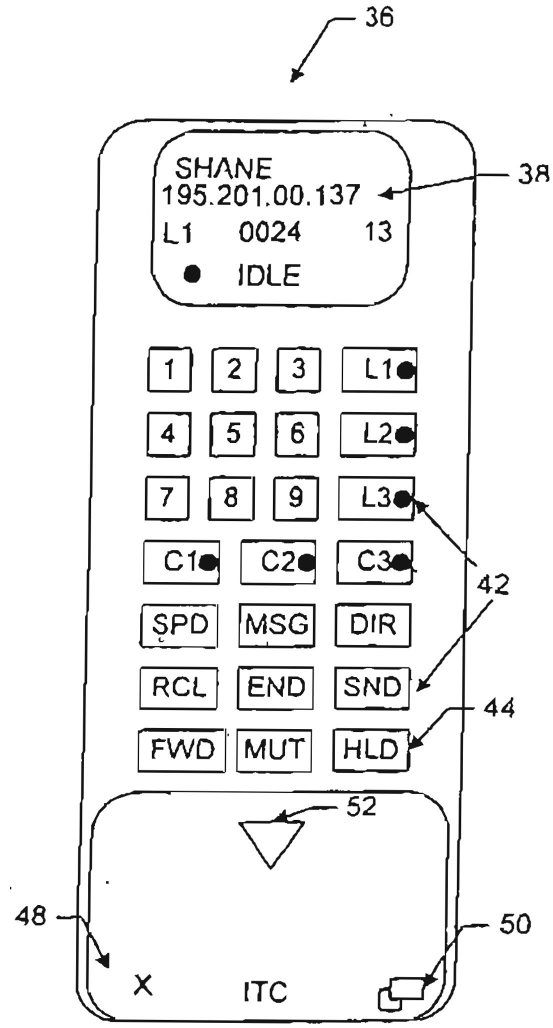

36
SHANE
195.201.00.137
38
L1 0024 13
IDLE
1 2 3 L1
4 5 6 L2
7 8 9 L3
C1 C2 C3
42
SPD MSG DIR
RCL END SND
44
FWD MUT HLD
52
48
50
X
ITC

FIG. 6

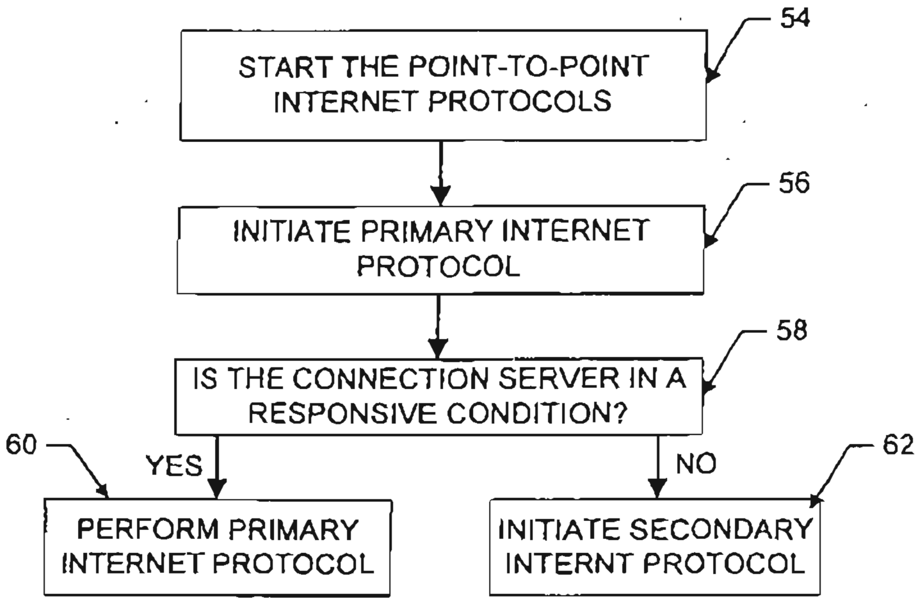

54
START THE POINT-TO-POINT INTERNET PROTOCOLS
56
INITIATE PRIMARY INTERNET PROTOCOL
58
IS THE CONNECTION SERVER IN A RESPONSIVE CONDITION?
60
YES
NO
62
PERFORM PRIMARY INTERNET PROTOCOL
INITIATE SECONDARY INTERNT PROTOCOL

FIG. 7

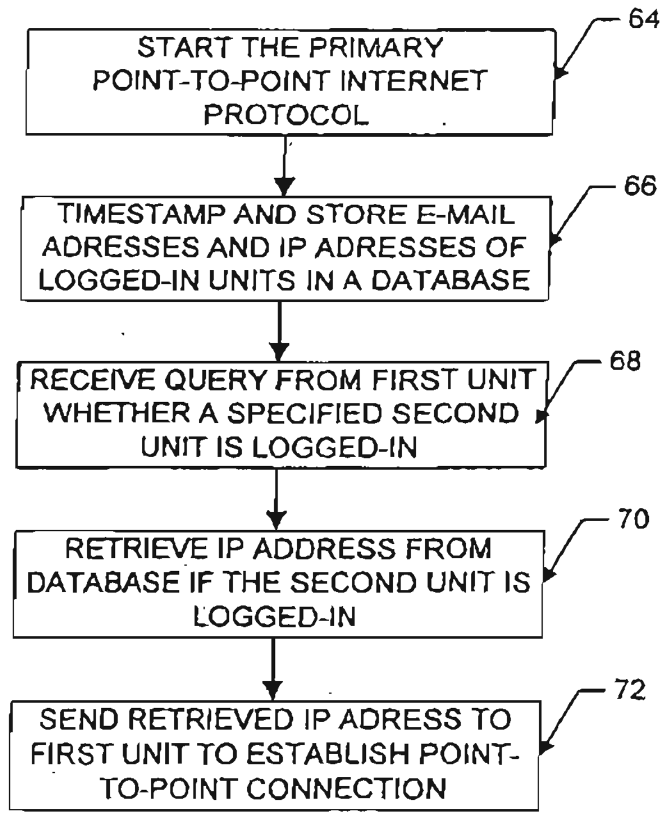

64
START THE PRIMARY POINT-TO-POINT INTERNET PROTOCOL
66
TIMESTAMP AND STORE E-MAIL ADRESSES AND IP ADRESSES OF LOGGED-IN UNITS IN A DATABASE
68
RECEIVE QUERY FROM FIRST UNIT WHETHER A SPECIFIED SECOND UNIT IS LOGGED-IN
70
RETRIEVE IP ADDRESS FROM DATABASE IF THE SECOND UNIT IS LOGGED-IN
72
SEND RETRIEVED IP ADRESS TO FIRST UNIT TO ESTABLISH POINT-TO-POINT CONNECTION

FIG. 8

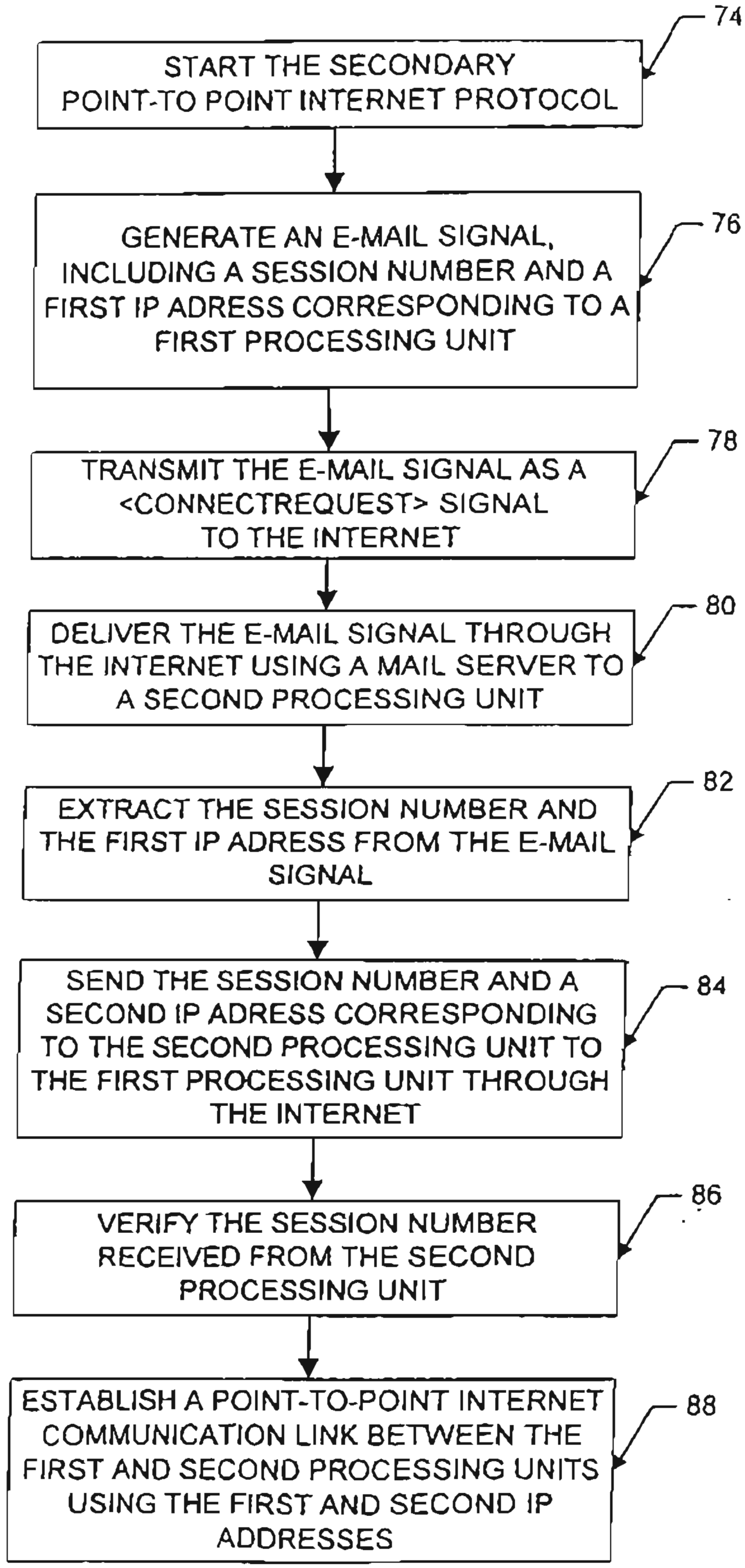

START THE SECONDARY POINT-TO POINT INTERNET PROTOCOL
74
GENERATE AN E-MAIL SIGNAL, INCLUDING A SESSION NUMBER AND A FIRST IP ADRESS CORRESPONDING TO A FIRST PROCESSING UNIT
76
TRANSMIT THE E-MAIL SIGNAL AS A <CONNECTREQUEST> SIGNAL TO THE INTERNET
78
DELIVER THE E-MAIL SIGNAL THROUGH THE INTERNET USING A MAIL SERVER TO A SECOND PROCESSING UNIT
80
EXTRACT THE SESSION NUMBER AND THE FIRST IP ADRESS FROM THE E-MAIL SIGNAL
82
SEND THE SESSION NUMBER AND A SECOND IP ADRESS CORRESPONDING TO THE SECOND PROCESSING UNIT TO THE FIRST PROCESSING UNIT THROUGH THE INTERNET
84
VERIFY THE SESSION NUMBER RECEIVED FROM THE SECOND PROCESSING UNIT
86
ESTABLISH A POINT-TO-POINT INTERNET COMMUNICATION LINK BETWEEN THE FIRST AND SECOND PROCESSING UNITS USING THE FIRST AND SECOND IP ADDRESSES
88

FIG. 9

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 518 596 A1**

(12) EUROPEAN PATENT APPLICATION

(21) Application number : 92305234.4

(22) Date of filing : 08.06.92

(51) Int. Cl.[5] : **H04L 12/46**

(30) Priority : 14.06.91 US 716041

(43) Date of publication of application : 16.12.92 Bulletin 92/51

(84) Designated Contracting States : DE FR GB IT

(71) Applicant : DIGITAL EQUIPMENT CORPORATION
146 Main Street
Maynard, Massachusetts 01745 (US)

(72) Inventor : Perlman, Radia J.
10 Huckleberry Lane
Acton, Massachusetts 01720 (US)

(74) Representative : Goodman, Christopher et al
Eric Potter & Clarkson St. Mary's Court St. Mary's Gate
Nottingham NG1 1LE (GB)

(54) Router using remote address resolution to enable bridge like data forwarding.

(57) A communications system has a first communications link (110) and a second communications link (112), at least one end station (111,113) capable of communicating on each of the communications links, an apparatus (100) for forwarding a packet from the first link to the second link, the apparatus capable of detecting a network layer header on a data packet, the network layer header having a destination address. There is assigned, to the apparatus, an apparatus mask having a forwarding mask length for distinguishing the destination address into a subnet address part and into a host address part. Also there is assigned, to an end station, an end station mask having an end station mask length for distinguishing the destination address into a subnet address part and into a host address part. And there is assigned a greater length to the forwarding mask length than to the end station mask length, to enable the end station in using the end station mask to identify all end stations on the first link and the second link as being on a single link, and to enable the apparatus in using the forwarding mask to distinguish which of the first link or the second link an end station addressed by the network layer address is located.

A selected end station transmits an Address Request Protocol (hereinafter ARP) message onto the first communications link, the ARP message requesting a data link address from a receiving end station, and the apparatus sends the ARP message to a second forwarding apparatus connected to the second link having the receiving end station connected thereto. The apparatus receives an ARP response message containing the data link address of the receiving end station from the second forwarding apparatus. And finally, the apparatus forwards the ARP response message to the selected end station. The forwarding apparatus forwards a data packet as a bridge in the event that a data link address in a message packet is not a data link address of the apparatus. The forwarding apparatus forwards a data packet as a router in the event that a data link address in the message packet is a data link address of the apparatus.

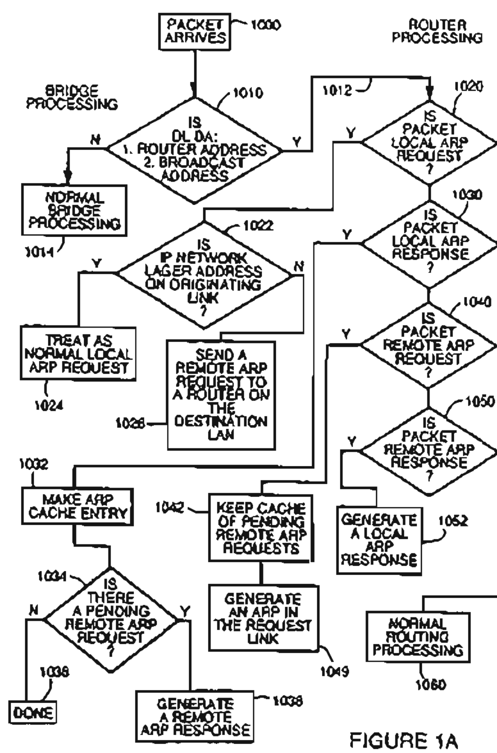


FIGURE 1A

EP 0 518 596 A1

Jouve, 18, rue Saint-Denis, 75001

FIELD OF THE INVENTION

This invention relates to forwarding messages from a first link to another link, and more particularly relates to reducing the time required to forward data packets.

BACKGROUND OF THE INVENTION

Communications systems between computers are presently capable of connecting tens of thousands of computers. Typically, a computer will originate a message directed to another computer, and will transmit the message as a sequence of data packets onto the communications system. Because of the large number of computers connected to the system, and the large number of data packets transferred between the computers, throughput of packets transferred by the system is an important issue.

Communications systems are often divided into a number of links. Typically, a link may be a local area network, where each local area network is capable of supporting a few hundred computers. A local area network will hereinafter be referred to as a LAN. The LANs are connected together by a number of different standard devices which forward packets. With the increasingly large size of modern communications systems, the time required to forward a data packet between LANs becomes an important parameter of system design.

Other types of links in a communications system may be, for example, a wide area network formed by joining other links such as LANs, a point to point connection between computers, etc. Maintaining high throughput of system packet traffic is also an important problem in all link to link connections.

Also, all types of links may be connected together by standard devices.

Before discussing standard devices used to connect links together, data packets and the headers of data packets added by different layers of the communications protocol must be discussed. A data packet is typically formed in a higher level of the communications protocol, and finally is transferred down to the Transport Layer which passes the packet into the Network layer. The Network layer attaches a header, the Network Layer Header, to the data packet, and then passes the packet into the Data Link Layer. The Data Link Layer then attaches a header, the Data Link Layer Header, to the data packet. The packet is then transmitted onto the communications system by the physical layer.

A packet, once transmitted onto the communications system, is then forwarded from link to link until it reaches its destination end station.

A first type of device connecting links of the communications system is a bridge. A bridge operates in the Data Link level of the communications protocol, which is the level immediately above the physical level. A bridge receives data packets from one link, typically a LAN, and then parses the Data Link Header. The bridge then makes a decision on what to do with the data packet, where the decision is based upon the contents found in the Data Link Header.

A second type of device linking LANs is a router. A router operates in the network layer, a layer above the data link layer'. A router operates by parsing both the Data Link Header and the Network Layer Header, and making decisions based on the contents of both headers.

In some designs a bridge may be on the order of 200 times faster than a router in forwarding a data packet from a first link to a second link.

Even though a router is slower in forwarding packets from one link, such as a LAN, to another link, it is necessary to use routers rather than bridges at certain locations between multiple numbers of links. The router performs functions beyond those of a bridge, such as: forwarding along better routes than a bridge; incrementing a "hop count" field of a forwarded packet to show the number of passes of the packet through a router in order to prevent indefinite looping of the packet; preventing certain management traffic such as "hello" messages from end stations on one link from being forwarded to the other link; maintaining "network layer addresses" of stations on the links that it connects; fragmentation and reassembly of packets because of different protocols employed by different links; performing explicit handshaking protocols with end stations connected to links connected to the router; participating in routing algorithms, and other functions.

However, a difficulty in operation of large computer communications networks is that the time required for a router to forward messages may result in lower throughput.

SUMMARY OF THE INVENTION

The invention resides in an apparatus for forwarding packets, and solves the difficulty of a router requiring too much time for forwarding a packet, along with the difficulty of a bridge increasing address request protocol (ARP) traffic by forwarding ARP messages.

A communications system has a first communications link and a second communications link, at least one end station capable of communicating on each of the communications links, an apparatus for forwarding a packet from the first link to the second link, the apparatus capable of detecting a network layer header on a data packet, the network layer header having a destination address. There is assigned, to the apparatus, an apparatus mask having a forwarding mask length for distinguishing the destination address into a subnet address part and into a host address part. Also there is assigned, to an end station, an end station mask having an end station mask length for distinguishing the destination address into a subnet address part and into a host address part. And there is assigned a greater length to the forwarding mask length than to the end station mask length, to enable the end station in using the end station mask to identify all end stations on the first link and the second link as being on a single link, and to enable the apparatus in using the forwarding mask to distinguish which of the first link or the second link an end station addressed by the network layer address is located.

Also there is a first means for a selected end station to transmit a local Address Request Protocol (hereinafter ARP) request message onto the first communications link, the local ARP request message requesting a data link address of a receiving end station. Further, there is a second means, in response to the local ARP request message, for the apparatus to create a remote ARP request message and to send the remote ARP request message to a second forwarding apparatus connected to the second link having the receiving end station connected thereto. There is also a third means for the apparatus to receive a remote ARP response message containing the data link address of the receiving end station from the second forwarding apparatus. And there is a fourth means, responsive to the remote ARP request message, for the apparatus to create a local ARP response message and to send the local ARP response message to the selected end station.

The forwarding apparatus forwards a data packet as a bridge in the event that a data link address in a message packet is not a data link address of the apparatus.

The forwarding apparatus forwards a data packet as a router in the event that a data link address in the message packet is a data link address of the apparatus.

BRIEF DESCRIPTION OF THE DRAWINGS

A more detailed understanding of the invention may be had from the following description of preferred exemplary embodiments to be understood in conjunction with the accompanying drawing wherein:

Fig. 1 is a logic diagram of two LANs connected by a router in accordance with a preferred embodiment of the invention.

Fig. 1A is a flow chart of a brox in accordance with the invention.

Fig. 2 is a field diagram of a data packet in accordance with a preferred embodiment of the invention.

Fig. 3A is field diagram of an address field.

Fig. 3B is a field diagram of a mask for an address field.

Fig. 3C is is a field diagram of a mask for an address field.

Fig. 4 is a block diagram of a communications system having multiple links, in accordance with the invention.

Fig. 5 is a field diagram of an address in accordance with the invention.

Fig. 6A is a field diagram of an address in accordance with a preferred embodiment of the invention.

Fig. 6B is a field diagram of a bridge router box (brox) mask in accordance with a preferred embodiment of the invention.

Fig. 7A is a field diagram of an address in accordance with a preferred embodiment of the invention.

Fig. 7B is a field diagram of an end station mask in accordance with a preferred embodiment of the invention.

Fig. 8 is a block diagram of a communications system in accordance with the invention.

DETAILED DESCRIPTION

FIRST EXEMPLARY EMBODIMENT

Capitalization will be used in this document to highlight names of fields of a packet, in order to improve readability of the document.

Referring now to Fig. 1, there is shown a communications link connection apparatus 100, a "brox", in accordance with the invention. The term "brox" is coined herein, and is defined as a box for connecting communications links in accordance with the present invention. A brox forwards as a router under certain conditions, but under other conditions forwards as a bridge. The word "brox" is an acronym (created from the capitalized letters Bridge Router bOX) for a box that behaves as a bridge or as a router. Also a brox may have behavior modes that are neither those of a standard bridge nor those of a standard router.

For a multiple hop data packet transmission, operation of the invention can be described in simple terms for a network using a Transmission Control Protocol-Internet Protocol (TCP-IP) type protocol as follows. TCP-IP is a well known protocol suite which has been developed in the United States. The present invention works partirularly well with the TCP-IP protocol, but will also work with similar protocols.

But first, a TCP-IP type of protocol will be briefly described. In a TCP-IP type protocol, a destination address in the Network Layer Header of a packet is resolved into a subnet address and a host address by applying a mask to the total address. A mask is assigned to a link. A station is configured with a network layer address and a mask for each link to which the station is attached. The mask has as many bit positions as the address in the packet.

The mask has some bits set equal to "1", and some bits set equal to "0". The bits of the mask that equal "1" correspond to bits of the address that identify the link. The bits of the mask that equal "0" correspond to bits of the address that identify the host, or end station, on that link.

For example: in the event that the network layer address is 32 bits (or 4 octets), then the mask is also 32 bits; in the further event that the mask has 16 bits set =1, and 16 bits set =0, then the station interprets the bits in the address corresponding to the bits =1 of the mask as identifying the link. The bits in the address corresponding to "0s" in the mask identify the host, or end station, on that link.

As further example, if the first two octets of the mask contain 8 bits =1 each, and the second two octets of the mask contain all bits =0, then the address of the link is contained in the two octets of the address corresponding to the bits =1 of the mask. Also, the host, or station, address on that link is contained in the address bits corresponding to the bits =0 of the mask.

Terminology used in the art of computer communications will now be addressed. The terms "link" and "subnet" are substantially synonyms. Also, the terms "host" and "end station" are substantially synonyms. The terms "link" and "end station" are frequently used in discussions of OSI standard type computer communications systems. The terms "subnet" and "host" are frequently used in discussions of TCP-IP type computer communications systems. A link or a subnet may be a local area network, or may be another type of system for computer communications. A host or end station is the computer attached to the link or the subnet.

A further standard feature of a TCP-IP type protocol is that when an initiating station initiates a transmission to an intended receiving station, the initiating station "knows" only the Network Layer address of the intended receiving station. In order to "learn" the Data Link Layer address of the intended receiving station, the initiating station transmits an Address Request Protocol (ARP) message. A bridge forwards the ARP message, but a router ignores the ARP message. In the event that the intended receiving station receives the ARP message, either by being on the same link as the station or by having the ARP request bridged to the link containing the intended receiving station, then the intended receiving station sends to the initiating station an ARP response containing the Data Link Layer address of the intended receiving station. Upon receipt of the ARP response, the initiating station sends data packets to the intended receiving station by placing the Data Link Layer address learned from the ARP response into the Data Link Header of the data packets.

Also, as a standard practice in a TCP-IP type protocol, an initiating station first tests the Network Layer address of the intended receiving station against its own Network Layer address. Each station on the communications link of the initiating station has the same subnet address, and so the initiating station tests the subnet address of the intended receiving station against the subnet address its own subnet address.

In the event that the two subnet addresses are equal, the initiating station transmits an ARP message.

In the event that the two subnet addresses are different, then the initiating station transmits a data packet to a router connected to the communications link of the initiating station. The router maintains a forwarding table, and so "knows" the communications link containing the intended receiving station. The router then forwards the data packet to the proper link, where a further router places the Data Link Layer address of the intended receiving station into the Data Link Layer Header of the data packet, and forwards the data packet onto the proper link for receipt by the intended receiving station. The originating station does not learn the data link address of the receiving station.

Turning now to a simple discussion of the invention, both the router and the bridge are replaced with the "brox" of the invention. Also, in the invention, the network layer addresses of the end stations on the various links are carefully chosen so that all end stations on all links connected by the invention have in their network layer address: firstly, a sequence of bits that are identical for all end stations attached to any of the links connected by broxs; secondly, a sequence of bits that are identical for stations on a single link, but different for stations on other links; and, thirdly, a sequence of bits that are unique among stations on any one link.

The forwarding apparatus of the invention is the brox. A brox has assigned a mask that exposes both the sequence of identical bits and the sequence of unique bits. In contrast, the end stations have a shorter mask that exposes only the sequence of identical bits.

All end stations linked by the invention have the same sequence of identical bits. Accordingly, when an initiating station tests the network layer address of the intended receiving station against its own network layer address in order to determine if the initiating station has permission to send a local ARP message, then permission to send a local ARP message will be granted for an intended end station linked by the invention, whether the intended end station is on the same link as the initiating station or whether it is on a different link.

Also in the invention it is important to distinguish four (4) different types of ARP messages. These four different types of ARP messages are:

1. a local ARP request. A local ARP request is transmitted by an end station onto a link connected to the end station. A local AR request asks for a data link address of an intended destination station. A local ARP request is identical to the standard TCP-IP compatible ARP request discussed hereinabove.

2. a local ARP response. A local ARP response

is received by an end station, and the local ARP response delivers a data link address to the end station. A local ARP response is identical to the standard TCP-IP ARP response discussed hereinabove.

3. a remote ARP request. A remote ARP request is created by a brox on the link of an originating end station, and is created by the brox in response to receipt by the brox of a local ARP request. The remote ARP request is transmitted to a remote brox on the link of a remote intended destination station.

4. a remote ARP response. A remote ARP response is created by a remote brox on the link of a remote intended destination station, and is created in response to the remote brox receiving a local ARP response. The remote ARP response is transmitted to a brox on the link of the originating end station, and in response to receiving the remote ARP request, the brox creates and transmits a local ARP response to the originating end station.

By carefully choosing the addresses of all end stations linked by the invention, by using broxs for forwarding between the various links, and by using short masks for stations and long masks for broxs, useful results flow from the invention. The first useful result is that data packets are forwarded at bridge speed between end stations linked by the invention. A second useful result is that local ARP traffic on one link is confined to that link and is not forwarded as it would be if the links were joined by a conventional bridge. A third useful result is that all links joined by the invention may be reached by remote ARP messages generated in response to any end station on any of the links, without flooding every link with unnecessary local ARP messages.

Forwarding rules followed by a brox may be summarized as follows:

1. in the event that the brox recognizes the address found by parsing the Data Link Destination Address field of a packet as an address used by that brox, then the brox receives the packet and functions as a router.

2. in the event that the brox does not recognize the destination address found by parsing the Data Link header as an address of the brox, then the brox bridges the packet.

3. in the event that the brox recognizes the packet as a local ARP request, and the intended destination end station is on a remote link as determined by the brox using its long mask, then the brox creates a remote ARP request and sends it to a brox on the link of the intended destination end station.

4. later the brox receives a remote ARP response from a brox attached to the link of the intended destination end station. The brox then creates a local ARP response in response to the original local ARP request. The local ARP response contains the data link layer address of the intended receiving end station. The brox then transmits the local ARP response onto the link having the originating end station. The local ARP response is received by the originating end station, and so the originating end station learns the data link address of the intended destination station.

A significant benefit of the invention is that, once the originating end station learns the data link address of the intended destination end station, forwarding of later sent data packets is at bridge speed, rather than router speed. The forwarding delay at each brox, in some designs, may be as much as 200 times less when the brox functions as a bridge rather than as a router. Accordingly, the invention greatly speeds forwarding of data packets.

Referring now to Fig. 2, there is shown a typical field structure for a data packet used by an end station of LAN 110, 112 shown in Fig. 1. Data packet 120 is shown having a Data Link Header 122 and a Network Layer Header 124. When data packet 120 is created and transmitted onto a LAN, the network layer attaches Network Header 124 to the packet, and then the packet is handed down to the data link layer. The data link layer then attaches the Data Link Header 122 to the packet. Upon transmission, data packet 120 may have additional fields preceding the Data Link Header 122 such as, for example, preamble fields, and the precise structure of such preamble fields will depend upon the standard to which LAN 110 is designed. Such preamble fields are not shown in Fig. 2, as Fig. 2 focuses on those fields used by the invention.

Data Link Header 122 contains Data Link Destination Address field 126, and Data Link Source Address field 128. Other Data Link Header fields 130 are also shown in Fig. 2, but are not further described herein in that the invention focuses on the Data Link Destination Address field 126 and the Data Link Source Address field 128 of the Data Link Header 122. Data Link Destination Address field 126 is abbreviated DL D. Data Link Source Address field 128 is abbreviated DL S.

Network Layer Header 124, as shown in Fig. 2, has: Network Layer Destination address field 140, NL D field 140; Network Layer Source address field 142 NL S; and other fields 144. Also, data fields 146 follows the Network Layer Header 124. The Network Layer Destination Address field 140 is abbreviated NL D. The Network Layer Source address field 142 is abbreviated NL S.

In exemplary network designs, the Network Layer Destination address field 140 and the Network Source Layer field 142 may each be assigned a fixed length. The length is often expressed in terms of octets. An octet is a data structure that is ordinarily 8 bits long. For example, in a TCP-IP compatible network both the Network Layer Destination Address field 140 and the Network Layer Source Address field 142 are assigned a length of 32 bits, or 4 octets each.

Turning now to Fig. 1A, there is shown a flow chart of the logic of an exemplary brox 100. At block 1000 the packet arrives at the brox. Control passes to decision block 1010 where the packet data link destination address contained in the Data Link Destination Address field 126 of the packet is determined. Then, the packet data link destination address is tested in order to determine if it is either a router address or a broadcast address used by a router. In the event that decision block 1010 answers "yes", the packet data link destination address is either a router address or a broadcast address, then control passes to line 1012 for router type processing.

In the event that decision block 1010 answers "no", then control passes to block 1014 where the packet is further processed by normal bridge forwarding.

Router processing proceeds along line 1012 to a series of decision blocks, block 1020, block 1030, block 1040, and block 1050. At these decision blocks the type of packet is determined.

At block 1020 the packet is tested in order to determine if it is a local ARP request, and in the event that it is, decision block 1020 answers "yes", and control passes to decision block 1022 for handling as a local ARP request. In the event that decision block 1020 answers "no", then control passes to decision block 1030.

At decision block 1030 the packet is tested in order to determine if it is a local ARP response, and if it is decision block 1030 answers "yes", then control passes to block 1032 for further processing as a local ARP response. In the event that decision block 1030 answers "no", then control passes to decision block 1040.

At decision block 1040 the packet is tested in order to determine if it is a remote ARP request, and if it is decision block 1040 answers "yes", then control passes to block 1042 for processing as a remote ARP request. In the event that decision block 1040 answers "no", then control passes to decision block 1050.

At decision block 1050 the packet is tested in order to determine if it is a remote ARP response, and if it is decision block 1050 answers "yes", then control passes to block 1052 for further processing as a remote ARP response. In the event that decision block 1050 answers "no", then control passes to block 1060.

At block 1060 the packet is forwarded to the proper link in accordance with normal routing processing.

We return to discuss the processing in the event that the packet tested with a "yes" at decision blocks 1020, 1030, 1040, and 1050.

First we discuss processing in the event that decision block 1020 answers "yes", the packet is a local ARP request, and control passes to decision block 1022. A branch to decision block 1022 means that a host on a link connected to the brox issued a local ARP request. At decision block 1022 the packet is tested in order to determine if the network layer address of the intended destination host is on the originating link, and if it is decision block 1022 answers "yes", then processing branches to block 1024. At block 1024 the packet is treated as a normal local ARP request.

In the event that decision block 1022 answers "no", the packet network layer destination address is not on the originating link, then control passes to block 1026. At block 1026 the brox sends a remote ARP request to a brox or router on the destination LAN.

Next we discuss processing in the event that the packet is a remote ARP request as determined by a "yes" response at decision block 1040, where control passed to block 1042. At block 1042 a cache entry is made to record that a pending remote ARP request has been received. Control then passes to block 1044 where a local ARP request is generated on the relevant link, where the link is connected to a port of the brox. Processing in blocks 1042 and 1044 means that the brox is the remote brox in a remote ARP request.

Next we discuss processing in the event that the packet is a local ARP response, that is that decision block 1030 answered "yes", and processing passed to block 1032. At block 1032 an entry is made into the cache of pending remote ARP requests, and control passes to decision block 1034. At decision block 1034 the question, "Is there a pending remote ARP request that matches this local ARP response ?" is asked, and in the event that the answer is "no", then control passes to block 1036 where processing of this packet stops because processing is done.

In the event that decision block 1034 answers "yes", there is a matching pending remote ARP request, then control passes to block 1038. At block 1038 the brox generates a remote ARP response message and sends the response to the originating brox.

Next we discuss processing in the event that decision block 1050 answered "yes", the packet is a remote ARP response, and control passes to block 1052. At block 1052 the brox generates a local ARP response and transmits it onto the relevant link connected to the brox. Processing at block 1052 means that the brox originally generated a remote ARP request, now the remote ARP response has arrived, and the remote ARP response is used to create a local ARP response directed to the originating host.

Normal operation of a bridge and normal operation of a router will now be described.

Bridges

The Data Link Header 122 contains a number of fields, and the fields principally used by the bridge are: the Data Link Destination Address field 126; and, the Data Link Source Address field 128 (Fig. 2).

The bridge compares the address found in the Data Link Destination Address field with a forwarding table maintained in a database contained in the bridge, and also compares the contents of the Data Link Source Address field of the packet with a list of source addresses maintained for each link connected to the bridge. The bridge then, typically makes forwarding decisions based upon the contents of these fields.

Typical design rules for operation of a bridge are as follows, and include both rules for receipt of a packet and rules for forwarding a packet.

For receipt of a packet, a bridge tests the contents of the Data Link Destination Address field of the packet against internally maintained forwarding tables.

For forwarding, the bridge decides what to do with the packet, for example, as follows. Typically, for a bridge that uses the flooding and backward learning algorithm, the bridge makes the following decisions, based upon the contents of the Data Link Destination Address field of the packet: if the packet Data Link Destination Address is in the forwarding table of a link attached to the bridge, then forward the packet to the proper link, except, if the packet destination is on the link from which the packet originated, then disregard the packet; and, if the destination address is not in the forwarding table, then flood the packet to all of the links connected to the bridge, but not the link from which the packet originated. Also, typically, a bridge forwards packets having certain multicast or broadcast addresses in their Data Link Destination Address field, such as end station hello messages. A bridge attempts to make the links that it joins together operate as an extended LAN.

Also, if the content of the Data Link Source Address field of the packet is absent from the bridge forwarding tables, then the bridge adds to its appropriate forwarding table a correlation between the address contained in the Data Link Source Address field of the packet and the link from which the packet arrived. Any subsequent packets addressed to that address are then forwarded onto the correlated link. By updating its forwarding table using the arrival link of unknown packets, a bridge learns the correlation between arrival links and the source address of end stations either on those links or connected to those links from other links, and thereby builds up entries into its forwarding tables. Further, for example, there are many other ways that entries in in a bridge forwarding table may be compiled.

Routers

A router receives a packet in the event that the Data Link Destination Address field DL D 126 of the packet contains the data link address of the router or a special multicast address used by routers, otherwise the router ignores the packet.

In the event that a router receives a packet, the router uses its mask to analyze the network layer destination address carried in the Network Layer Destination Address field NL D 140 of the packet. By using its mask, the router breaks the packet network layer destination address into a subnet, or link, part and into a host address part.

The router "knows" a route to the destination link as a result of a forwarding table maintained by the router. The forwarding table is built up by the router participating in router protocol algorithms. The router forwards the packet by placing an appropriate data link destination address in the Data Link Destination Address field DL D of the packet, and by transmitting the packet onto the proper link.

Routers perform other functions not directly related to the present invention, such as: running routing protocols in order to decide on routes to maintain between links when there are choices of multiple routes, that is, participating in routing algorithms; isolating links, by, for example, preventing certain management traffic such as end station hello messages on one link from being forwarded to another link; fragmentation and reassembly of packets because of different protocols employed by different links; performing explicit handshaking protocols with end stations connected to links connected to the router; and other functions.

Referring now to Fig. 3A, there is shown the structure of a Network Layer Address. Fig. 3A may refer to the address of a station on a link. Alternatively, Fig. 3A may refer to a Network Layer Destination Address field of a message packet. As a further alternative, Fig. 3A may refer to a Network Layer Source Address field of a message packet. For convenience, the field structure of Fig. 3A may be referred to as the structure of a Network Layer Destination (often abbreviated as NL D) address field of a message packet, although, as will be apparent to those skilled in the art, the discussion could equally well apply to the Network Layer Source Address field (often abbreviated as NL S address field) 142 of a message packet. Also, Fig. 3A will be used to describe a Network Layer Address assigned to a station.

As shown in Fig. 3A, NL D address field 140 contains 4 bytes, byte 150, byte 152, byte 154, and byte 156. Each byte 150 152 154 156 is an octet, and thereby contains 8 bits.

Referring now to Fig. 3B, there is shown mask 158. Mask 158 has 4 bytes, byte 160, byte 162, byte 164, and byte 166. A mask is assigned to a station, for example the mask is assigned to end station A 111A, or, for example, B 111B, or, for example, end station C 113C, or to brox 100. In the event that a station detects a message packet, the station applies mask 158 to NL D address field 140 in order to determine the address represented by the NL D address field 140. For example, in the example shown in Fig. 3A and Fig. 3B, the mask 158 is shown having all ones in byte 160, byte 162, and byte 164. In contrast, byte 166 contains all zeros. Accordingly, the corresponding bytes of NL D address field 140, that is byte 150 corresponding to byte 160 of the mask, byte 152 corresponding to byte 162 of the mask, and byte 154 corresponding to the mask byte 164 represent the subnet address. The term "subnet" is synonymous with "link", and may accordingly stand for a local area network LAN, or any other type of link. Byte 156 of NL D address field 140 corresponds to the zeros of mask byte 166, and accordingly represents the address of a "host". The term "host" is synonymous with "station", and so may stand for an "end station", a "brox", or any other type of station. Mask 158 indicates, by the zeros in byte 166, that the host address represented by NL D address field 140 is contained in byte 156.

Accordingly, mask 158 interprets NL D address field 140 so that bytes 150, 152, 154 represent a subnet address. And also mask 158 indicates that byte 156 represents a host address.

As shown in Fig. 3B, mask 158 has byte 160 indicated as byte B1. Byte 162 is indicated as byte B2. Byte 164 is indicated byte as B3. Byte 166 is indicated as byte B4. The labels B1, B2, B3, and B4 indicate the position of the byte within mask 158.

SECOND EXEMPLARY EMBODIMENT

Operation of communications system 101, as shown in Fig. 1, will now be discussed. In the event that end station A 111A decides to send a message packet to end station C 113, then the following events occur:

1. End station A 111A first attempts to learn the data link layer address of end station C 113. In the first step to learn the data link layer address of end station C 113, end station A 111A tests its network layer address against the network layer address of end station C 113 in order to determine if the two end stations are on the same link. And if the two end stations are on the same link, then end station A 111A has permission to transmit an Address Request Protocol message (a local ARP request message). An local ARP request message transmitted by an end station will be hereinafter referred to as a local ARP request.

End station A 111A uses a short mask, as shown in Fig. 3C, in making the determination as to whether or not end station C 113 is on the same link as end station A 111A, and because of the careful choice of end station network layer addresses, concludes that end station C 113 is on the same link as end station A 111A. As shown in Fig. 3C, the short mask used by end station A 111A has byte 161 and byte 163 both contain eight ones, and byte 165 and byte 167 both contain eight zeros. Accordingly, end station A 111A "sees" only byte 150 and byte 152 of the address fields shown in Fig. 3A, as the subnet address of itself and of end station C 113. By careful choice of the end station addresses, both end station A 111A and end station C 113 have the same value of bytes 150 152 in their network layer addresses. Accordingly, end station A 111A concludes that it and end station C 113 are on the same link.

Physically, as is shown in Fig. 1, end station A 111A is on link 110 and end station C 113 is on link 112. However, because of the careful choice of end station network layer addresses, and by end station A 111A using a short mask, end station A 111A concludes that it and end station C 113 are on the same link.

2. In response to its conclusion that end station C 113 is on the same link, end station A 111A transmits a local ARP Request onto link 110.

3. The local ARP Request is detected by brox 100. Logic unit 115 detects that the packet is a local ARP request. In response to receiving the local ARP request, brox 100 creates a new local ARP request. Brox 100 sends the new local ARP request onto link 112 with the data link layer address of end station C 113 in the data link destination address field 126. By using the long mask of Fig. 3B, where byte 164 contains eight ones, brox 100 can determine the link to which end station C 113 is connected. And by using the portion of the mask containing "0s", brox 100 can determine the network layer address of intended destination end station C 113.

End station C 113 receives the second local ARP Request message from LAN 112, and responds by creating a local ARP response message, and then transmitting the local ARP Response message onto link 112. The local ARP Response message contains the data link layer address of end station C 113.

5. Brox 100 detects the local ARP response message and processes it by logic 117. Brox 100 then determines that this local ARP response is in completion of a pending ARP request. Brox 100 then creates a second local ARP response message, and transmits the second local ARP response message onto link 110.

6. End station A 111A receives the second local ARP response message on LAN 110, and extracts from it the data link layer address of end station C 113.

7. End station A 111A transmits a message pack-

et to end station C 113 by placing the data link layer address of end station C 113 into the Data Link Destination Address field 126 of the message packet.

8. Brox 100 detects the message packet addressed to end station C 113, concludes that the data link destination address field 126 of the packet does not contain a data link destination address of brox 100, and therefor forwards the message packet as a bridge, and the message packet is thereby forwarded onto link 112. This bridge type forwarding operation is fast, as brox 100 made the decision by parsing only the Data Link Header field 122 of the message packet.

9. End station C 113 then detects the message packet by recognizing its data link destination address in the data link destination address field 126 of the message packet, and receives the message packet.

10. End station A 111A then may transmit a sequence of message packets to end station C 113 by: inserting the data link address of end station C 113 into the data link destination address field 126 of the message packet; and then brox 100 forwards the message packets as a bridge.

Advantages of the invention may be seen as follows. Brox 100 blocked the local ARP request message transmitted by end station 111A from being forwarded onto link 112, and in that respect functioned as a router. Brox 100 forwards the message packets transmitted by end station A 111A onto link 110, where the message packet contains the data link layer address of end station C 113 in the data link destination address field 126. Brox 100 thus rapidly forwards message packets, and also isolates local ARP traffic onto a single link.

A benefit of the invention is that local ARP request and local ARP response messages on one link, for example link 110, are blocked from the other link, in this example link 112, by brox 100. A further benefit of the invention is that in the event that an end station on one link, say link 112, sends a sequence of data packets to an end station on the other link, then all packets after the first packet are forwarded at bridge speed.

THIRD EXEMPLARY EMBODIMENT

A multiple hop embodiment of the invention will now be discussed. Referring now to Fig. 4, there is shown a more complex communications system 170. Communications system 170 has link 172, link 174, link 176, and link 178. Link 172 has end stations 172A, 172B, and 172C. Additionally, link 172 may have a further plurality of end stations, and may, for example, support up to several hundred end stations.

Link 174 has end stations 174A, 174B, 174C, and 174D. Also, Link 174 may support a further plurality of end stations.

Link 176 has end station 176A, end station 176B, and end stations 176C. Further, Link 178 has end station 178A, end station 178B, and end station 178C. Also, Link 176 and 178 may each support a further plurality of end stations.

Although links 172 174 176 178 are shown in Fig. 4 as straight lines, each link may, for example, be a token ring communications system such as an IEEE 802.5 token ring or an ANSI/IEEE FDDI token ring.

Brox 180 has a connection 182 to link 172, and also has a connection 184 to link 174. Brox 190 has connection 192 to link 174, and also has a connection 194 to link 176. Additionally, brox 190 has a connection 196 to link 178. Brox 200 has a connection 202 to link 174, and a connection 204 to a wide area network through a communications link, as indicated by jagged arrows 206.

The addresses of stations on links 172 174 176 178 are given by the address field of Fig. 3A. Bytes 150 and byte 152 of the addresses are the same for all stations on all links 172 174 176 178. Bytes 154 and byte 156 are different for the various links and stations as described hereinbelow.

Referring to Fig. 4, address 210 shows fields 154 and 156 of the address fields of Fig. 3A as assigned to stations on link 172. Field 154 contains; 11111000. Byte 156 may contain any combination of ones and zeros. The value of byte 156 is assigned to give a unique value for each end station 172A, 172B, 172C, and so forth. Only one byte 156 is reserved for the host address for end stations connected to link 172, and so the number of stations that may be individually addressed are 2**8, or 256 stations.

Address 212 show bytes 154 and 156 of addresses of Fig. 3A as assigned to stations on link 174. Byte 154 contains: 11110000. Byte 156 contains any value, and a unique value of byte 156 is assigned to each station connected to link 174. For example, end stations 174A, 174B, 174C, 174D, and connection 184 to brox 180, and connection 192 to brox 190 are each assigned a unique value of byte 156. Again, byte 156 may refer to as many as 256 stations.

Address 214 shows bytes 154 and 156 of addresses of Fig. 3A as assigned to stations on link 176. Byte 154 of address 214 contains: 11100000. Byte 156 contains any value, and a unique value of byte 156 is assigned to each station on link 176. For example, end station 176A, 176B, 176C, and connection 194 to brox 190, each have assigned a unique value of byte 156.

Address 216 shows bytes 154 and 156 of addresses of Fig. 3A as assigned to stations on link 178. Byte 154 of address of 216 contains; 11000000. Byte 156 has assigned any value, and has a unique value assigned for each station connected to link 178, for example, end stations 178A, 178B, 178C, and connection 196 to brox 190.

Each end station in communications system 170 uses mask 221 as shown in Fig. 7B. Mask 221 has all ones in byte B1 160, all ones in byte in B2 162, all zeros in byte B3 164, and all zeros in byte B4 166. Accordingly, whenever any end station in system 170 applies mask 221 to an address of an intended receiving station lying within system 170, the intended receiving station will appear to the transmitting end station to lie on the same link as the transmitting end station. This result arises from the following application of mask 221 to the address of any end station on system 170:

AND (A_my, mask)

AND (A_receive, mask),

where A_my is the network layer address of the end station,

and

where A_receive is the network layer address of the intended receiving end station.

The AND operations are followed by a test of the equality of the two AND operations.

In the event that the two AND operations are equal, then the transmitting end station concludes that it is on the same link as the intended receiving end station. Accordingly, in the event that the two AND operations are equal, the transmitting end station has permission to transmit a local ARP request message in order to inquire: "What is the data link address of the intended receiving end station?".

Referring now to Fig. 6A and Fig. 6B, there is shown an address in Fig. 6A. Part of the address of Fig. 6A is a subnet address, and part of the address is a host address, as determined by the appropriate mask.

There is shown a mask for a brox in Fig. 6B. The mask of Fig. 6B is assigned to broxs 180 190 200. Byte B1 160, and byte B2 162, both contain 8 ones, as shown in Fig. 6B. Byte B3 164 contains: 11111000. Byte B4 166 contains all zeros. When mask 220, Fig. 6B, is applied to the address 210 of link 172, or to address 212 of link 174, or to address 214 of link 176, or to address 216 of link 178, then the arrangement of ones in byte 164 of mask 220 allows the various links to be distinguished. That is, the longest address in byte 154 is in address 210 of link 172, and consists of five (5) ones. These five ones are masked for incorporation in the subnet address by the arrangement of five ones in byte 164 of mask 220, as shown in Fig. 6B. Accordingly, by making use of mask 220 assigned to brox 180, 190, or 200, a determination may be made as to which link a particular station address is located.

By making use of the 0s in mask 220 of Fig. 6B, an identification may be made of the host address 224 as shown in Fig. 6B. The host address, as shown in Fig. 6B, will comprise the last three bits of byte 154 and all eight bits of byte 156. Accordingly, byte 156 of address 210 will uniquely identify any of the stations connected to link 172. Alternatively, address 212, at byte 156, will identify any station connected to link 174, through a masking of the 0s of the mask 220 of Fig. 6b. Additionally, the end stations of link 176 and link 178 may likewise be identified by the 0s of mask 220 taken with the address 214 for link 176, and address 216 taken for link 178.

Since the host address of brox mask 220 of Fig. 6B uses eleven (11) bits, each link may have as many as 2**11 or two thousand forty eight (2,048) unique station addresses.

Referring now to Fig. 7A and Fig. 7B there is shown in Fig. 7B an end station mask 221. End station mask 221 has byte B1 160 contain: eight ones, byte B2 162 also contains eight ones. However, bytes B3 164 and byte B4 166 contain all zeros. Accordingly, an end station using mask 221, when applied to any address shown as address 210, 212, 214, 216 will determine that the addresses is given by bytes 150 and byte 152. Bytes 150 and 152 were chosen to be the same for all stations in communications system 170. Accordingly, by use of mask 221, all end stations of communications systems 170 will determine the same subnet address 222. A consequence of all end stations on communications system 170 determining the same subnet address through use of end station mask 221 is that, in accordance with the ordinary rules of operation for a TCP-IP communications type system, the end stations will have permission to transmit a local ARP request message.

Turning now to Fig. 5, there is shown the full address for addresses 210, 212, 214 and 216. As shown in Fig. 5, bytes 150 and 152 contain the same value for each of addresses 210, 212, 214 and 216. The addresses 210, 212, 214, and 216 differ only in the different values contained in byte 154 and byte 156.

Operation of the Multiple Hop System

In the event that end station 172A desires to transmit a data message to end station 178C, the following events occur:

1. End station 172A does an AND operation between its address and the address of end station 178C. End station 172A uses the end station mask 221 of Fig. 7B. End station 172A then concludes that the subnet address 222 of itself and the subnet address 222 of end station 172A are equal, and so concludes that it may transmit a local ARP request message to end station 178C in order to learn the data link address of end station 178C.

2. Brox 180 parses the data lint address of the local ARP request message, concludes that the local ARP request message is not addressed to brox 180, and accordingly receives the local ARP request message. Brox 180 recognizes the data packet as a local ARP request message, recognizes that the network layer address contained in

the local ARP request message is the network layer address of end station 178C, and then creates a remote ARP request message and sends the remote ARP request message to brox 190. Brox 180 knows to send the remote ARP request message to brox 190 because brox 180 maintains a forwarding table showing the end stations attached to various broxs. This forwarding table is built up by inter-brox traffic, analogous to the inter-router traffic used by routers to build up forwarding tables.

3. Brox 190 receives the remote ARP request message, and transmits on connection 196 to link 178 a standard local ARP request message containing the data link address of end station 178C.

4. End station 178C responds to the local ARP request message by transmitting a local ARP response message containing the data link layer address of end station 178C onto link 178.

5. Brox 190 receives the local ARP response message from end station 178C, and brox 190 creates a remote ARP response message, and sends it to brox 180.

6. Brox 180 then creates a standard local ARP response message and transmits it through connection 182 to link 172, with the data link layer address of end station 172A in the Data Link Destination Address field 126 of the local ARP response message.

7. End station 172A then interprets the local ARP response message and extracts from it the data link layer address of end station 178C.

8. End station 172A then transmits a message packet directed to end station 178C, where the message packet contains the data link layer address of end station 178C in the Data Link Destination Address field 126.

9. Brox 180 parses the data link header 122 of the message packet, discovers the Data Link Destination address field 126 does not contain the address of connection 182, and therefor forwards the message packet as a bridge.

10. The message packet is forwarded through connection 184 to link 174 where brox 190 detects the message packet at connection 192.

11. Brox 190 parses the data link header 122 of the message packet, concludes that the data link destination address field 126 does not contain the data link address of connection 192, and accordingly forwards the message packet as a bridge. Brox 190 bridges the message packet through connection 196 onto link 178 on the basis of a forwarding table maintained within brox 190.

12. The message packet is detected by end station 178C detecting its own data link address address in the Data Link Destination Address field 126 of the message packet. Accordingly, the intended receiving station 178C receives the message packet.

Each station in communication system 170 as shown in Fig. 4 uses end station mask 221 as shown in Fig. 7B, that is: on link 172 end stations 172A, 172B, 172C; on link 174 end stations 174A, 174B, 174C 174D; on link 176 end stations 176A, 176B, 176C; and on link 178 end stations 178A, 178B, 178C. The short mask 221 of Fig. 7B, when used by the end stations of communication system 170, causes all of the end stations of communication system 170 to conclude that they are on the same link. By coming to this conclusion, the end stations are permitted to transmit a local ARP request message for the data link address of an intended receiving station.

Each of the broxs, 180, 200, 190 use the longer brox mask 220, as shown in Fig. 6b. This longer brox mask 220 has ones in the first five (5) positions of byte 164, and these ones permit the broxs to distinguish the subnet addresses of links 172, 174, 176, 178.

In the event that an end station on communication system 170 prepares to transmit a message packet to an end station connected to communication system 170 through brox 200 by wide area network communication link 204, then the address in bytes 150 and 152 of the proposed Network Layer Destination Address field 140 of the message packet will differ from bytes 150, 152 of an address of links 172, 174, 176, 178. Accordingly, a brox receiving a local ARP request message for such a distant end station will forward, in performing as a router, and forward the message packet to brox 200 which then sends it on the wide area network link 204 to an appropriate brox having a link to the intended receiving end station.

As can be seen from Fig. 6B and Fig. 7B, a brox sees a longer subnet address then does an end station. As shown at byte 164 in Fig. 6B, the subnet address seen by a brox is 5 bits longer than a subnet address seen by an end station using mask 221 from Fig. 7B. The shorter end station mask causes end station to see all of the end stations on communications network 170 as being on the same link. The longer brox mask 220, as shown in Fig. 6B, permits a brox to distinguish the intended receiving end station and also the link to which the intended receiving end station is attached to.

Further, the assignment of the addresses of the end station is carefully done in order for the end stations to be distinguishable in accordance with the above discussion of the brox mask and the end station mask as shown in Fig. 6B and Fig. 7B. That is, each end station on communications system 170 was assigned a value of byte 154 as follows: link 172 was assigned a value of byte 154 of 11111000; link 174 was assigned a value of byte 154 of 11110000; link 176 was assigned a value of byte 154 of 11100000; link 178 was assigned a value of byte 154 of 11000000. This careful selection of addresses is the key to distinguishing end stations on different links by use of a short end station mask 221 and long brox mask 220.

FOURTH EXEMPLARY EMBODIMENT

Referring now to Fig. 8, communications system 300 is shown. Network layer addresses for the end stations of communications system 300 will now be discussed. The network layer address are used in the Network Layer Destination Address field 140 of Fig. 2. The network layer address may, for example, have 4 bytes, 150, 152, 154, 156 as set out in address field 140 in Fig. 3A. The value of byte 150 may be represented by the numeral N1. The value of byte 152 may be represented as numeral N2. The value of byte 154 may be represented as numeral N3. The value of byte of 156 may be represented as numeral N4. The four byte address may then he represented as follows:

N1.N2.N3.N4

In the above symbolic representation of the network layer address, the numbers N1, N2, N3, N4, refer to the bytes of address 140, that is bytes 150, 152, 154, 156, respectively.

Careful selection of the values of the bytes N1, N2, N3, N4 permit the invention to utilize a short end station mask for an end station to gain permission to send a local ARP request, and a long brox mask to enable a brox to distinguish end stations on different links.

As a further example of representation of a network layer address, a mask may be, for example 12 bits long, that is the mask boundaries may not coincide with octet boundaries. For example, the mask shown in Fig. 6B uses 5 bits from octet 164. Still, a symbolic representation of the network layer address may be given as:

A1.A2.A3

where A1, A2, and A3 are numbers used to refer to parts of the network layer address. For example, A1 may represent a group of related LANs, A2 may represent a particular LAN of the group, and A3 may represent the host address.

An exemplary assignment of end station address follows.

Referring now to communications system 300 as shown in Fig. 8, LAN 302 is connected by brox BR4 304 to LAN 306. LAN 308 is connected brox BR6 310 to LAN 306. LAN 306 is connected by brox BR3 312 to LAN 314. LAN 314 is connected by brox BR1 316 to LAN 320. LAN 320 is connected by brox 322 to LAN 324. LAN 320 is connected by brox BR5 326 to LAN 328.

Each LAN, 302, 306, 308, 314, 320, 324, 328 has connected in communications connection, a plurality of end stations. For simplicity, only particular end stations will be directly discussed. LAN 302 is shown with end station A 340. LAN 328 is shown with end station B 342.

Addresses are assigned to the end stations on each LAN in accordance with the present invention. For example, addresses may be assigned to the LANs as follows. Each LAN is assigned an address 140 comprising 4 bytes, bytes 150, 152, 154, 156. As a simplification, only three of the bytes will be discussed herein, bytes 152, 154, 156. The highest byte 150 may be assigned an arbitrary number, however each link has the same value of byte 150.

For example, in communication system 300 byte 152 is assigned the value "19". Accordingly, all stations on LAN 302 are assigned the address 19.3.•. The "•" means any unique number to distinguish the stations. All stations connected to LAN 306 are assigned the address 19.4.•. All stations connected to LAN 308 are assigned the address 19.8.•. All stations connected to LAN 314 are assigned the address 19.5.•. All stations attached to LAN 320 are assigned the address 19.5.•. All stations connected to LAN 324 are assigned the address 19.6.•. All stations attached to LAN 328 are assigned the address 19.7.•. The stations on each link include the plurality of end stations, as well as the connections to the respective broxs.

We now consider the event wherein end station A 340 on LAN 302, having a network layer address of 19.3.8 decides to send a message packet to end station B on LAN 328, having a network layer address of 19.7.5. The following events occur:

1. End station A 340 tests the network layer address of the intended receiving station B 342 in order to determine if end station 340 has permission to transmit a local ARP request message in order to learn the data link layer address of end station B 342. Addresses other than a TCP-IP 32 bit address may be used with the invention. For example, in the event thai a three octet address is used, the end station A 340 may use a mask having the values:

11111111.00000000.00000000

and each brox in communication system 300 utilizes a mask having the values:

11111111.11111111.11111111

in making forwarding decisions. Accordingly, end station A 340 utilizes its end station mask and concludes that the link where end station B 342 resides is "19.•.•", the same as the link where end station A 340 resides. Accordingly, end station A transmits a local ARP request message on LAN 302.

2. Brox BR4 304 detects the local ARP request message, interprets the local ARP request message, and creates a remote ARP request message directed to brox BR5 326. Brox BR4 304 knows to send the remote ARP request message to brox BR5 326 as a result of the broxs participating in a routing algorithms, as is well known in those skilled in the art of computer communications. Accordingly, brox BR4 304 transmits the remote ARP request message to brox BR5 326.

Brox BR5 326 receives the remote ARP request message, and in response thereto transmits a standard local ARP request message onto LAN 328.

3. End station B 342 receives the local ARP request transmitted onto LAN 328, and generates a local ARP response message containing the data link address of end station B 342, where the data link address corresponds to the station having network layer address 19.7.5.

4. Brox BR5 326 detects the local ARP response message transmitted by end station B 342, recognizes it as a completion of a pending remote ARP request message, and creates a remote ARP response message. Brox BR5 326 then sends the remote ARP response message to brox BR4 304. Brox BR4 304 receives the remote ARP response message, and in response thereto creates a local ARP response message. Brox BR4 304 then transmits the local ARP response message onto LAN 302.

5. End station A 340 receives the local ARP response message from LAN 302, and interprets the local ARP response message, and stores the data link layer address of end station B 342.

6. End station A 340 then constructs a data packet containing the data link layer address of end station B 342 in Data Link Destination Address field 126 of a message packet 120.

7. Brox BR4 304 detects the data packet, parses the Data Link Header 122 and finds that the contents of the Data Link Destination Address field 126 are not an address used by brox BR4 304, and so concludes to forward the message packet as a bridge.

The data packet transmitted by end station A 340 is forward by all of the intermediate broxs behaving as bridges, brox BR4 304, brox BR3 312, brox BR1 316, and brox BR5 326. The data packet is thus forwarded at each hop, in some designs, in less than 1/200th of the time that would be required if a brox operated as a router.

The local ARP Request transmitted by end station A 340 onto LAN 302 is blocked by brox BR4 304. That is, the local ARP Request is not forwarded to LAN 306 as it would be if brox BR4 304 acted as a standard bridge. That is, brox BR4 304 acts as a standard router and isolates LAN 302 from LAN 306, in that it does not forward ARP messages between the LANs. Likewise, all local ARP requests and local ARP responses are isolated to the LAN on which they were created by the broxs functioning as routers.

Accordingly, data traffic may be forwarded between any end station on any link in communications network 300 by each of the broxs forwarding rapidly as a bridge. Also, the local ARP traffic generated on each LAN is isolated from each of the other links by each brox functioning as a router.

The invention, in all embodiments, has the beneficial effect that it speeds forwarding of data packets and so improves throughput in the data communications system.

A further benefit of the invention is that, in the event that an intermediate forwarding station is an old style router that does not change into a bridge in accordance with the Rules of the invention hereinabove, the invention will work perfectly well with all of the intermediate broxs functioning in accordance with the invention. Any intermediate "old style" routers already installed in an old system will not interfere with the improvements gained from new broxs added to the system.

Accordingly, the invention greatly improves the speed at which a message is forwarded over multiple links of a complex communications system.

Claims

1. In a communications system having a first communications link and a second communications link, at least one end station capable of communicating on each said communications link, an apparatus for forwarding a packet from said first link to said second link, said apparatus being capable of detecting a network layer header on a data packet, said network layer header having a destination address, said apparatus comprising:

   means for assigning an apparatus mask having a forwarding mask length to said apparatus for distinguishing said destination address into a subnet address part and into a host address part;

   means for assigning an end station mask having an end station mask length to said at least one end station for distinguishing said destination address into a subnet address part and into a host address part;

   means for assigning a greater length to said forwarding mask length than to said end station mask length, to enable said end station in using said end station mask to identify all end stations on said first link and said second link as being on a single link, and to enable said apparatus in using said forwarding mask to distinguish on which of said first link or said second link an end station addressed by said network layer address is located.

2. The apparatus as in claim 1 further comprising:

   means for a selected end station to transmit a local Address Request Protocol (hereinafter ARP) request message onto said first communications link, said local ARP request message requesting a data link address of a receiving end station;

means, in response to said local ARP request message, for said apparatus to create a remote ARP request message and to send said remote ARP request message to a second forwarding apparatus connected to said second link having said receiving end station connected thereto;

means for said apparatus to receive a remote ARP response message containing said data link address of said receiving end station from said second forwarding apparatus;

means, responsive to said remote ARP request message, for said apparatus to create a local ARP response message and to send said local ARP response message to said selected end station.

3. The apparatus as in claim 1 further comprising:

means associated with said forwarding apparatus to forward a data packet as a bridge in the event that a data link address in a message packet is not a data link address of said apparatus.

4. The apparatus as in claim 1 further comprising:

means associated with said apparatus to forward a message packet as a router in the event that a data link address in said message packet is a data link address of said apparatus.

5. A communications system having a plurality of communications links, a forwarding apparatus for forwarding a message packet from a first link to a second link, said first link and said second link selected from said plurality of links, said apparatus being capable of detecting a network layer header on a data packet, said network layer header having a destination address, said system comprising:

means for assigning an apparatus mask having a forwarding mask length to said apparatus for distinguishing said destination address into a subnet address part and into a host address part;

means for assigning an end station mask having an end station mask length to said said at least one end station for distinguishing said destination address into a subnet address part and into a host address part;

means for assigning a greater length to said forwarding mask length than to said end station mask length, to enable said end station in using said end station mask to identify all end stations on said plurality of links as being on a single link, and to enable said apparatus in using said forwarding mask to distinguish which of said plurality of links an end station addressed by said network layer address is located;

means, responsive to said greater length of said forwarding mask than said end station mask, for a selected end station to transmit a local ARP request message onto an attached link, said local ARP request message requesting a data link address of a receiving end station, said receiving end station connected to a second link that is not said attached link;

means, responsive to said local ARP request message, for said apparatus to send a remote ARP request message to a second forwarding apparatus attached to said second link;

means, responsive to said remote ARP request message, for said second forwarding apparatus to send a second local ARP request message to said intended receiving end station;

means, responsive to said second local ARP request message, for said intended end station to send a local ARP response message to said second forwarding apparatus, said second local ARP response message containing said data link address of said intended end station;

means, responsive to said local ARP response message, for said second forwarding apparatus to send a remote ARP response message to said apparatus;

means, responsive to said remote ARP response message, for said apparatus to send a second local ARP response message to said selected end station;

means, responsive to said second local ARP response message, for said selected end station to receive a data link address of said intended receiving end station, and for said selected end station to transmit a message packet containing said data link address of said selected receiving end station, and for said apparatus to forward said message packet as a bridge in response to said data link address of said intended receiving end station being contained in said message packet.

6. A communications system having a plurality of communications links, a forwarding apparatus for forwarding a message packet from a first link to a second link, said first link and said second link selected from said plurality of links, said apparatus being capable of detecting a network layer header on a data packet, said network layer header having a destination address, comprising:

means for assigning an apparatus mask having a forwarding mask length to said apparatus for distinguishing said destination address into a subnet address part and into a host address part;

means for assigning an end station mask having an end station mask length to said said at least one end station for distinguishing said destination address into a subnet address part and into a host address part;

means for assigning a greater length to said forwarding mask length than to said end station mask length, to enable said end station in using said end station mask to identify all end stations on said plurality of links as being on a single link, and to enable said apparatus in using said forwarding mask to distinguish which of said plurality of links an end station addressed by said network layer address is located.

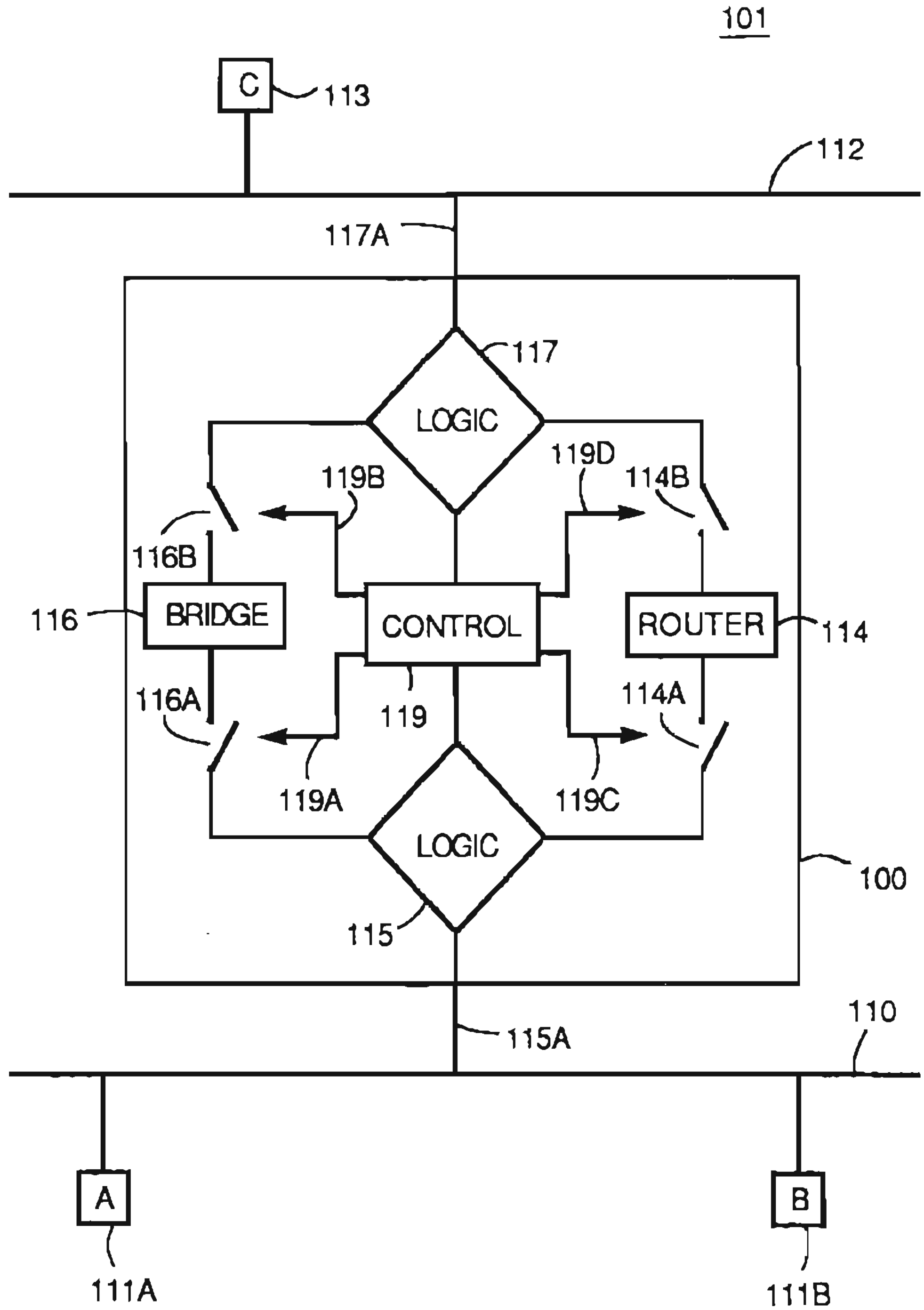

101
C
113
112
117A
117
LOGIC
119B
119D
114B
116B
116
BRIDGE
CONTROL
ROUTER
114
116A
114A
119
119A
119C
LOGIC
100
115
110
115A
A
B
111A
111B

FIGURE 1

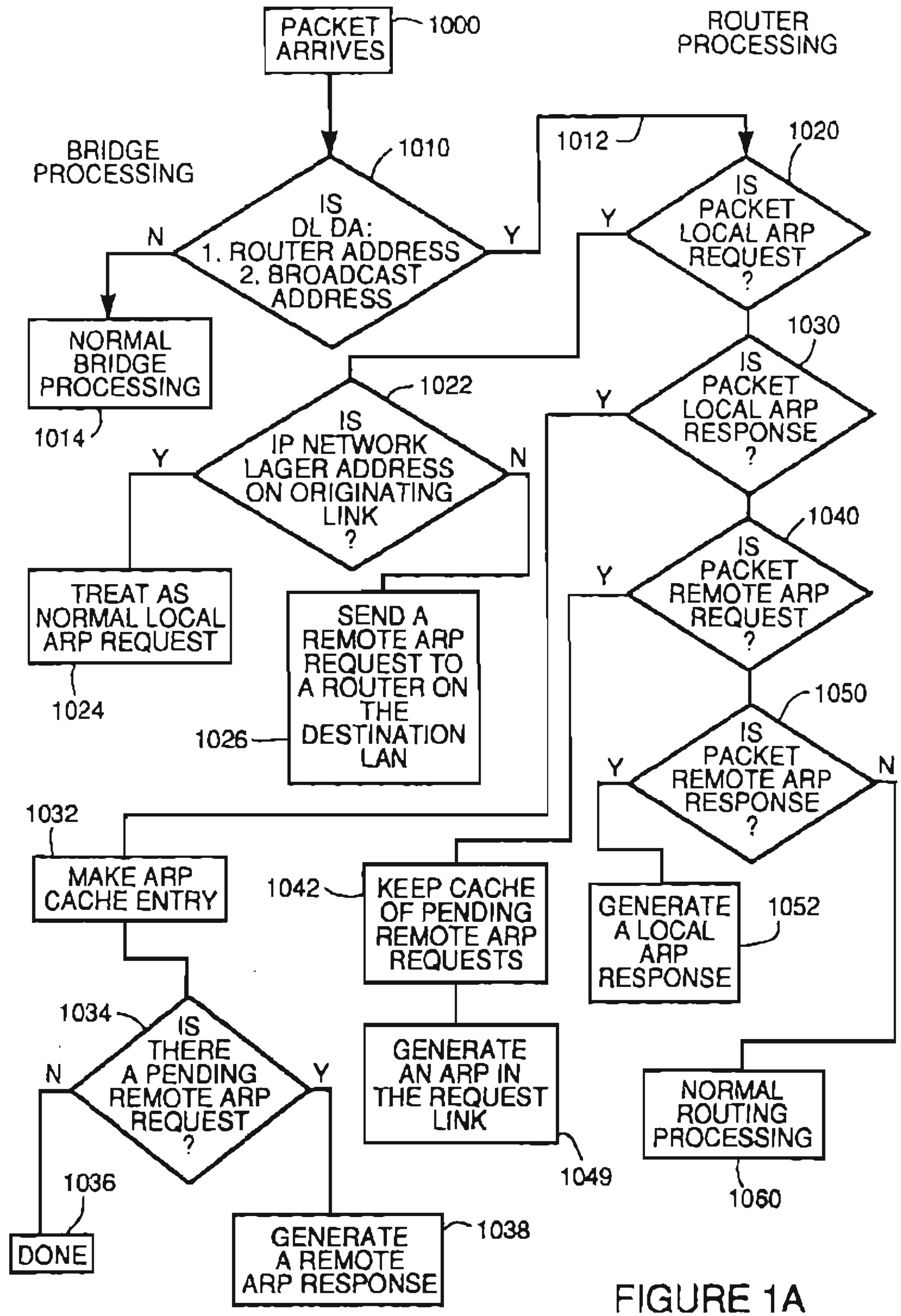

PACKET ARRIVES
1000
ROUTER PROCESSING
BRIDGE PROCESSING
1010
1012
1020
IS DL DA: 1. ROUTER ADDRESS 2. BROADCAST ADDRESS
N
Y
IS PACKET LOCAL ARP REQUEST ?
NORMAL BRIDGE PROCESSING
1014
1030
IS PACKET LOCAL ARP RESPONSE ?
1022
IS IP NETWORK LAGER ADDRESS ON ORIGINATING LINK ?
TREAT AS NORMAL LOCAL ARP REQUEST
1024
SEND A REMOTE ARP REQUEST TO A ROUTER ON THE DESTINATION LAN
1026
1040
IS PACKET REMOTE ARP REQUEST ?
1050
IS PACKET REMOTE ARP RESPONSE ?
1032
MAKE ARP CACHE ENTRY
1042
KEEP CACHE OF PENDING REMOTE ARP REQUESTS
GENERATE A LOCAL ARP RESPONSE
1052
1034
IS THERE A PENDING REMOTE ARP REQUEST ?
GENERATE AN ARP IN THE REQUEST LINK
1049
NORMAL ROUTING PROCESSING
1060
1036
DONE
GENERATE A REMOTE ARP RESPONSE
1038

FIGURE 1A

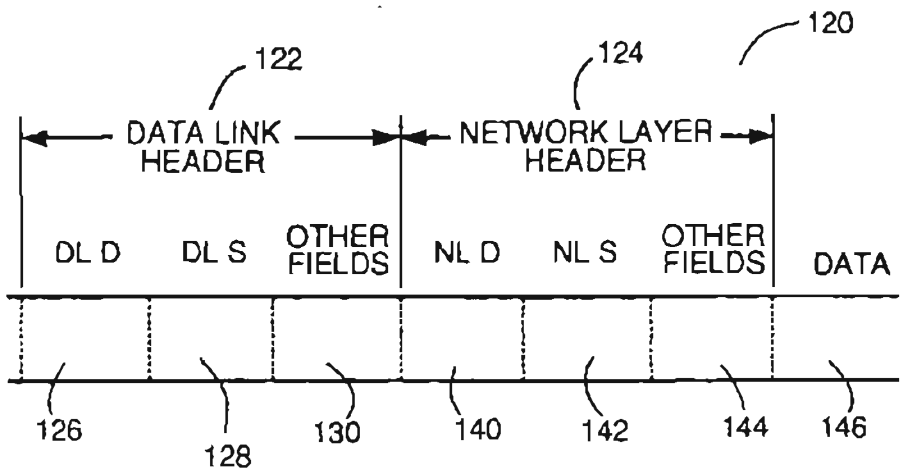


FIGURE 2

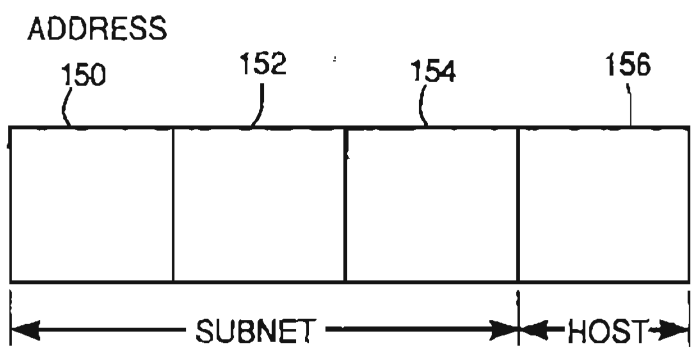


FIGURE 3A

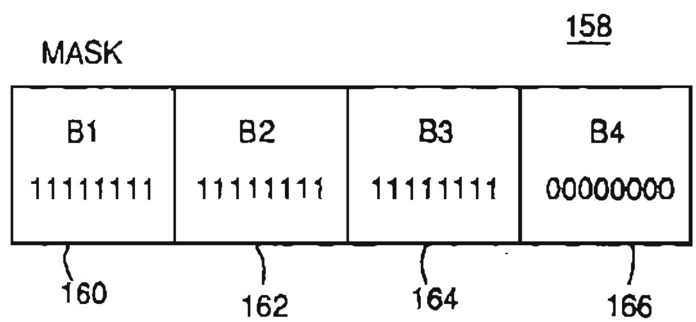


FIGURE 3B

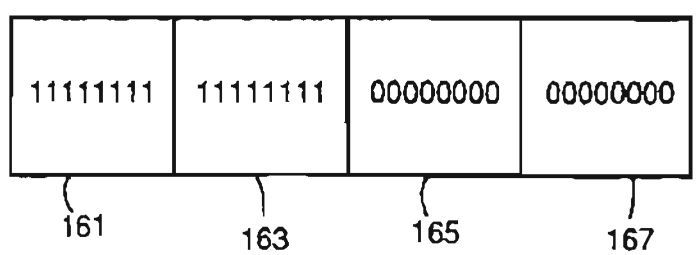


FIGURE 3C

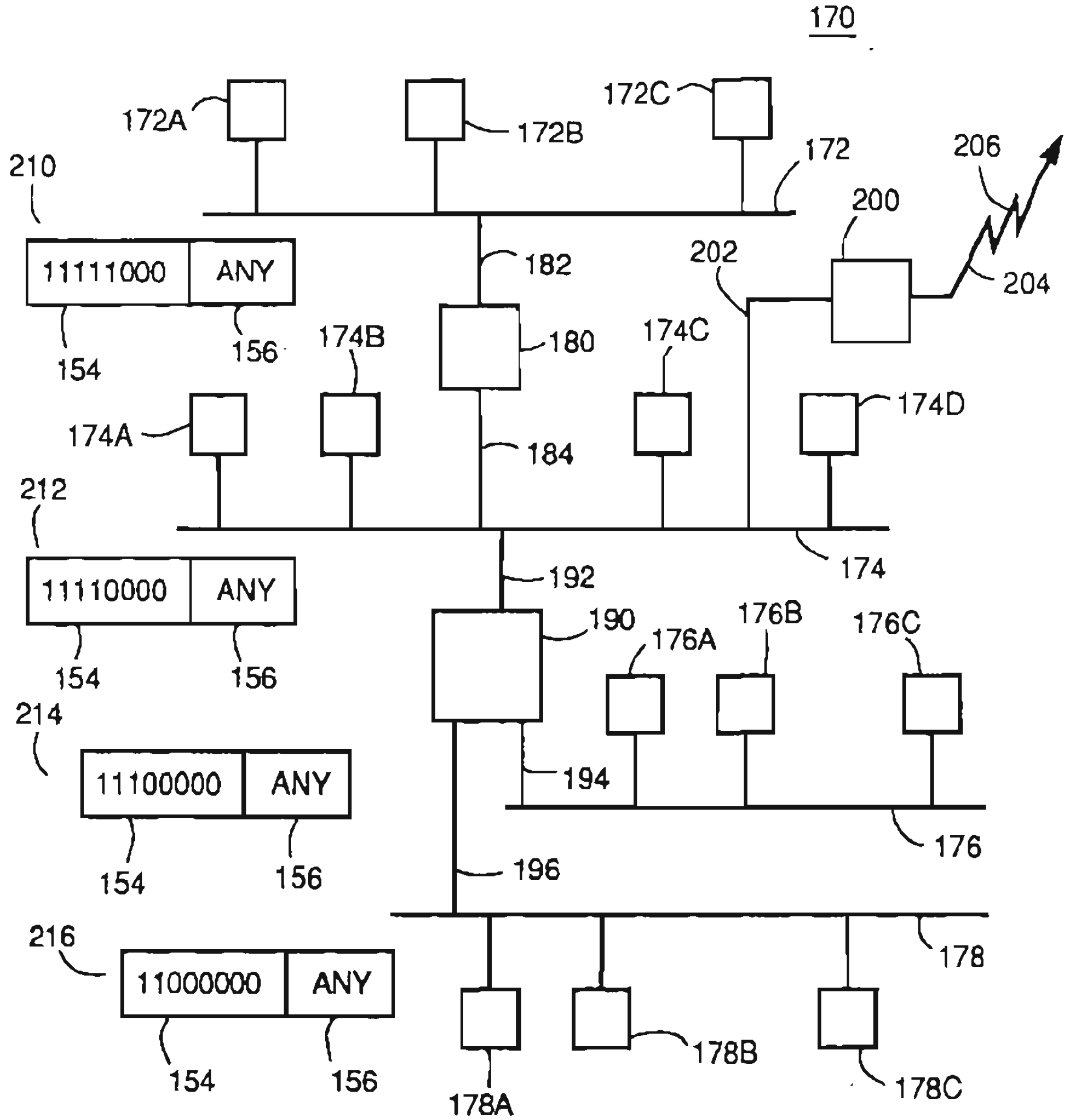


FIGURE 4

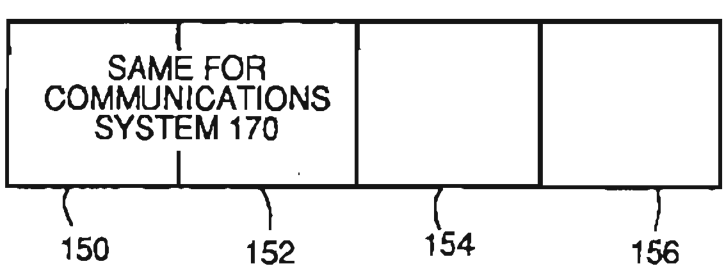


FIGURE 5

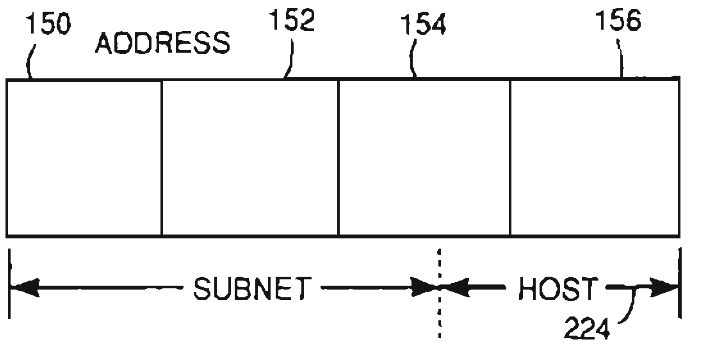


FIGURE 6A

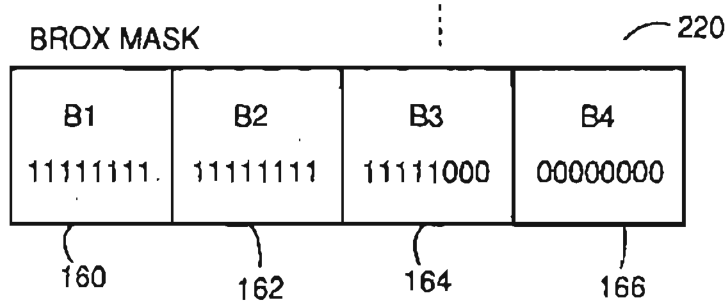


FIGURE 6B

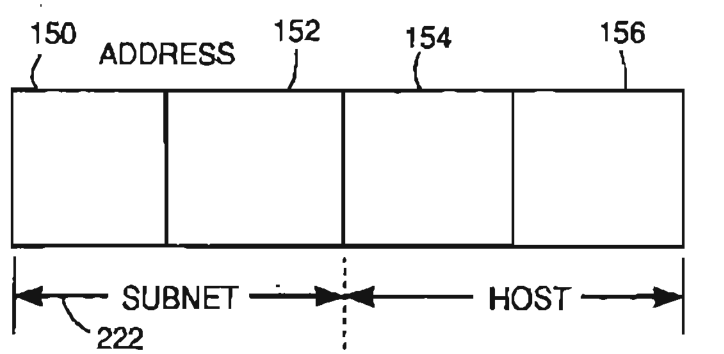


FIGURE 7A

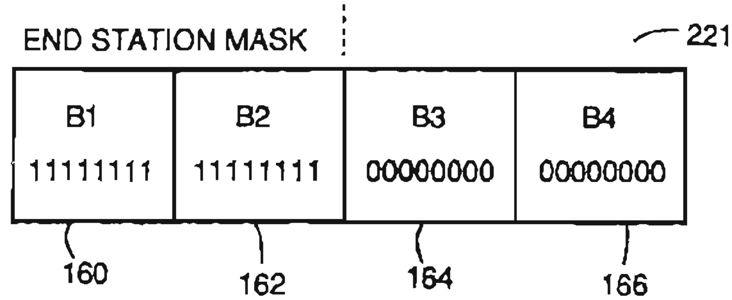


FIGURE 7B

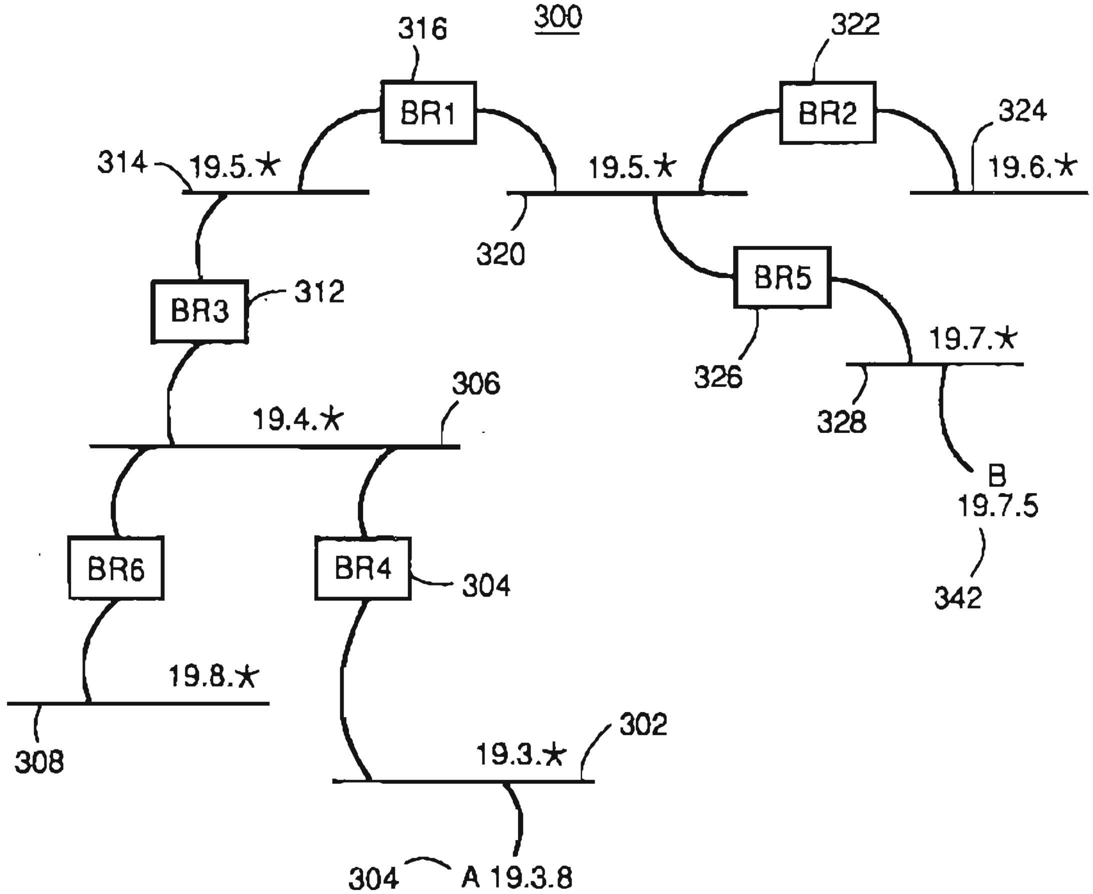

300
316
BR1
322
BR2
324
314
19.5.*
19.5.*
19.6.*
320
BR5
BR3
312
19.7.*
306
326
328
19.4.*
B
19.7.5
BR6
BR4
304
342
19.8.*
302
308
19.3.*
304
A 19.3.8

FIGURE 8

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 92 30 5234

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
A	EP-A-0 357 136 (PHILIPS) * column 2, line 15 - column 3, line 32 * ---	1,5,6	H04L12/46
A	EP-A-0 255 767 (AT&T) * column 2, line 34 - column 3, line 50 * ---	1,5,6	
A	IEEE NETWORK: THE MAGAZINE OF COMPUTER COMMUNICATIONS vol. 2, no. 1, January 1988, NEW YORK US pages 49 - 56 L. BOSACK ET AL. 'Bridges and Routers, Observations Comparisons and Choosing. Problems in Large LANs' * page 52, left column, line 10 - line 25 * * page 52, right column, line 7 - line 41 * ---	2-4	
P,A	EP-A-0 465 201 (DEC) * the whole document * -----	2	
			TECHNICAL FIELDS SEARCHED (Int. Cl.5)
			H04L

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	22 SEPTEMBER 1992	MESSELKEN M.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0401)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: 0 559 047 A1

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 93102782.5

(51) Int. Cl.5: H04M 1/66

(22) Date of filing: 23.02.93

(30) Priority: 02.03.92 US 844268

(43) Date of publication of application: 08.09.93 Bulletin 93/36

(84) Designated Contracting States: AT BE DE ES FR GB IT NL PT

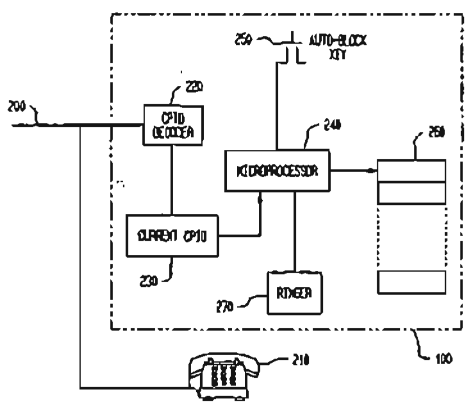
(71) Applicant: ROLM COMPANY
4900 Old Ironside Drive P.O. Box 58075
Santa Clara, CA 95052(US)

(72) Inventor: Iglehart, David
2006 Westridge Drive
Austin, TX 78704(US)
Inventor: Ford, Gordon
1004 Lime Rock Drive
Round Rock, TX 78681(US)

(74) Representative: Fuchs, Franz-Josef, Dr.-Ing.
Postfach 22 13 17
D-80503 München (DE)

(54) Future blocking of incoming telephone calls.

(57) Method and apparatus for screening telephone calls for use in conjunction with a telephone set as, opposed to a switch, which provides a capability of storing a telephone number for use in blocking calls from the telephone number, without the need for re-entering the telephone number.



EP 0 559 047 A1

Technical Field of the Invention

The present invention pertains to method and apparatus for screening telephone calls for use in conjunction with a telephone set, as opposed to a switch, which provides a capability of storing a telephone number for use in blocking calls from the telephone number, without the need for re-entering the telephone number.

Background of the Invention

Several methods exist in the prior art for providing screening of incoming telephone calls. In a first prior method, a called party utilizes a display telephone to visually identify a caller's telephone number and then chooses whether or not to answer. In a second prior art method, a secretary manually screens calls and chooses whether or not to transfer the caller to the called party. In a third prior art method, the called party blocks all incoming calls by placing his/her telephone off hook or by activating a Do Not Disturb feature. In a fourth prior art method, the called party can screen incoming calls utilizing an audio-monitoring capability which is associated with most answering machines. In a fifth prior art method, the called party can order a service from a telephone company that enables him/her to key in an access code after receiving an incoming telephone call to cause the telephone company to block future calls from the telephone number. In a sixth prior art method, the called party may enter a telephone number into a data base of telephone numbers to be blocked by centralized switching equipment.

In general, the above-described prior methods suffer from several disadvantages. Namely, they: (a) do not provide immediate control to the called party; (b) do not permit him/her to identify the telephone number of the incoming call; (c) do not automate the process of identifying and listing telephone phone numbers to be blocked; and (d) can only be used to block calls from telephone numbers that are known in advance.

In particular, the first prior method has the disadvantages that: (a) not all telephones have displays; (b) the called party may not know the telephone numbers of callers he/she wishes to block; and (c) the called party has to utilize a more complicated, manual method to identify and list the telephone number he/she wishes to block.

In particular, the second prior art method has the disadvantages that it is complicated and does not provide called party control. Specifically, the called party has to rely on a secretary who knows in advance which callers should be blocked. If a called party determines during a call that he/she does not wish to receive further calls from the caller, he/she would have to communicate with the secretary to block future calls from that caller.

In particular, the third prior art method has the disadvantages that it is impractical and unspecific. The called party can block all calls or no calls, with no ability to identify and list telephone numbers.

In particular, the fourth prior art method has the disadvantages that it requires the called party has to listen to part of an incoming call to determine if he/she wants to receive it. Moreover, it requires him/her to be rude to callers he/she does not want to receive, making them talk into his/her answering machine before he/she answers. Also, some callers that he/she does not want to speak to may simply hang up rather than leave a message.

In particular, the fifth prior art method has the disadvantages that it relies on a capability of a central switching unit and that the user would have to pay a monthly service fee for blocking.

In particular, the sixth prior art method has the disadvantages that the called party would have to know in advance the telephone numbers he/she wishes to block and the called party lacks an automatic way to identify and list, during a call, the caller's telephone number, so that it can be blocked in the future. Furthermore, such a method relies on a central switching data base to store the list of numbers to be blocked, to which not all users have access.

As a result, there is a need in the art for a method and apparatus for screening telephone calls for use in conjunction with a telephone set, as opposed to a switch, which provides a capability of storing a telephone number for use in blocking calls from the telephone number, without the need for re-entering the telephone number.

Summary of the Invention

Embodiments of the present invention advantageously satisfy the above-described need in the prior art by providing method and apparatus for screening telephone calls for use in conjunction with a telephone set, as opposed to a switch, which provides a capability of storing a telephone number for use in blocking calls from the telephone number, without the need for re-entering the telephone number.

In particular, embodiments of the present invention provide method and apparatus which enable a called party to specify, preferably with a single key stroke, during a call, that further calls from the caller's telephone number be blocked, i.e., will not ring the called party's telephone. It is important to note that method and apparatus is provided by the capabilities of the telephone and not by those of a central switching unit. Further, the inventive apparatus can either be contained in the telephone or be arranged separate from the telephone. As those of ordinary skill in the art will readily appreciate, in utilizing embodiments of the present invention, the called party does not have to rely on support by others, does not have to have a display, can have the caller's telephone number identified and listed immediately, can store a large number of extensions for blocking, and does not have to have central switching support or pay a monthly fee for a blocking service.

Specifically, an embodiment of the present invention comprises: (a) means for determining the telephone number of an incoming call; (b) indication means for receiving an indication that future calls from the telephone number of the incoming call are to be blocked; (c) means, responsive to the indication means, for storing the telephone number in a data base; and (d) means, responsive to the incoming call, for determining whether the telephone number is stored in the data base and, if so, blocking the incoming call.

As an example of the advantageous use of the present invention, consider the following scenario. John, a hard-working businessman, sits down with his wife and family for dinner. As on most nights, the telephone starts ringing promptly at six from sales calls, and John finds himself listening to a computer trying to sell him smoked cheese. A smile crosses John's face as he presses an "Auto-block" key on his telephone and hangs up, secure in the knowledge that he will never be bothered by calls from that telephone number again.

The invention will be better understood from the following more detailed description taken with the accompanying drawings and claims.

Brief Description of the Drawings

FIG. 1 shows a block diagram of an embodiment of the present invention for use in blocking incoming telephone calls.

Corresponding elements in each of the drawings have the same reference numbers.

Detailed Description

FIG. 1 shows a block diagram of apparatus 100 which is an embodiment the present invention for use in blocking incoming telephone calls. As shown in FIG. 1, telephone line 200 supplies input to apparatus 100 and to telephone 210. In the embodiment shown in FIG. 1, the ringer in telephone 210 is turned off.

Telephone line 200 provides calling party identification (CPID), commonly referred to as Automatic Number Identification (ANI), which is available with telephone services such as Integrated Services Digital Network (ISDN) or Customized Local Area Signalling Services (CLASS), all of which is well known to those of ordinary skill in the art.

As shown in FIG. 1, output from telephone line 200 is applied as input to CPID decoder 220. CPID decoder 220 is apparatus which is well known to those of ordinary skill in the art for decoding the telephone number of incoming a call on telephone line 200 into a static digital format which can be utilized by other elements in apparatus 100. Output from CPID decoder 220 is applied as input to current CPID 230. Current CPID 230 is memory storage means which is well known to those of ordinary skill in the art such as solid state random access memory which stores information produced by CPID decoder.

Output from current CPID 230 is applied as input to microprocessor 240. Microprocessor 240 is any appropriate general purpose microprocessor of a type which is well known to those of ordinary skill in the art which is capable of monitoring a number of stimuli and making decisions on the basis of those stimuli to provide predetermined responses.

Auto-block key 250 is means of providing stimuli which is well known to those of ordinary skill in the art to microprocessor 240 for use in indicating that a called party has decided to block future calls from the telephone number contained in current CPID 230. Thus, whenever the called party depresses auto-block key 250, microprocessor 240 retrieves the telephone number stored in current CPID 230 and transfers it to data base 260.

Data base 260 is means of memory storage which is well known to those of ordinary skill in the art such as, for example, random access memory, for retaining a list of telephone numbers to be blocked. Finally, ringer 270 is a device which is well known to those of ordinary skill in the art for alerting a called party that an incoming call is being received. As will be explained below, ringer 270 will not be activated for blocked calls.

The following describes the operation of apparatus 100 shown in FIG. 1. When an incoming call is being received by apparatus 100, CPID decoder decodes the CPID and stores it in current CPID 230. Microprocessor 240 retrieves the CPID stored in current CPID 230 and compares it with the telephone numbers stored in data base 260. The comparison of the CPID of the incoming calls with the telephone numbers in the data base may be performed in any one a multiplicity of methods which are well known to those of ordinary skill in the art. If there is a match, i.e., the incoming call is to be blocked, ringer 270 is not activated and the called party does not answer the call and, advantageously, is not even made aware of its existence. However, if there is a match, microprocessor 240 causes ringer 270 to be activated to alert the called party of the incoming call.

Later, at any time during the call, or at any time while the telephone number is stored in current CPID 230, if the called party desires to block future calls from the telephone number he/she depresses auto-block key 250. In response, by generation of a signal or by polling, microprocessor 240 retrieves the telephone number from current CPID 230 and stores it in data base 260.

As those of ordinary skill in the art readily appreciate, embodiments of the present invention may either be contained in telephone 210 or may be separate from telephone 210. Further, if apparatus 100 or telephone set 210 has a display, the number to be blocked may be displayed thereby. Still further, apparatus 100 may further comprise a keypad for use in interacting with microprocessor 240 or the keypad of telephone 210 may be used to interact with microprocessor 240 for use in deleting telephone numbers from the data base. Lastly, although embodiments of the present invention have been described which utilize an auto-block key, it should be understood that this is not a limitation on the scope of the present invention. Specifically, in an embodiment which utilizes a keypad or the keypad of telephone 210, the notification of the desire to block future calls from a telephone number may be indicated by a predetermined one or more keypad presses.

It is to be appreciated and understood that the specific embodiments of the invention described hereinbefore are merely illustrative of the general principles of the invention. Various modifications may be made by those skilled in the art consistent with the principles set forth hereinbefore.

Claims

1. Apparatus for use in conjunction with a telephone set for blocking future incoming telephone calls which comprises:

   means for determining the telephone number of an incoming call;

   indication means for receiving an indication that future calls from the telephone number of the incoming call are to be blocked;

   means, responsive to the indication means, for storing the telephone number in a data base; and

   means, responsive to the incoming call, for determining whether the telephone number is stored in the data base and, if so, blocking the incoming call.

2. The apparatus of claim 1 further comprising means for generating the indication.

3. The apparatus of claim 2 wherein said means for generating the indication comprises a key.

4. A telephone set for blocking future incoming telephone calls which comprises:

   means for determining the telephone number of an incoming call;

   indication means for receiving an indication that future calls from the telephone number of the incoming call are to be blocked;

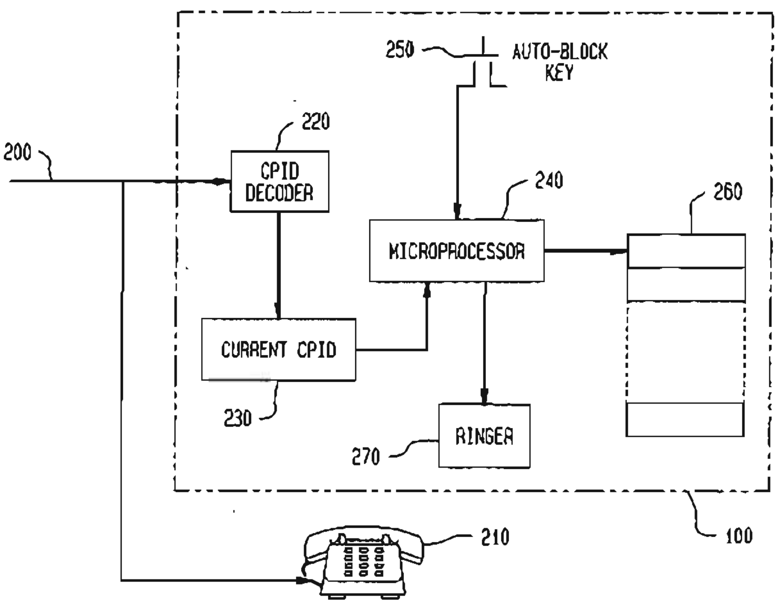
   means, responsive to the indication means, for storing the telephone number in a data base; and

   means, responsive to the incoming call, for determining whether the telephone number is stored in the data base and, if so, blocking the incoming call.

5. The telephone set of claim 4 further comprising means for generating the indication.

6. The apparatus of claim 5 wherein said means for generating the indication comprises a key on the telephone set.

7. The telephone set of claim 4 wherein the indication is provided by a predetermined sequence of one or more keys on the telephone set.


250
AUTO-BLOCK KEY
220
200
CPID DECODER
240
260
MICROPROCESSOR
CURRENT CPID
230
270
RINGER
210
100

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 93 10 2782

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
X	DE-A-3 234 093 (SIEMENS A.G.) * page 6, line 32 - page 12, line 23; figures 1,2 * ---	1,2	H04M1/66
X	PATENT ABSTRACTS OF JAPAN vol. 14, no. 371 (E-963)10 August 1990 & JP-A-21 34 950 (NEC) * abstract * ---	1-7	
X	PATENT ABSTRACTS OF JAPAN vol. 14, no. 187 (E-917)16 April 1990 & JP-A-20 36 657 (CANON) * abstract * ---	1-7	
X	PATENT ABSTRACTS OF JAPAN vol. 14, no. 328 (E-952)13 July 1990 & JP-A-21 11 146 (OMRON TATEISI ELECTRON) * abstract * ---	1-7	
X,P	PATENT ABSTRACTS OF JAPAN vol. 16, no. 150 (E-1189)14 April 1992 & JP-A-40 04 647 (MITSUBISHI ELECTRIC CORP.) * abstract * -----	1-7	TECHNICAL FIELDS SEARCHED (Int. Cl.5) H04M

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	02 APRIL 1993	DELANGUE P.C.J.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0401)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 597 691 A1**

(12) EUROPEAN PATENT APPLICATION

(21) Application number : 93308975.7

(22) Date of filing : 10.11.93

(51) Int. Cl.^{5} : **G06F 13/12**

(30) Priority : 12.11.92 US 975037

(43) Date of publication of application : 18.05.94 Bulletin 94/20

(84) Designated Contracting States : DE FR GB

(71) Applicant : International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor : Chamberlain, Paul Clement
10402-A Golden Meadow
Austin, Texas 78758 (US)
Inventor : Smith, Brian Rutledge
2303 Falcon Drive
Round Rock, Texas 78681 (US)

(74) Representative : Bailey, Geoffrey Alan
IBM United Kingdom Limited Intellectual Property Department Hursley Park
Winchester Hampshire SO21 2JN (GB)

(54) Queuing system and method of operation.

(57) A queuing system and method of operation are provided that reduces latency and increases efficiency in a general purpose queuing system. The technique of the present invention is applied in an intermediate node that receives an entity, such as information, from a first node, and transfers that entity to a second node. The technique comprises the steps of (a) receiving at the intermediate node (B) a first block of the entity sent by the first node (A) ; (b) upon receipt of the block, initiating the sending of a subsequent block of the entity to the intermediate node ; (c) concurrently with step (b), transferring the first block of the entity to the second node (C) ; (d) upon receipt of an acknowledgement from the second node (C), causing the intermediate node to transfer a portion of the entity to the second node (C), the portion transferred being all of the entity that has at the time of the transferral been received by the intermediate node (B) from the first node (A) since the previous transfer was made ; and (e) repeating steps (b) and (d) until all of the entity has been transferred.

The above technique is adaptive to many environments and will optimize throughput for systems that need to transfer entities such as information. This system and method can handle mismatched flow problems from diverse environments and provides optimal flow for solutions that require guaranteed transfers. This algorithm can change and adapt to varying circumstances. It can be altered in real-time for communication systems. If the block size changes the modification does not alter the smooth flow of the algorithm.

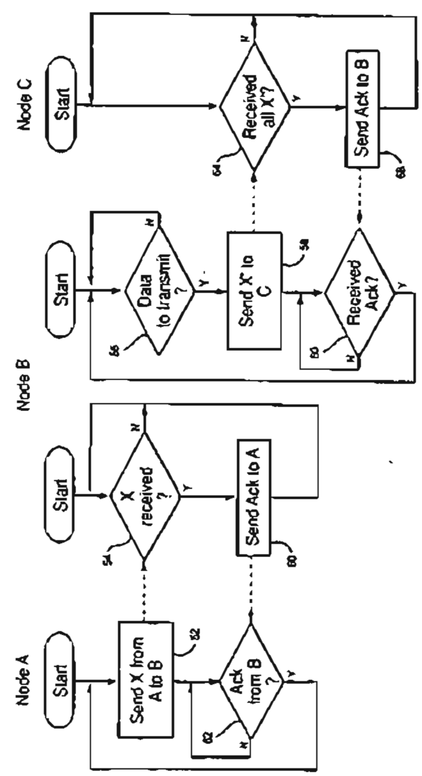


FIG. 4

EP 0 597 691 A1

Jouve, 18, rue Saint-Denis, 75001 PARIS

This invention relates to queuing systems and more particularly to transporting of entities, such as items or information, from one location to another using an intermediate queue.

As computer manufacturers develop faster and more efficient computer communication networks, increasingly there are cases of mismatched bus and communication media speeds. One example is the IBM Microchannel and an IBM Fiber Channel (FCS) Microchannel adapter. The IBM Microchannel is capable of sustaining approximately 50 megabytes/sec, whereas the IBM FCS adapter can support either 25 megabytes/sec or 100 megabytes/sec. This mismatch also occurs with the Microchannel and an IBM Token Ring adapter. Because of these mismatches, data transfers can often be very inefficient with respect to the given communication media speed. They can also be very efficient, but have a long delay in starting transmission. This delay is often referred to as latency. These two problems, latency and efficiency, are classic in the field of communications.

There are numerous applications which require optimization of either latency or throughput. There are also those that require optimization of both. Customers are increasingly interested in low latency and very efficient use of the communication media. The present state of the art fails to provide an adaptive yet simple throughput mechanism between systems when trying to minimize latency and maximize efficiency.

It is therefore an object of the present invention to provide an improved queuing technique for an entity being transferred from a first node to a second node via an intermediate node.

Accordingly the present invention provides a method of operating an intermediate node to receive an entity from a first node and to transfer the entity to a second node, the first and second nodes being connected to the intermediate node by transmission links, the method comprising the steps of:

(a) receiving at the intermediate node a first block of the entity sent by the first node;

(b) upon receipt of the block, initiating the sending of a subsequent block of the entity to the intermediate node;

(c) concurrently with step (b), transferring the first block of the entity to the second node;

(d) upon receipt of an acknowledgement from the second node, causing the intermediate node to transfer a portion of the entity to the second node, the portion transferred being all of the entity that has at the time of the transferral been received by the intermediate node from the first node since the previous transfer was made; and

(e) repeating steps (b) and (d) until all of the entity has been transferred.

Viewed from a second aspect the present invention provides a queuing system in an intermediate node for receiving an entity from a first node and transferring the entity to a second node, the first and second nodes being connected to the intermediate node by transmission links, the system comprising: reception means in the intermediate node for receiving a first block of the entity sent by the first node; initiation means, responsive to the reception means indicating receipt of the block for initiating the sending of a subsequent block of the entity to the intermediate node; a transfer means, operating concurrently with the initiation means, to transfer the first block of the entity to the second node; the transfer means further, upon receipt by the intermediate node of an acknowledgement from the second node, transferring a portion of the entity to the second node, the portion transferred being all of the entity that has at the time of the transferral been received by the intermediate node from the first node since the previous transfer was made; the initiation means and transfer means repeating their functions until all of the entity has been transferred.

The present invention reduces latency and increases efficiency in a general purpose queuing system. The present invention is adaptive to many environments and will optimize throughput for systems that need to transfer information or other types of entities from point A to point C through intermediate point B. Example environments for utilizing the invention described herein include transfer of data via a communication channel, movement of people/equipment/goods via a transportation system, mail delivery scheduling, telephonic switching, etc.

An intermediate node of a multi-node system controls information flowing through it by queuing received information and transferring the received information to a subsequent node independent of the block size of the information being transferred. Subsequent blocks of information are transferred upon completion of a previous transferred block, rather than upon completion of an incoming block being received.

This procedure can handle mismatched flow problems from diverse environments and provides optimal flow for solutions that require guaranteed transfers. Better performing algorithms exist, but they cannot guarantee that the element being transferred will get from system A to C.

This procedure can change and adapt to varying circumstances. It can be altered in real-time for communication systems. If the block size changes the modification does not alter the smooth flow of the algorithmic procedure. The block size could be changed by a customer desiring to have real-time control over latency and throughput. In the case of IBM's FCS adapter, it may be desirable to expedite certain services and not others. It provides fine-tuned control over the data flowing through the system. When the setup time is very small, one could use a standard communication meter and small block size to get good results. However, if the setup time were sizable, the incurred overhead with a small block size would be very high. The invention disclosed herein is better in both cases, especially the latter.

It can be seen that the present invention provides an adaptive flow control system. In preferred embodiments the technique provides an efficient yet adaptive communication system, being able to match dissimilar path speeds used for transporting information.

The present invention will be described further, by way of example only, with reference to an embodiment thereof as illustrated in the accompanying drawings, in which:

Figure 1 is a block diagram of a system in accordance with the preferred embodiment of the invention, including a sending, intermediate, and receiving node;

Figure 2 is a flow diagram of a simple algorithm used to transfer information between nodes;

Figure 3 is a flow diagram of a standard algorithm used to transfer information between nodes;

Figure 4 is a flow diagram of an adaptive flow algorithm used to transfer information between nodes in accordance with the preferred embodiment of the invention;

Figure 5 is a block diagram of a multi-node environment, such as used in a switched telecommunication system;

Figure 6 is a typical data processing system, which can provide the functionality of a sending and intermediate node; and

Figure 7 is a block diagram of a communications adapter.

Referring initially to Figure 1, there are several parameters that should be defined before describing the preferred system and method.

- Systems A (10) and B (20) communicate over link AB (12) with link speed M.
- Block moves between A and B are of size < = x.
- Systems B (20) and C (30) communicate over link BC (14) with link speed N.
- Block moves between B and C are of any size. Blocks can be any quantity of items/people/information being conveyed or transferred between points.
- There exists a setup time for transfers between B and C of Ts.
- M and N are not necessarily equal.
- Y is the size of data transferred.
- Ttotal is the total time required in the transfer.

Simple Algorithm

Referring to Figure 2, the simplest technique for transferring data from A (10) to C (30) is to:

- Transfer x from A to B (at 22)
- When x arrives at B (24), transfer x to C (at 26) and send acknowledgement to A (at 28)
- If done (32), exit (34); else go to the beginning (22)

The equation for Ttotal = Y/M + Y/N + Y*Ts/x

Standard Algorithm

Referring to Figure 3, a technique at the next level of complexity would be:

- Transfer x from A to B (at 36)
- Dual transfer
  - When x arrives at B (38), send acknowledgement to A (40); when link BC clear (42), transfer x to C (44)
  - and, when acknowledgement received from B (46), transfer another x from A to B (36)
- If done(48), exit; else go to the dual transfer (36)

The equation for Ttotal = x/M + Y/N + Y*Ts/x

Adaptive Flow Algorithm

The adaptive algorithm employed in the preferred embodiment of the present invention uses the ratio of M to N, and a value p, where p = ceil(log(Y/x)/log(M/N))-1 and ceil() is the ceiling function. sigma-i(n) is the sum from j=0 to j=i of n raised to the jth power.

p + 2 is the total number of transfers for the adaptive algorithm. Referring to Figure 4, the adaptive algorithm flows as follows:

- Transfer x from A to B (52)

- First dual transfer
  - When x arrives at B (56); transfer x to C (58)
  - When x at B (54), send acknowledgement to A (60) to initiate (62) another transfer of x from A to B (52)
- Second through (p + 2)th dual transfer
  - Upon receipt of an acknowledgment from C (60), whatever is at B (designated by x' as determined at 56; where x' is larger or smaller than x, due to differing link speeds M and N), transfer that to C (58).
  - When x at B (54), send acknowledgement to A (60) to initiate (62) another transfer of x from A to B (52)
- If done, exit; else go do the ith transfer
- Node C, upon receipt of block x' (64), sends an acknowledgment to B (66). The determination as to whether the block x' has been received is made using any conventional technique known in the communication art for conveying a length of data being sent within the data packet, such as in a packet header file.

The equation for Ttotal using the adaptive algorithm is:

$$Ttotal = x/M + Y/N + ceil(log(Y/x)/log(M/N) + 1) * Ts$$

Formula Derivation

```
Ttotal = x/M + Ts + x/N                     1st transfer
          + (M/N) * x/N + Ts                2nd transfer
          + ...
          + (M/N)^i * x/N + Ts              ith transfer
          + ...
          + (M/N)^p * x/N + Ts              p+1st transfer
          + (Y - x * sigma-p(M/N))/N + Ts   p+2nd transfer
```

Solving for Ttotal:

When M/N != 1:

$$Ttotal = x/M + (p+2)*Ts + Y/N \text{ and}$$

$$p = ceil(log(Y/x)/log(M/N)) - 1$$

When M/N = 1:

The adaptive flow algorithm reduces to the standard algorithm.

Tables 1-4 demonstrate transfer times for various communication channel scenarios using the above described algorithms. Table 1 shows Ttotal for a 1 Megabyte file transferred using 1K blocks, where the channel speed between A and B is 50 Megabytes/second and the channel speed between B and C is 25 Megabytes/second. This table also shows two set-up time (Ts) examples (10 and 100 microseconds). Not only is the total transfer time less using the adaptive algorithm, but overhead is minimized. The overhead % $\triangleq$ (1 - (Ttotal / min(M,N) / Y)) * 100, where Y is the file size. The overhead ratio $\triangleq$ (overhead %)/(adaptive overhead %).

Table 1. Comparison of Algorithms: 1k blocks, 1M file, M=50MB/s, N=25MB/s

Algorithm	Ts, us	Total time,s	Overhead, %	Overhead Ratio
Simple	10	0.07	75%	230.8
Standard	10	0.05002	25%	77.1
Adaptive	10	0.04013	0.33%	1
Simple	100	0.16	300%	107.1
Standard	100	0.1401	250%	89.4
Adaptive	100	0.04112	2.8%	1

Table 2. Comparison of Algorithms: 4k blocks, 1M file, M=50MB/s, N=25MB/s

Algorithm	Ts, us	Total time,s	Overhead, %	Overhead Ratio
Simple	10	0.0625	60%	132.4
Standard	10	0.04258	6.5%	15.2
Adaptive	10	0.04017	0.425%	1
Simple	100	0.085	112.5%	45.9
Standard	100	0.0654	63.5%	25.9
Adaptive	100	0.04098	2.45%	1

Table 3. Comparison of Algorithms: 1k blocks, 100M file, M=50MB/s, N=25MB/s

Algorithm	Ts, us	Total time,s	Overhead, %	Overhead Ratio
Simple	10	7.0	75%	15000
Standard	10	5.00002	25%	5000
Adaptive	10	4.0002	0.005%	1
Simple	100	16	400%	6593
Standard	100	14.0001	250%	5495
Adaptive	100	4.00182	0.045%	1

Table 4. Comparison of Algorithms: 4k blocks, 100M file, M=50MB/s, N=25MB/s

Algorithm	Ts, us	Total time,s	Overhead, %	Overhead Ratio
Simple	10	6.25	60%	8654
Standard	10	4.25008	6.25%	962
Adaptive	10	4.00026	0.0065%	1
Simple	100	8.5	112.5%	2394
Standard	100	6.5004	62.5%	1330
Adaptive	100	4.00188	0.047%	1

Tables 2-4 similarly show various results when using the above described algorithms, for various file and block sizes.

The adaptive algorithm can be implemented using standard programming techniques as follows. One need only count the amount of data that has come from A to B (keep total at system/node B) while the transfer from B to C is occurring. Once the B to C transfer is complete, send the total accounted for data at B (the portion received and counted) on to C. Thus, only node B is concerned with the possibly dissimilar data rates of link AB and link BC. Further, the block size can be dynamically changed at A without disrupting the adaptive algorithm, as the actual block size being used in the transfer of information is not used by B when determining whether to send information to C. This greatly simplifies system design by consolidating the transfer decision at a single node independent of the actual block size being used. The block could be changed to allow greater control over the latency and throughput of a particular flow of information, or to expedite a particular item through the system. The block size would be changed at the sending node, either manually by a user or automatically by the sending node's controller or computer. As the other system node(s) queue and transfer information irrespective of the block size, this size can be dynamically changed by the sender.

As shown in Figure 5, the technique of the preferred embodiment of this invention could similarly be extended to a system having multiple intermediate nodes 80, such as in a switched point-to-point communication system, with the adaptive algorithm running in each intermediate node (a node other than the originating 78 or final 82 node). Thus, each intermediate node handles the data flow mismatch for its respective sending and receiving nodes.

Figure 6 shows the preferred embodiment data processing system 84, which comprises a CPU 90, read only memory 96, random access memory 94, I/O adapter 98, user interface adapter 102, communication adapter 114, and display adapter 116 all interconnected via a common data path, or bus, 92. Each of the above components accesses the common bus using conventional techniques known to those of ordinary skill in the art, and include such methods as dedicating particular address ranges to each component in the system, with the CPU being the bus master. Other conventional techniques known to those of ordinary skill in the art include direct memory access, or DMA, used to transfer data at high speed from external devices such as DASD 100 or network 110 to the data processing system's random access memory (RAM) at 94. As is further shown in Figure 6, these external devices 100 and 110 interface to the common bus 92 through respective adapters 98 and 114. Other external devices such as the display 118 similarly use an adapter 116 to provide data flow between the bus 92 and the display 118. User interface means are provided by adapter 102, which has attached thereto such items as a joystick 112, mouse 106, keyboard 104, and speaker 108. Each of these units is well known as such and so will not be described in detail herein.

Figure 6 corresponds to the logical functions of Figure 1 in the following manner. Link 12 between system A 10 and system B 20 corresponds to bus 92 of Figure 6. System A of Figure 1 is the sender of data, and could be any of CPU 90, RAM 94, or I/O adapter 98 of Figure 6. In the preferred embodiment, data is provided to the communications adapter 114 from RAM 94 using conventional DMA techniques across bus 92. Link 14 of Figure 1 corresponds to network 110 of Figure 6. System C 30 of Figure 1 corresponds to a similar communications adapter 114 in a similar data processing system 84 also residing on network 110. Other embodiments of this invention could similarly use entire data processing systems 84 at each of System A, B, and C of Figure 1, and interconnected using traditional communication techniques.

Figure 7 shows in greater detail the communication adapter 114, which enables the essential features of System B (Figure 1) in the preferred embodiment. The adapter 114 is comprised of a microcontroller 122 coupled to a buffer 124, a transceiver 120 and a transceiver 126. Microcontrollers are commonly known in the art, and comprise a CPU 121, read only memory 123 and random access memory 125. Transceivers are used to interface to bus or network protocols by inserting/extracting the actual data to be transferred, as well as handling status signalling, within the particular bus or network protocol, as is commonly known in the art. The transceiver 120 receives data at 12 from the bus 92 of Figure 6. The transceiver 126 is an optical transceiver, and link 14 is an optical fiber, although it is apparent that the system of the invention could employ any type of transport mechanism. When data arrives at transceiver 120, it is buffered at 124, and the CPU is notified at 128. The CPU 122 maintains a count of the number of bytes received across link 12. The CPU 122, upon receipt of an acknowledgment at 130 which arrived across link 14 from System C (Figure 1), can initiate at 132 a transmittal of buffered information 124 across link 14 using transceiver 126.

The adaptive flow algorithm can be generalized to solve problems outside of the communications environment. It can handle parts inventory/shipping problems, military troop movement, mail delivery scheduling, and many other real world mismatched flow problems. In each case, the user defines the given parameter x to yield an acceptable latency at the beginning, and then follows the algorithm to determine total flow time. The simple and standard algorithms each are O(n) overhead algorithms, whereas the adaptive flow algorithm is O(log(n)). Therefore, as n grows, the adaptive flow algorithm overhead time will grow as log(n) and the others will grown as n. For large n, the first two algorithms require considerable processing and overhead compared to the adaptive flow algorithm.

Claims

1. A method of operating an intermediate node (B) to receive an entity from a first node (A) and to transfer the entity to a second node (C), the first and second nodes being connected to the intermediate node by transmission links, the method comprising the steps of:
(a) receiving at the intermediate node (B) a first block of the entity sent by the first node (A);
(b) upon receipt of the block, initiating the sending of a subsequent block of the entity to the intermediate node;
(c) concurrently with step (b), transferring the first block of the entity to the second node (C);
(d) upon receipt of an acknowledgement from the second node (C), causing the intermediate node to transfer a portion of the entity to the second node (C), the portion transferred being all of the entity that has at the time of the transferral been received by the intermediate node (B) from the first node (A) since the previous transfer was made; and
(e) repeating steps (b) and (d) until all of the entity has been transferred.

2. A method as claimed in Claim 1, wherein the entity is information.

3. A method as claimed in Claim 2 wherein said information is of total length Y and comprises a plurality of blocks having a block length "x".

4. A method as claimed in Claim 3 wherein the block length "x" comprises a plurality of data bytes, and a count of the data bytes received at the intermediate node (B) is maintained in order to determine the length of the portion to be transferred at step (d).

5. A method as claimed in any preceding claim wherein the transmission link between the first (A) and intermediate (B) nodes operates at a different data rate to the transmission link between the intermediate (B) and second (C) nodes.

6. A method as claimed in Claim 5, wherein the portion transferred at step (d) has a length different to the block length of the blocks sent by the first node (A).

7. A method as claimed in any preceding claims wherein the initiating step (b) is carried out by sending an acknowledgement of receipt of each block to the first node (A).

8. A queuing system in an intermediate node (B) for receiving an entity from a first node (A) and transferring the entity to a second node (C), the first and second nodes being connected to the intermediate node by transmission links, the system comprising:
reception means in the intermediate node (B) for receiving a first block of the entity sent by the first node (A);
initiation means, responsive to the reception means indicating receipt of the block, for initiating the sending of a subsequent block of the entity to the intermediate node;
a transfer means, operating concurrently with the initiation means, to transfer the first block of the entity to the second node (C);
the transfer means further, upon receipt by the intermediate node (B) of an acknowledgement from the second node (C), transferring a portion of the entity to the second node (C), the portion transferred being all of the entity that has at the time of the transferral been received by the intermediate node (B) from the first node (A) since the previous transfer was made;
the initiation means and transfer means repeating their functions until all of the entity has been transferred.

9. A system as claimed in Claim 8, wherein the entity is information.

10. A system as claimed in Claim 9 wherein said information is of total length Y and comprises a plurality of blocks having a block length "x".

11. A system as claimed in Claim 10 wherein the block length "x" comprises a plurality of data bytes, and a count of the data bytes received at the intermediate node (B) is maintained in order to determine the length of the portion to be transferred by the transfer means.

12. A system as claimed in any of claims 8 to 11, wherein the transmission link between the first (A) and intermediate (B) nodes operates at a different data rate to the transmission link between the intermediate (B) and second (C) nodes, and the portion transferred by the transfer means has a length different to the block length of the blocks sent by the first node (A).

13. A system as claimed in any of claims 8 to 12, wherein the initiation means initiates the sending of the subsequent block by sending an acknowledgement of receipt of each block to the first node (A).

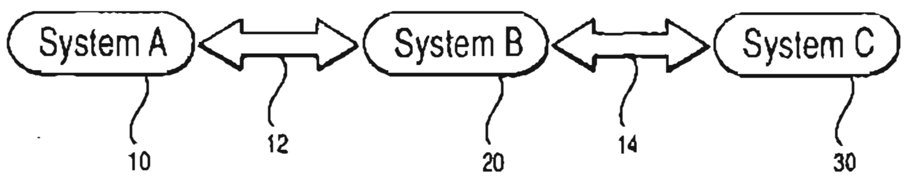

System A
System B
System C
10
12
20
14
30

FIG. 1

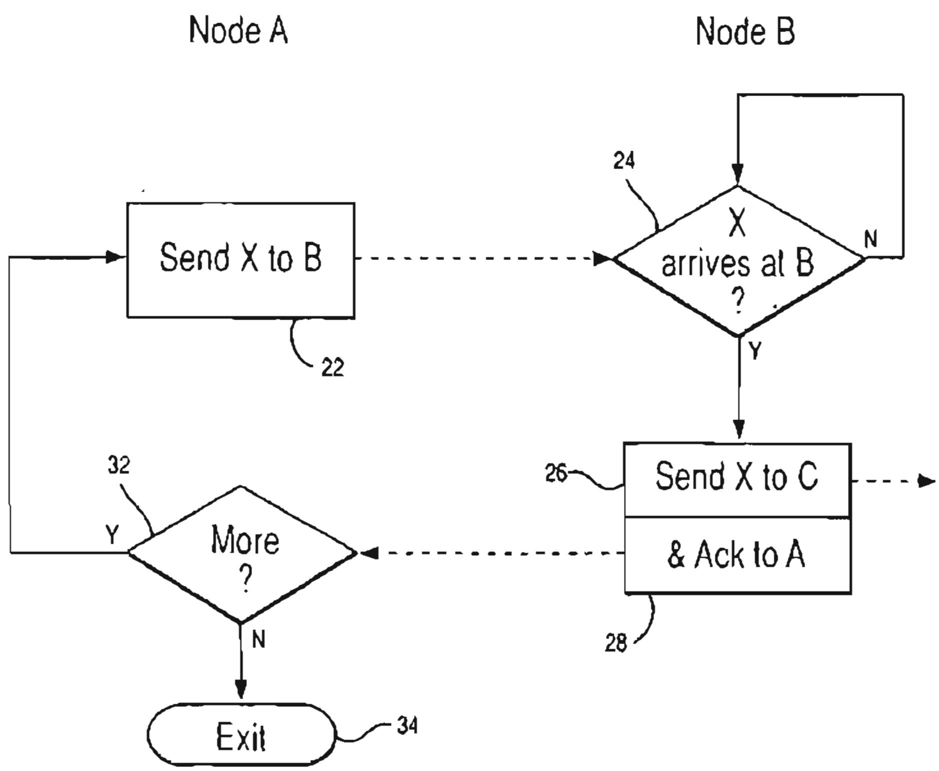

Node A
Node B
Send X to B
22
24
X
arrives at B
?
N
Y
26
Send X to C
& Ack to A
28
32
Y
More
?
N
Exit
34

FIG. 2

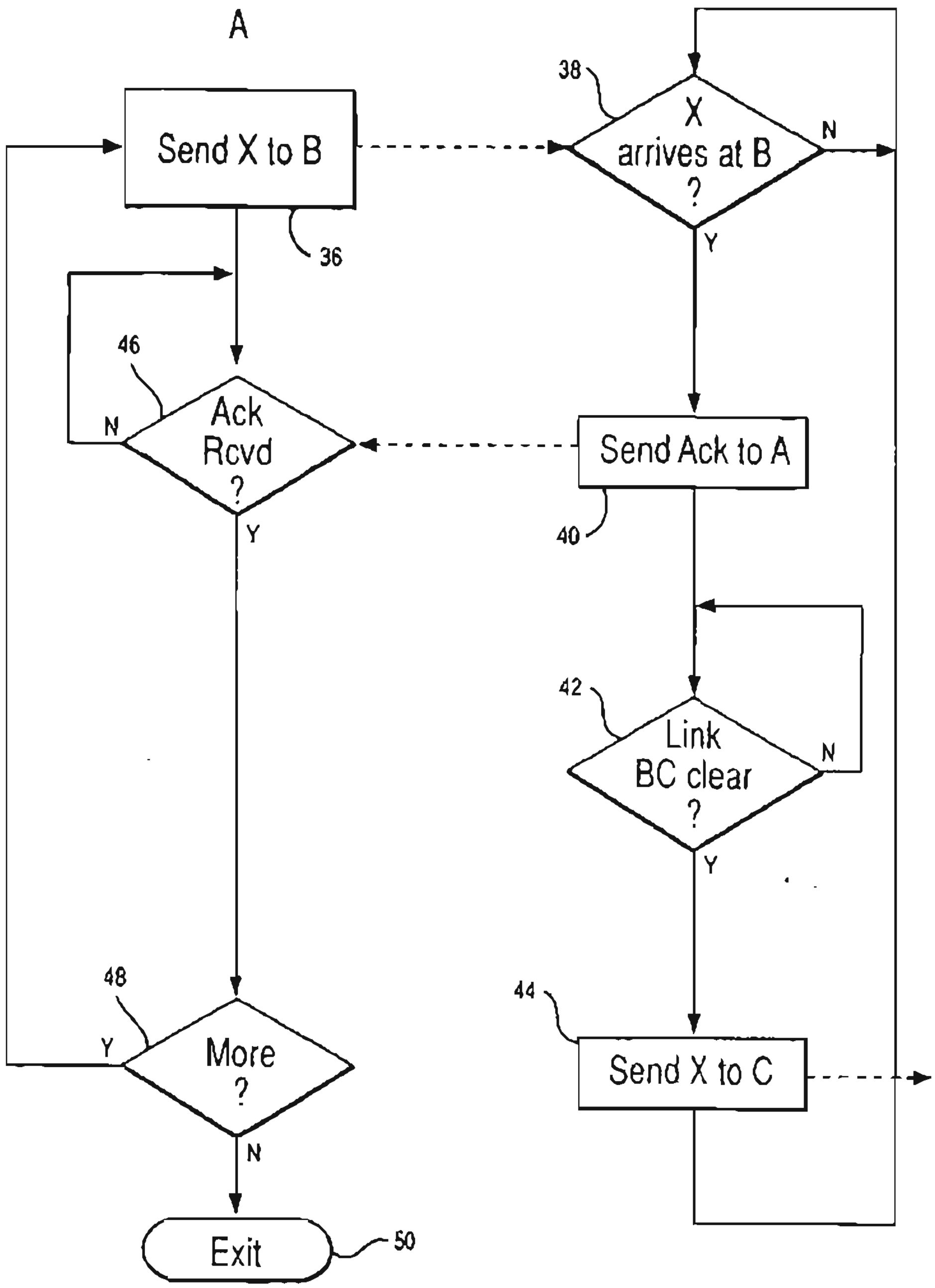

A
Send X to B
36
X arrives at B ?
38
N
Y
46
Ack Rcvd ?
N
Y
Send Ack to A
40
42
Link BC clear ?
N
Y
48
More ?
Y
N
44
Send X to C
Exit
50

FIG. 3

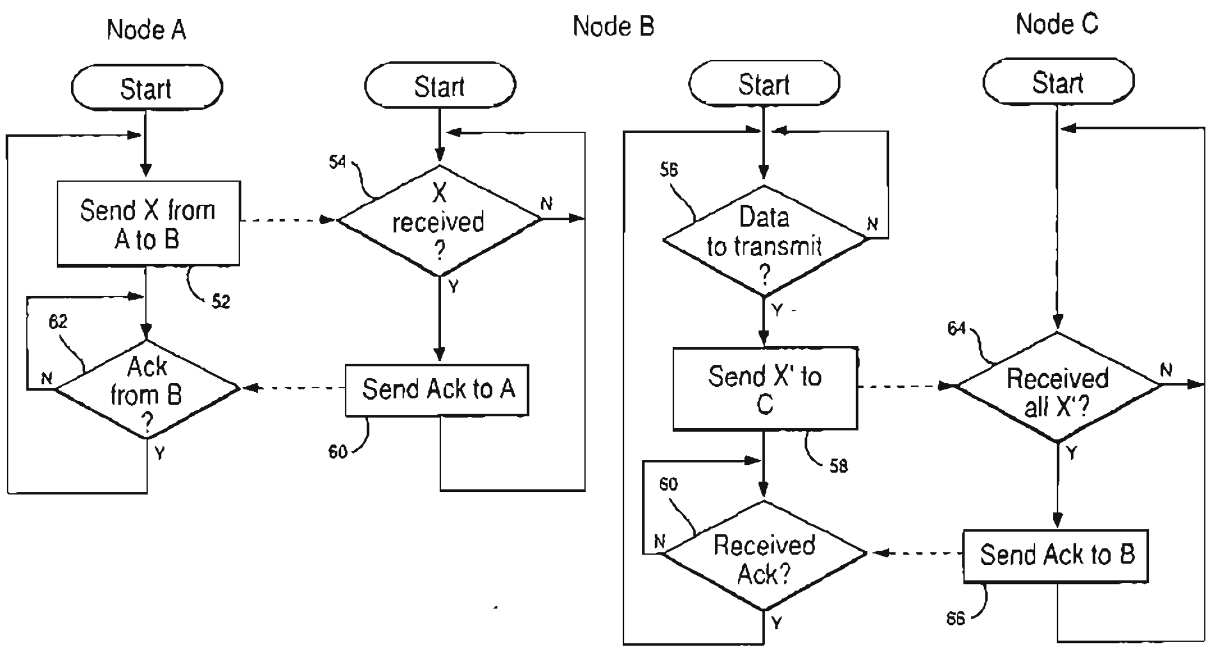

Node A
Node B
Node C
Start
Start
Start
Start
Send X from A to B
52
62
Ack from B ?
N
Y
54
X received ?
N
Y
Send Ack to A
60
56
Data to transmit ?
N
Y
Send X' to C
58
60
Received Ack?
N
Y
64
Received all X'?
N
Y
Send Ack to B
66

FIG. 4

FIG. 5

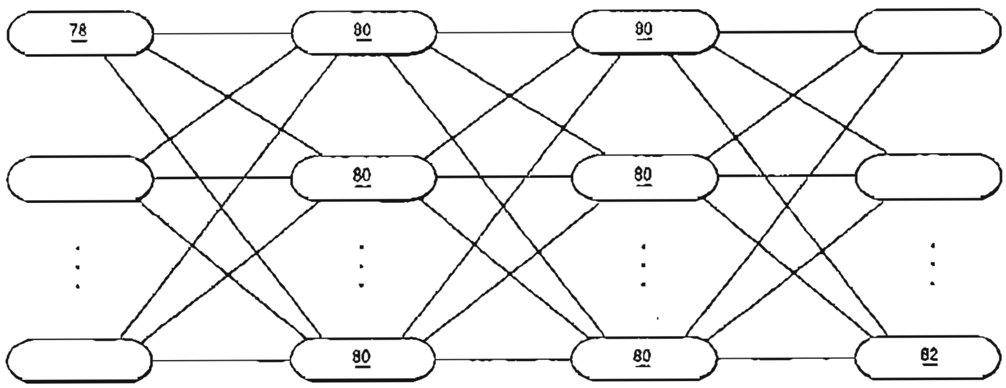
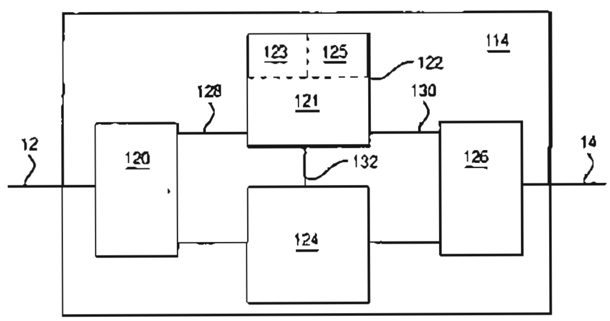

78
80
80
80
80
80
80
82

FIG. 7


123
125
122
114
128
121
130
12
120
132
126
14
124

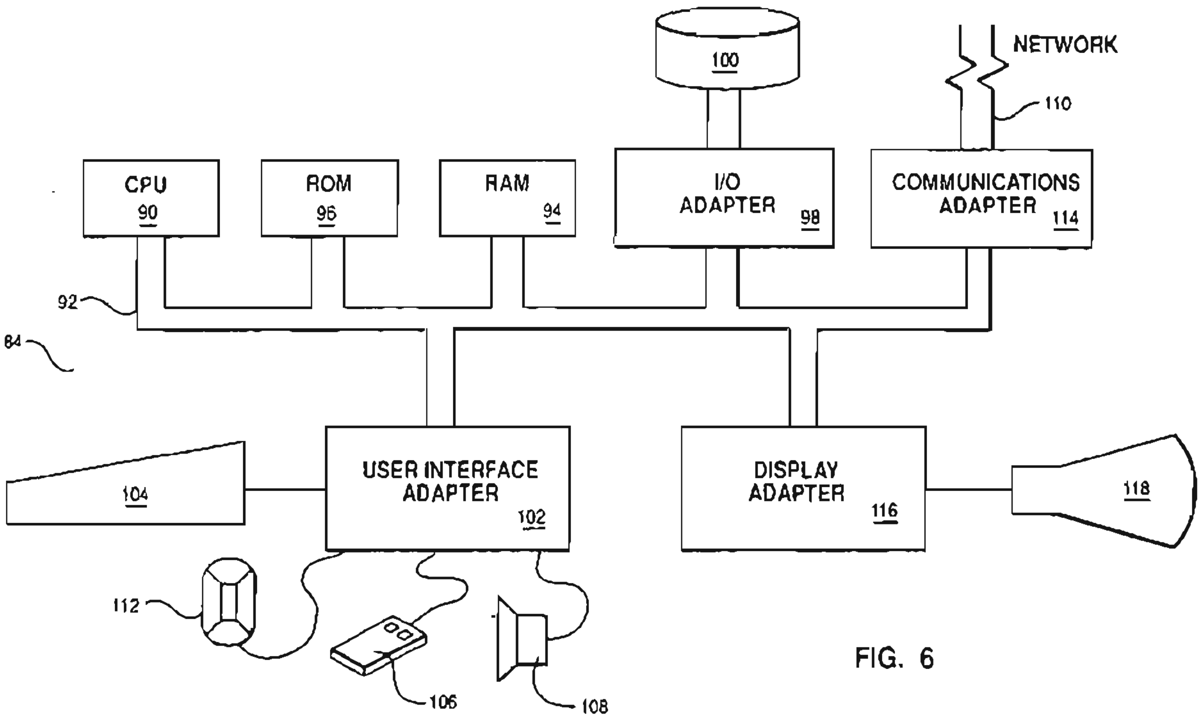

100
NETWORK
110
CPU
90
ROM
96
RAM
94
I/O
ADAPTER
98
COMMUNICATIONS
ADAPTER
114
92
84
USER INTERFACE
ADAPTER
102
104
DISPLAY
ADAPTER
116
118
112
106
108

FIG. 6

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 93 30 8975

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.5)
A	WO-A-84 03192 (AMERICAN TELEPHONE AND TELEGRAPH COMPANY) * page 32, line 8 - page 43, line 14 * * figures 1,6 *	1,2,8,9	G06F13/12
A	COMPUTER COMMUNICATIONS REVIEW, vol.21, no.4, September 1991, NEW YORK US pages 307 - 315 B. S. DAVIE 'A host-network interface architecture for ATM' * the whole document *	1,2,8,9	

TECHNICAL FIELDS SEARCHED (Int.Cl.5)

G06F
H04L

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
BERLIN	3 March 1994	Masche, C

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: 0 632 672 A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 94304718.3

(22) Date of filing: 28.06.94

(51) Int. Cl.[6]: H04Q 11/04

(30) Priority: 29.06.93 US 85264

(43) Date of publication of application: 04.01.95 Bulletin 95/01

(84) Designated Contracting States: DE FR GB

(71) Applicant: International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor: Baugher, Mark John
4813 Ramsey Avenue
Austin, Texas 78756 (US)
Inventor: Van Horn, Isabel Bardeen
12701 Marimba Trail
Austin, Texas 78729 (US)

(74) Representative: Moss, Robert Douglas
IBM United Kingdom Limited
Intellectual Property Department
Hursley Park
Winchester Hampshire SO21 2JN (GB)

(54) System and method for bandwidth reservation for different traffic classes.

(57) A system and method ensure transmission of data elements between computers at a preselected quality of service. A computer in a layered reference model communication network requests a preselected quality of service for selected transmissions. An existing session is employed whereby transmissions having a variety of preselected of service guarantees are multiplexed onto the existing session. Data elements having the preselected qualities of service are transferred from service channels of another such layer. In one embodiment a system and method are provided for use between layer N+1 and layer N, wherein elements from two or more service channels from N+1 are transferred to one or more service channels in layer N such that a preselected quality of service is provided for the elements being transferred. Elements from a service channel having a higher priority, deadline, or period are transferred from the layer N=1 service channel to the layer N service channel before elements from a service channel having a lower priority, deadline, or period. In yet another embodiment, the number of elements are constrained whereby no more than a fixed number of such elements will receive service in a layer N service channel at a given point in time.

EP 0 632 672 A2

Jouve, 18, rue Saint-Denis, 75001 PARIS

Technical Field

The present invention relates to data processing systems and, more particularly, to such systems providing for multimedia connections or sessions.

Background of the Invention

It has long been known to provide computer workstations interconnected by digital communication networks whereby users of the individual workstations may communicate with one another over the network, previously common, for example, by means of a typed note, data or program file transmitted to another user. More recently, users have increasingly requested desktop conferencing, remote presentations, and other multimedia applications between network users. However, such multimedia applications, having associated therewith data-intensive sound, voice, and video flows. This requires concomitant high bandwidth communication links between distributed computing systems with minimal communication delay, maximum throughput, and instantaneous burst communication capability. The requirements of such multimedia applications accordingly make scheduling appropriate resources to provide for necessary quality of service very difficult.

Prior art has recognized that certain data in a network, such as that associated with multimedia, may require priority handling. Thus, for example, a "quality of service" (QOS) has been defined in the literature, hereinafter described in more detail. This seeks to describe various parameters which may be specified in an attempt to define certain minimum requirements which must be met for transmission of given data types over the network. See, for example, quality of service standards set forth in the Open System Interconnect Standard X.214 of the International Standards Organization interface and the quality of service standards defined in CCITTQ.931 (ISDN), Q.933 (frame relay), and Q.93B (B-ISDN ATM) drafts.

As yet another example there is an architected priority mechanism in the IEEE 802.5 Token Ring. A station on the ring with a high priority frame to send may indicate this in an access control field of a passing frame. When a station sending the frame releases the token, it releases the token at the priority of the AC field, and eventually sets it back to its original priority as specified in an IEEE 802.5 medium access control protocol. The IEEE standard and implementations thereof merely specify a protocol for increasing and decreasing priority, but each station is unconstrained in its use of priority beyond this protocol.

This in turn gives rise to a serious problem associated with the prior art. In seeking to accommodate situations in which a high priority channel is required to guarantee real time service for multimedia traffic, one approach, since each station is unconstrained, has been for users to indiscriminately increase the priority of their flows. Such increases often result in no guarantees for multimedia quality of service in that no discrimination is provided, e.g. all users simply increase their priorities.

As yet another example of this, unconstrained use of priorities has resulted in bridges and routers loading so much high priority data as to flood the token rings and the like with this priority traffic such that multimedia traffic obtains no guaranteed priority. Again, this results from no discrimination between differing connections, sessions, and transmit operations.

Clearly other instances in the communication art have recognized the notion of a need for differing priority of data types, whether in the form of multiple channels with different priorities (such as the IBM LAN Streamer Token Ring Adapter Card with two transmit channels, and the 100 Mbps Ethernet System with priority channels) and the synchronous/asynchronous approach of, for example, the FDDI standards, a representative example of which is the FDDI SMT 7.X.

Moreover, it is clear in the literature that the notion of scheduling data in differing priorities is well known. See for example Liu and Layland, Scheduling Algorithms For Multiprogramming in a Hard-Real-Time Environment, Journal of the Association for Computing Machinery, Vol. 20, #1, January, 1973, where "rate-monotonic priority assignment" is discussed, page 50. Also see, for example, Dominico Ferrari, A Scheme for Real Time Channel Establishment in Wide Area Networks, IEEE Journal of Selected Areas in Communications, Vol. 8, #3, April, 1990, page 368. In this reference modification of an earliest due date (EDD) policy is presented which governs differing levels of priority assigned to tasks.

Similarly, the notion of specifying performance requirements in real time communication services is further addressed in another reference to Dominico Ferrari, Client Requirements for Real-Time Communication Services, IEEE Communications Magazine, Nov. 1990, page 65, wherein it is noted that a client and server will negotiate a specification for their respective requirements for services including delay bounds, throughput bounds, and the like.

From the foregoing it is clear that notions in the art have developed of varying degrees of priority service (based upon deadlines variously computer as burst/throughput or as a specified delay bound) being required in communication networks. However, several problems have remained in implementing a successful system which addresses the needs for guaranteeing real time service for multimedia traffic. First there is the aforementioned problem of lack of discrimination amongst the traffic whereupon users indiscriminately simply designate all their traffic to a higher priority, thereby "congesting" the network.

Yet another problem not effectively addressed by the prior art relates to the emergence of heterogeneous networks from differing vendor implementations of multimedia sessions. Their equipment has differing capabilities e.g. speed and capacity, and make quality of service guarantees problematic since over-reservation (or overcommitment) of resources can result in long-term throughput or short-term delays to violate pre-specified QOS. To be practical, this requires that in providing for reserved bandwidth connections, a solution must be provided which minimizes changes to application program interfaces and underlying client implementations. Yet another problem relates to failure to provide for multiple priority queues or channels extending through multiple layers since end-to-end delivery among digital computing devices requires services of multiple layers of the OSI model. The link layer guarantees service between two ends of a digital communication link and network over multiple links. Transport services at the endpoint must all provide QOS guarantees using priority service to meet deadlines.

Thus, particularly with the proliferation of multimedia data content, the industry was in need of a communication system providing for multimedia flows with quality of service guarantees at the transport and network layers utilizing reserved bandwidth networks whereby the session might be controlled to ensure that traffic from the session remains within traffic descriptors.

Summary of the Invention

According to the present invention there is provided a system and method for effecting multimedia quality of service sessions in a communication network.

Accordingly, there is provided a method for transmitting data elements between computers of a multilayered computer communication network at least at one predetermined assured quality of service, comprising:

generating a request at one of said computers for transmission of data elements; and

responsive to said request including specification of a predetermined quality of service for transmission, either refusing said quality of service if resources to provide said quality of service are not available or transmitting said data elements according to said quality of service wherein the transmission of said data elements includes transferring said data elements having said quality of service requirements through at least one service channel of a first layer of said network to at least one service channel of a second lower layer of said network which provides said predetermined quality of service.

In a second aspect of the invention there is provided apparatus for use in transmitting data elements between computers of a multilayered computer communication network at least at one predetermined assured quality of service, comprising:

means for generating a request at one of said computers for transmission of data elements; and

means responsive to said request including specification of a predetermined quality of service for transmission, for transmitting said data elements according to said quality of service wherein the means for transmitting said data elements includes means for transferring said data elements having said quality of service requirement through at least one service channel of a first layer of said network to at least one service channel of a second lower layer of said network which provides said predetermined quality of service.

The invention is embodied in computerized system components implementable in hardware or software or in combination. In a preferred embodiment the components regulate access to priority queues or transmission channels which are attached to the shared medium local area network or any point-to-point digital communications link over copper, fibre, radio or satellite transponders.

In one embodiment, a system and method are provided for use in an OSI layered reference model computer communication network for ensuring that transmissions of data elements between computers obtain a preselected quality of service. A computer in the network requests from another such digital service a preselected quality of service for selected transmissions. A determination is made if resource is available within the computers and network to meet the preselected quality of service. An existing session or connection between the computers is then employed such that transmissions having a variety of preselected quality of service guarantees, even commingled with such transmissions having no quality of service guarantees, are multiplexed onto the existing session or connection. In one implementation of the invention, the quality of service is altered in the session or connected by means of a session-modify command.

The priority, deadline, or period of each particular transmission on the session or connection is identified prior to transferring data elements having the preselected qualities of service on the session or connection. In one embodiment, the priority, deadline, or period is a computed function of a ratio of the burst and throughput quality of service parameters, and the session or connection transmission is effected by a reserved-send command. The data elements having the preselected qualities of service are thence transferred from service channels at one layer of the OSI layered reference model to one or more service channels of another such layer. The lower layer service channels are selected from a group comprising priority token ring service channels, B-ISDN ATM service channels (corresponding to services in the ASM adaptation layers), 100 Mbps Ethernet high and low priority channels, and synchronous/asynchronous FDDI service channels.

In another embodiment of the invention, wherein a pre-established session has been transferring data elements without a quality of service requirement, upon occurrence of the need for a quality of service guarantee, a new session or connection is established, in one form employing a call-modify command. Data elements belonging to this next session or connection having the preselected quality of service requirement are thence transferred from one or more service channels at one layer of the OSI layered reference model to one or more service channels of another such layer.

In both embodiments, a system and method are provided for use between a layer N+1 and layer N in an OSI layered reference model implementation, wherein elements from two or more service channels from layer N+1 are transferred to one or more service channels in layer N, such that a preselected quality of service is provided for the elements being transferred. A reserved-send-with-specified-priority operation is provided in which the reserved-send is augmented with a parameter that describes the urgency, deadline or period of the transmission, e.g. throughput/burst.

More particularly, elements from a service channel having a higher priority, deadline, or period are transferred from the layer N+1 service channel to the layer N service channel before elements from a service channel having a lower priority, deadline, or period.

In yet another embodiment, the number of elements from a service channel having a layer priority, deadline, or period transferred from the layer N+1 service channel to a layer N service channel are constrained whereby no more than a fixed number of such elements will receive service in a layer N service channel at a given point in time.

Brief Description of the Drawings

Fig. 1 is a block diagram showing the configuration of a typical workstation in accordance with the subject invention;

Fig. 2 is an illustration of a data processing system including three workstations interconnected by a network in accordance with the subject invention;

Fig. 3 is a representation of a layered open systems interconnection model showing the relationship of components of the subject invention to the layers;

Fig. 4 is a simplified illustration of a workstation network interconnection in one embodiment of the invention providing for separate sessions having differing priorities;

Fig. 5 is another simplified illustration of a workstation network interconnection in another embodiment of the invention providing for multiplexed multimedia flows on a single session having differing priorities;

Fig. 6 is a simplified illustration of a portion of the open system interconnection layered reference model for Fig. 3 depicting the abstraction of multiple layers with corresponding service access points.

Fig. 7 is a simplified illustration of a layered service channel model portion of the open system interconnection model of Fig. 3.

Fig. 8 is a flowchart detailing the program logic in accordance with one embodiment of the invention;

Figs. 9A and 9B is a flowchart detailing the program logic of another embodiment of the subject invention;

Detailed Description of the Preferred Embodiment

Referring now to Fig. 1, there is illustrated a typical hardware configuration of a workstation with a central processing unit 10, and a number of other units interconnected via a system bus 12. The workstation shown in Fig. 1 includes a random access memory (RAM) 14, read only memory (ROM) 16, and I/O adapter 18 for connecting peripheral devices such as disk units 20 to the bus, a user interface adapter 22 for connecting a keyboard 24, mouse 26, loudspeaker 28, microphone 32, and/or other user interface devices to the bus, a communication adapter 34, for connecting the workstation to a data processing network, and a display adapter 36 for connecting the bus to a display device 38.

Fig. 1 depicts a typical "intelligent" workstation, however, the workstation may in fact be a "dumb" terminal with only a limited processing capability under control of a host processor. Alternatively, the workstation may be a simple digital device for presenting audio or video streams. This is made clear in connection with Fig. 2.

Fig. 2 illustrates a data processing system comprising a number of workstations (here, three workstations 200, 220, and 230) interconnected by a pair of data networks 210 and 240, so as to permit communication between the workstations. It is assumed that the data processing system shown in Fig. 2 is of a type which permits concurrent real-time communication between the users. The network operates according to a conventional network protocol, such as the token ring protocol described in Token Ring Network Architecture reference, SC30-3374, IBM, 1989.

Fig. 2 depicts only one possible hardware configuration for a data processing network. Other configurations are possible. For example, the data processing system could be based upon a star network, or a host processor connected to a plurality of dumb terminals, or could further be based upon a plurality of remote processors connected by a communication network. The networks could also be based upon a telephone network, an ISDN network, or any other "dial up" networks. Moreover, the workstations could be located within the single workspace or within a local area, or could be remote from one another. A source for detailing technical planning information for configuring a network of workstations in accordance with the invention, is the IBM Extended Services for OS/2 Example Scenarios Manual, 1991.

Multimedia computing is the processing of various media, such as video, waveform audio, musical instrument digital interface (MIDI) streams, animation, graphics, and text. Such processing includes the capture, authoring (editing) and playback of media streams as well as other data processing applications. Multimedia documents which are stored on some non-volatile medium, such as a disk, are referred to as recorded multimedia applications. There are also live multimedia applications in which two or more people communicate with each other at the same time using a computer. Live multimedia applications are normally conducted across space and time indicating that live multimedia is inherently distributed. Even recorded multimedia applications require distributed file system services to share large volumes of stored media, such as video disk, audio information, or computer-generated images. Thus, it is critical that a prioritizing scheme in accordance with the invention for multimedia applications includes support for a distributed environment.

To reduce design complexity, most networks are organized as a series of layers, each one built upon its predecessor as described in Computer Networks, Tannenbaum, Andrew S., Prentice Hall (1988) and OSI, A Model for Computer Communications Standards, Black, Ulyess, Prentice Hall, 1991. The number of layers, the name of each layer, contents, and function of each layer differ from network to network. However, in each network, the purpose of the layers is to offer certain services to the higher layers, shielding those layers from the details of how the offered services are actually implemented. The purpose, function, and details of each of the layers and their interaction is set forth in the previously noted references and is familiar to communication programmers ordinarily skilled in the art.

The transport layer accepts data from the session layer, splits it up into smaller units and passes the units to the network layer to ensure that the pieces all arrive at the other end. Details of the transport layer and how it fits into the OSI architecture are shown in Fig. 18 of the Tannenbaum book and described in the related pages. A representative of network architecture that provide technical standards documents for the networking framework are ISO/IEC JTC 1/SC 21 Information Retrieval, Transfer and Management for OSI Secretariat: USA (ANSI) (3294) Basic Reference Model Management Framework (7498-4), and Management Information Model (3324) ISO, 1989.

One way of looking at the transport layer is to regard its primary function as enhancing the Quality of Service (QOS) provided by the network layer. QOS can be characterized by a number of specific parameters. The OSI transport service allows a user to specify preferred, acceptable, and unacceptable values for these parameters when a connection is made. Some of these parameters also apply to connectionless transports. The transport layer examines the parameters, and depending upon the kind of network services available to it, determines whether the transport layer can provide the necessary service. Representative QOS parameters are:

Connection Delay, which is the amount of elapsed time between a transport connection being requested and confirmation being received by the user, which, as with all parameters dealing with delay, the shorter the delay the better;

Connection Establishment Failure Probability is the probability of a connection not being established within the maximum establishment delay time. Network congestion, lack of table space, and other internal problems affect this value;

Throughput measures the number of bytes of user data transferred per second as measured over a recent time interval, and is measured separately for each direction;

Transit Delay measures the time between a message being sent by the transport user on the source machine and its being received by the transport user on the destination machine;

Residual Error Rate measures the number of lost or garbled messages as a fraction of the total sent in the sampling period. In theory, the residual error rate should be equal to zero since it is the job of the transport layer to hide all network layer errors.

Yet additional QOS parameters includes;

Transfer Failure Probability which measures how well the transport service is living up to its assigned tasks. When a transport connection is established, a given level of throughput, transit delay, and residual error are agreed upon. The transfer failure probability gives the fraction of times that these agreed upon goals were not met during some time period;

Connection Release Delay is the amount of time elapsing between a transport user initiating a release of a connection and the actual release occurring at the end;

Connection Release Failure Probability (CRFP) is the fraction of a connection release attempts which did not complete within the agreed-upon connection release delay interval;

Protection provides a way for the transport user to specify interest in having the transport layer provide protection against unauthorized third parties reading or changing reading or changing transmitted information;

Resilience yields the probability of the transport layer spontaneously terminating a connection due to internal problems or congestion.

The hereinbefore-noted QOS parameters are specified by a transport application when a connection is requested. Both the desired, minimum, and maximum acceptable values are given. In some cases, the transport layer immediately recognizes that the values are not achievable. When this occurs, the communication attempt fails, and an appropriate exception is noted. In other cases the transport layer knows that it cannot achieve the desired goal, but can achieve a lower, but still acceptable rate. The lower rate, minimum acceptable rate, and maximum acceptable rate are sent to the remote machine requesting the established of a connection. If the remote machine cannot handle the proposed value, but can handle a value above the minimum or below the maximum, then it may lower the parameter to its value. If it cannot handle any value above a minimum, then it rejects the connection attempt. Then, the originating transport application is informed of whether the connection was established or rejected.

This process is called open negotiation. Once the options have been negotiated, they remain that way through the life of the connection. The OSI Transport Service Definition, (ISO 8072) does not specify the QOS parameters. These are normally agreed upon by a carrier and customer. A T-connect request is employed to initialize communication, and the QOS is specified as part of this transactions. Details on the transport primitives are found in the aforementioned reference. Below the transport and network layers are the link or MAC layers in the OSI and IEEE 802 reference models to be hereinafter described. Some MAC protocols, such as synchronous FDDI, provide guarantees for throughput, delay, and delay variation to applications. Other MACs such as the Token Ring and Token Bus have architected priority mechanisms which can support quality of service guarantees (throughput, delay, etc.) when the subject invention is employed.

Priority assurance is an important factor in ensuring QOS, and is enabled by operation of a component which may be implemented in hardware logic or software. The component regulates access to the priority queues or transmit channels that are attached to the shared medium local area network section. Access to the priority queue or transmit channels will pass through this component, thus subjecting all communication transactions to rejection or tracking by the component. A more detailed discussion of this component and the related station's bandwidth manager component are described in Network Priority Management, U.S. Patent Application, (AT9-92-089) S/N 07/930,587, filed August 17, 1992.

Turning now to Fig. 3, depicted there is a schematic representation of several forms of a multilayered computer communication network model based upon the OSI layered reference model. Further detail of this OSI and related IEEE models may be found in OSI, A Model for Computer Communications Standards, infra. The first five layers of the OSI model are shown in Fig. 3 as reference numerals 40-56. The lowest layer is the physical layer OSI 1, 56, which is responsible for implementing a physical circuit between data terminal equipment and data circuit terminating equipment.

The data link or second layer, OSI 2, 54, is responsible for transfer of data across the link. The third or network layer, OSI 3, 52, specifies the interface of the user into a network and also defines network switching/routing and communications between networks. The fourth or transport layer, OSI 4, 50, provides an interface between the data communications network and the upper three layers. This layer is of particular interest inasmuch as it provides the user options in obtaining certain levels of quality, and is designed to keep the user isolated from some of the physical and functional aspects of the network.

The fifth or session layer, OSI 5, 48, serves as a user interface into the transport layer below, providing a means for exchange of data between users such as simultaneous transmission, alternate transmission, checkpoint procedures and the like. The remaining two layers, the presentation layer and application layer (not depicted), ensure that user applications can communicate with each other and further concern the support of the end-user application process.

It will be noted from Fig. 3 that there are other implementations in the art of such an OSI reference model bearing varying degrees of similarity thereto, a portion of one being depicted in the left part of Fig. 3 as the IEEE model. A physical layer 46 may be seen corresponding to the first layer 56 of the OSI model. The IEEE recognized the need to divide the data link layer OSI 2, 54, into two sublayers in order to handle different link configurations and thus a medium access control (MAC), 44, and logical link control (LLC), 42, were provided for. The IEEE model is specialized to links which are shared media, e.g. having more than two connected stations. The MAC sublayer is protocol-specific (such as to a LAN such as Ethernet) whereas the LLC, 42, serves as an interface to an upper layer protocol, typically the network layer (and isolates the network layer from the specific actions of the MAC sublayer). One purpose of depicting varying forms of a multilayered computer communication network in Fig. 3 is to illustrate that the invention admits to implementations in any number of such multilayered models, and is thereby not intended to be limited to application to the OSI reference model emphasized in the description herein.

Still referring to Fig. 3, a more practical implementation of the invention is shown in the righthand portion thereof as it relates to the theoretical reference models just described. More particularly, there is first shown an adapter card 62, implementing the physical layer and part of the MAC of the reference model. In a preferred form, this adapter card will provide for a plurality of priority channels, services or queues, as illustrated by the P0 and P5 channels. The figure is further intended to indicate that the adapter card 62 will be for at least two channels through which data elements may flow, each such channel having a differing priority. Although a P0 and P5 have been shown, the invention admits to implementations with more priority channels as required (with, conventionally, the right or higher numbered channels such as P5 indicating higher priority whereby data elements transferred through this channel will be transmitted in preference over those at a lower priority. A media access control software driver 60 is also shown implementing the MAC sublayer 44 in the IEEE model, for example, and finally a netbios or functionally equivalent form of LAN software, 58, is shown implementing the session layer 5 of the OSI reference model.

It is important to note that the invention contemplates implementation of multiple service channels in the lower layers of the architecture in a number of forms and is not intended to be limited to any particular such implementation. Thus the channels depicted with associated priorities P0 and P5 may be implemented as priority token ring service channels, B-ISDN ATM service channels, 100 Mbps Ethernet high and low priority channels, and even as lower layer service channels in the synchronous and asynchronous FDDI form of priority channels or queues.

Still referring to Fig. 3, as will be described hereinafter in greater detail, yet an additional feature of the invention is shown depicted therein. In a preferred embodiment, once a session has been established, shown at reference numeral 64, data elements having a preselected QOS belonging to the session or connection from one or more such just described service channels at one layer of the OSI layered reference model (such as the netbios layer 58), will be transferred to one or more service channels of yet another layer such as that of the MAC driver 60 or physical layer 62. This is schematically depicted by the line 64 splitting into two parallel downward arrows culminating in the two channels P0 and P5 having differing priorities, shown as arrows 66 and 68.

In another form of the invention, a pre-existing session shown as arrow 70 may have been established which may not have required a quality of service. Such a session may have been established by a call command. When a determination is made by the system that a multimedia transmission requiring a guaranteed quality of service is required, a new session may be established by a call modify, shown by the arrow 72 extending downwards through the various reference model layers and terminating in the P5 channel. This is intended to indicate that when such multimedia data transmissions are required, in implementations allowing for only one session per file or connection, this form of the invention will provide for a next session operating at higher priority to effect the desired quality of service guarantee at, for example, the higher P5 priority.

This latter form of the invention may be seen more clearly depicted in Fig. 4. In this embodiment there may be various forms of data in need of transfer shown generically as file 1, 84, and a second device 2, 86, each form of data having associated therewith either no quality of service requirement or some form of quality of service guarantee dependent upon the data type. For example, device 2 might be a real time video capture card having a relatively high throughput requirement of perhaps 150 KBS, whereas the data associated with file 1 may be digitized audio requiring a lesser QOS or perhaps even a binary executable file or text file with no particular required QOS.

As previously described, a first session 88 may have previously been established between two computers 80 and 82 on the network for transmission of non-QOS data such as that from file 1. Upon determination of the capability of the network to transfer QOS data, in one embodiment, a next session (shown schematically at reference numeral 90) may thereafter be established between the computers 80-82 wherein a QOS is bound to this next session 90 to carry the multimedia data associated with device 2, 86, at a guaranteed QOS.

Turning now to Fig. 5, this figure is intended to more practically represent the multiplexed flows of data at varying qualities of service at a single session, as briefly hereinbefore with reference to the rightmost portion of Fig. 3. In this implementation, again data having differing quality of service requirements (or none) may be seen graphically illustrated as file 1, 104, and device 2, 106. In this preferred embodiment, however, a session may have been established between computers 100 and 102 shown by line 108. It is contemplated that data elements corresponding to differing QOS needs may nevertheless be transmitted or multiplexed on the same single session 108. In this manner, the aforementioned real-time video captured by device 2, 106, may be given a high priority and transmitted on the session 108 in preference to but nevertheless multiplexed with lower priority data associated with file 1, 104.

Turning now to Figs. 6 and 7, it is an important feature of the invention that data elements belonging to sessions or connections having a preselected quality of service from one or more service channels at one layer of a layered reference model may be transferred to one or more service channels of another layer.

Thus, turning now to Figs. 6 and 7, they are intended to represent a more generalized case than that of Fig. 3, and represent the notion that multiple channels or priority queues may be provided at a plurality of layers providing quality of service guarantees.

Fig. 6 is an OSI depiction of a service access point (SAP) which is an addressable unit by which layer N provides service access to a layer N+1 user. The invention extends or replaces this abstraction by which a layer N service offers a multiplicity of service priorities that correspond to the deadline, period or quality of service to be provided to a layer N+1 user.

Turning now to Fig. 8, a flow diagram is provided which may be implemented in software for use by the network system in effecting one embodiment of the invention. First, a request for a quality of service transfer 130, is placed on the network 210 or 240 by one of the workstations 220 which may be a client or an application program. The QOS request is received, 132, by the host or server 220 or communications transport product, which then determines whether sufficient QOS resources are available in the network to service the request 134. If not, as indicated as block 136, the QOS session or connection is refused. If, on the other hand, sufficient resource is available, the QOS connection is established, 138, and transfer of data commences utilizing the aforementioned plurality of QOS service channels extending through the multiple layers of the reference model 140, in accordance with the invention.

Turning now to Figs. 9A and 9B, a more detailed flow diagram for an alternate embodiment of the invention is provided wherein elements from a service channel having a higher priority, deadline or period are transferred from a layer N+1 service channel to layer N service channel before elements from a service channel having a lower priority, deadline, or period are transferred. First, as in the case of the prior embodiment, one of the network computers may generate a QOS request 142, which is then received by the host or server 144. Again, the server or host determines whether QOS resources are available, 146, to service the request. If this is not the case, QOS guarantees are not provided. Upon a subsequent data transfer request 150, the server or host determines whether QOS service channels are empty, 152, and if not, a polling procedure is entered looping back on the block 152 for the channel to empty so that lower priority data may be transferred. Upon the QOS service channel's emptying, data is then transferred over non-QOS service channels 160.

If a QOS request has been received, 144, and QOS resources are available, 146, one or more priority channels will be reserved such that a QOS guarantee may be made, 154. Upon a subsequent data transfer request 156, the server or host will determine whether QOS data is involved, 158. If not, the hereinbefore described process relative to blocks 152 and 160 will be repeated. If, on the other hand, QOS data is involved in the data transfer request 158, the host or server will thereafter identify the period, priority, or deadline associated with the desires QOS transfer and thereafter will identify an appropriate service channel 164. The host or server thereafter will determine whether service channel constraints exist, 166. If so, the system next determines whether the service channel is at the predetermined limit 168, in which case again a polling loop is entered, looping back on the block 168 determination. When the service channel is detected as not being at a limit, the system then detects whether higher priority QOS service channels are empty, 170. If not, again a polling routine is entered looping back on the block 170 determination until higher priority traffic is handled whereupon flow exits to the right of block 170 and data is transferred in the desired manner on the service channels 172.

Referring back to block 166, if there were no service channel constraints 166, the logic flow exits from the right of block 166 to the subsequent determination of whether a higher QOS service channel or channels are empty, 170, and the process thence continues in the manner previously described.

From the foregoing relative to Figs. 9A and 9B, it is apparent that no attempt is made to put lower priority traffic in a higher priority channel. Rather, when two channels from layer N+1 to a single channel at layer N exist, in a preferred embodiment the number of lower priority requests will be constrained which go into the layer N channel from the layer N+1 lower priority service channel. In this manner, the system may determine how long a higher priority service channel request will have to wait, at maximum, to obtain access to the layer N service channel. The system moreover avoids putting lower priority requests into the service channel when there are higher priority requests waiting to go to the same channel.

While the invention has been shown and described with reference to particular embodiments thereof, it will be understood by those skilled in the art that the foregoing and other changes in form and detail may be made therein without departing from the spirit and scope of the invention.

Claims

1. A method for transmitting data elements between computers of a multilayered computer communication network at least at one predetermined assured quality of service, comprising:

   generating a request at one of said computers for transmission of data elements; and

   responsive to said request including specification of a predetermined quality of service for transmission, either refusing said quality of service if resources to provide said quality of service are not available or transmitting said data elements according to said quality of service wherein the transmission of said data elements includes transferring said data elements having said quality of service requirements through at least one service channel of a first layer of said network to at least one service channel of a second lower layer of said network which provides said predetermined quality of service.

2. A method according to claim 1 including, responsive to said specified quality of service being provided by pre-existing data transmission session, multiplexing said data transmission at said quality of service on said pre-existing session.

3. A method according to claim 2, wherein said data transmission is multiplexed with other data transmissions which either also have quality of service guarantees supported by said service channel of said second layer or have no predetermined quality of service.

4. A method according to claim 2 or claim 3, wherein said pre-existing session includes a preselected quality of service transmission, and wherein said method includes the further step of altering said quality of service for a next transmission within the session.

5. A method according to any one of the preceding claims, including establishing a data transmission session providing said at least one quality of service when said specified quality of service is not provided by any pre-existing data transmission sessions.

6. A method according to any one of the preceding claims, wherein said at least one service channel of said second layer is comprised of synchronous and asynchronous channels.

7. A method according to any one of the preceding claims, wherein said at least one channel of said first and second layers comprises multiple priority channels.

8. A method according to claim 7, wherein said step of transferring said data elements includes transferring said data elements through a plurality of service channels each having a different priority associated therewith and receiving a plurality of subgroups of said data elements having corresponding different qualities of service and priorities associated therewith through corresponding ones of said channels.

9. A method according to any one of the preceding claims wherein said at least one service channel of said second layer is taken from the group comprising:

priority token-ring channels, B-ISDN ATM channels, Ethernet high and low priority channels, and synchronous/asynchronous FDDI channels.

10. A method according to any one of the preceding claims, wherein a quality of service associated with each data transmission of a particular group of data elements is identified by identifying a priority, deadline, or period associated with said particular group of dat elements, or by determining burst and throughput quality of service parameters.

11. A method according to any one of the preceding claims, wherein the multilayer network includes at least an N+1 layer and an N layer for assured transmission of data elements, including the steps of:

establishing two or more service channels in said N+1 layer;

establishing one or more service channels in said N layer; and transferring said data elements from said two or more channels in said N+1 layer to said one or more channels in said N layer at said predetermined quality of service.

12. A method according to claim 11, wherein selected groups of said data elements from said two or more channels have a higher and a lower priority, deadline, or period respectively; and

wherein said selected groups having said higher priority are transferred from said N+1 layer through said two or more channels to said one or more channels of said N layer before said selected groups having said lower priority are transferred.

13. A method according to claim 12 including:

preselecting a fixed number of data elements to be received in said one or more channels at said lower priority in said transfer from said N+1 layer to said N layer;

servicing up to said fixed number of data elements in said one or more service channels in said N layer at said lower priority; and

discontinuing said transfer of said data elements at said lower priority when said fixed number of data elements per unit of time is reached.

14. Apparatus for use in transmitting data elements

between computers of a multilayered computer communication network at least at one predetermined assured quality of service, comprising:

means for generating a request at one of said computers for transmission of data elements; and

means responsive to said request including specification of a predetermined quality of service for transmission, for transmitting said data elements according to said quality of service wherein the means for transmitting said data elements includes means for transferring said data elements having said quality of service requirement through at least one service channel of a first layer of said network to at least one service channel of a second lower layer of said network which provides said predetermined quality of service.

15. Apparatus according to claim 14, wherein said multilayered network is an open systems interconnection network.

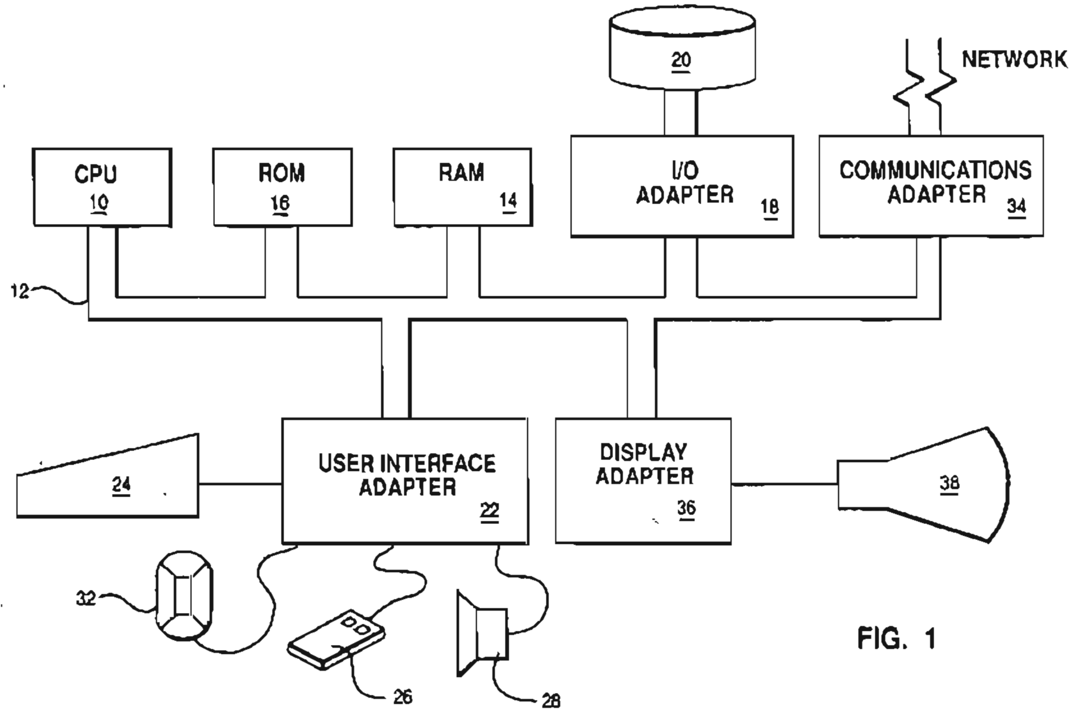

NETWORK
20
CPU
10
ROM
16
RAM
14
I/O
ADAPTER
18
COMMUNICATIONS
ADAPTER
34
12
USER INTERFACE
ADAPTER
22
DISPLAY
ADAPTER
36
24
38
32
26
28

FIG. 1

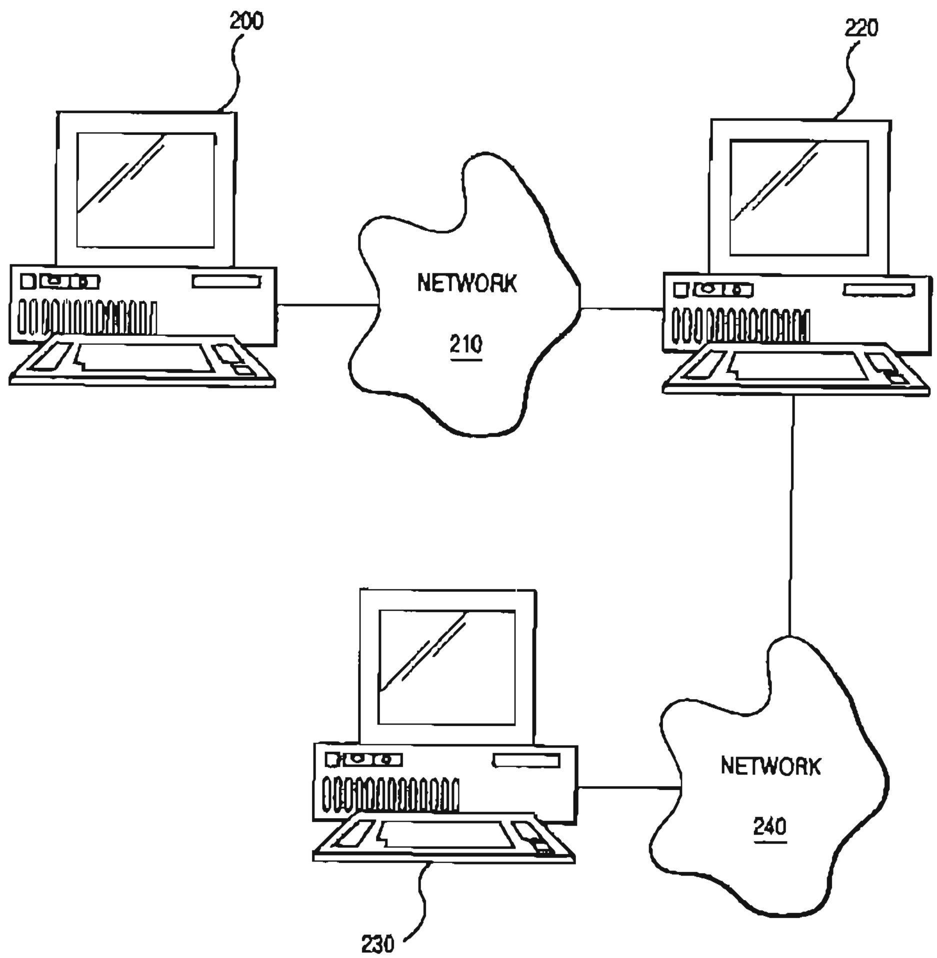


FIG. 2

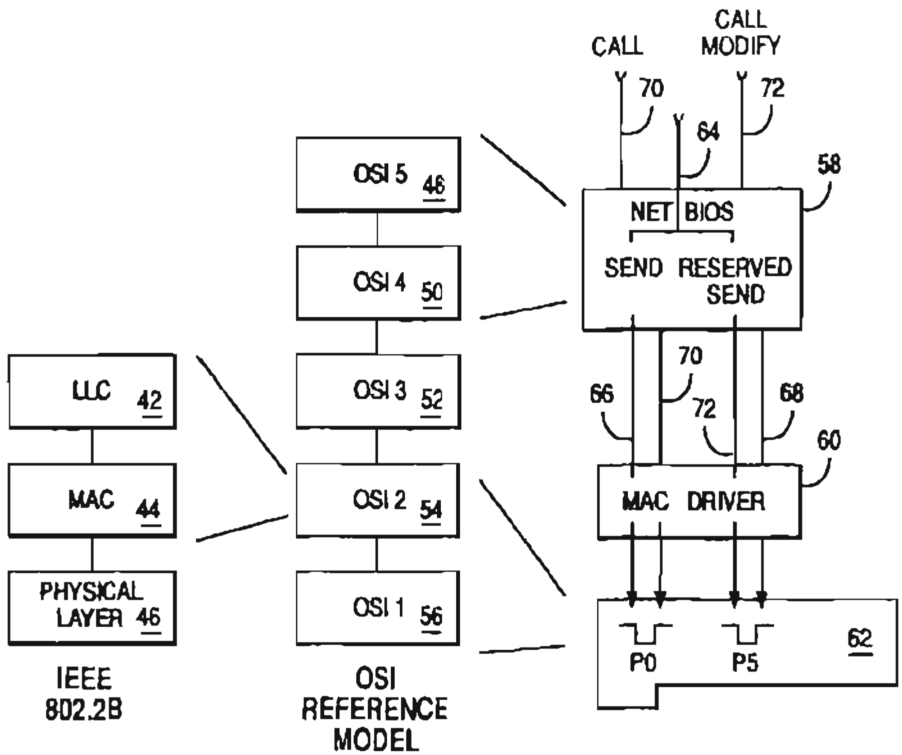
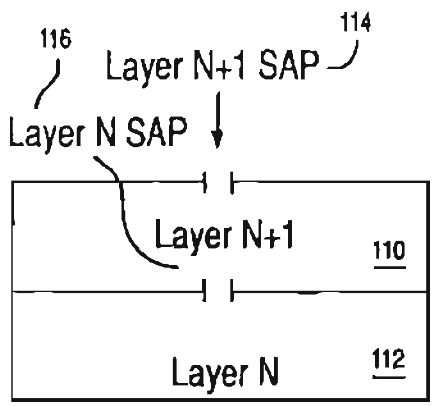
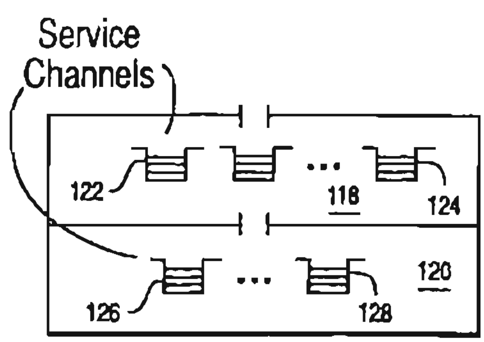


FIG. 3



OSI LAYERED REFERENCE MODEL

FIG. 6



LAYERED SERVICE CHANNEL MODEL

FIG. 7

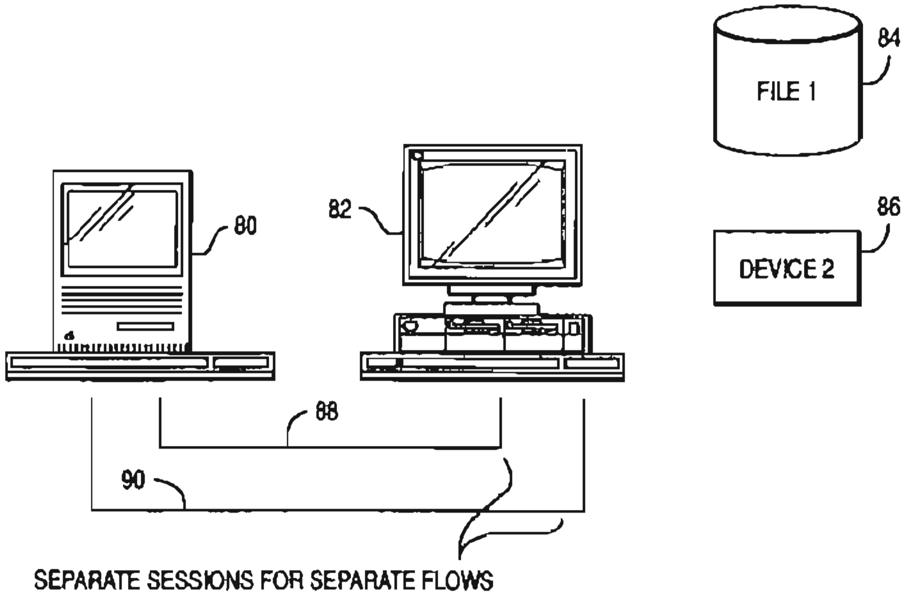

84
FILE 1
82
80
86
DEVICE 2
88
90
SEPARATE SESSIONS FOR SEPARATE FLOWS

FIG. 4

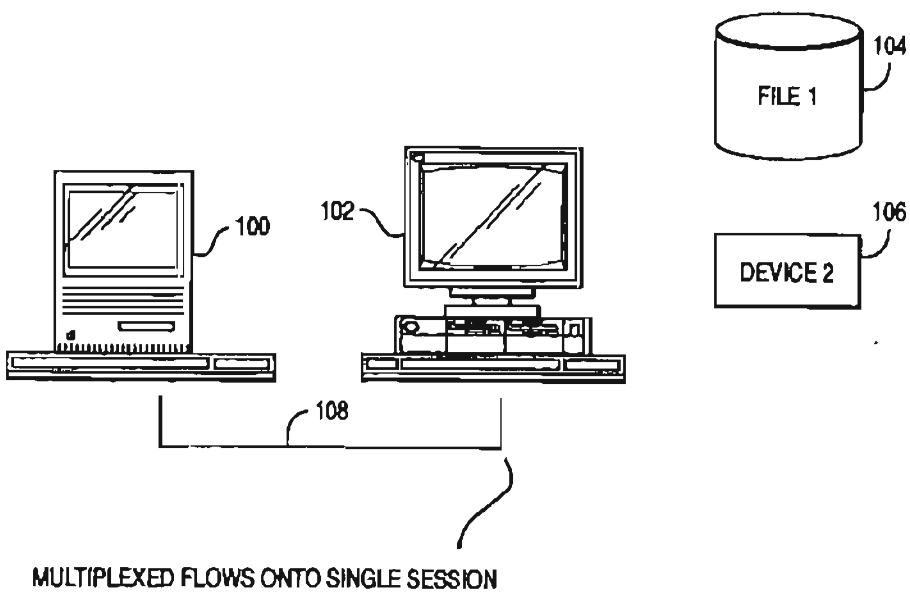

FILE 1
104
102
100
106
DEVICE 2
108
MULTIPLEXED FLOWS ONTO SINGLE SESSION

FIG. 5

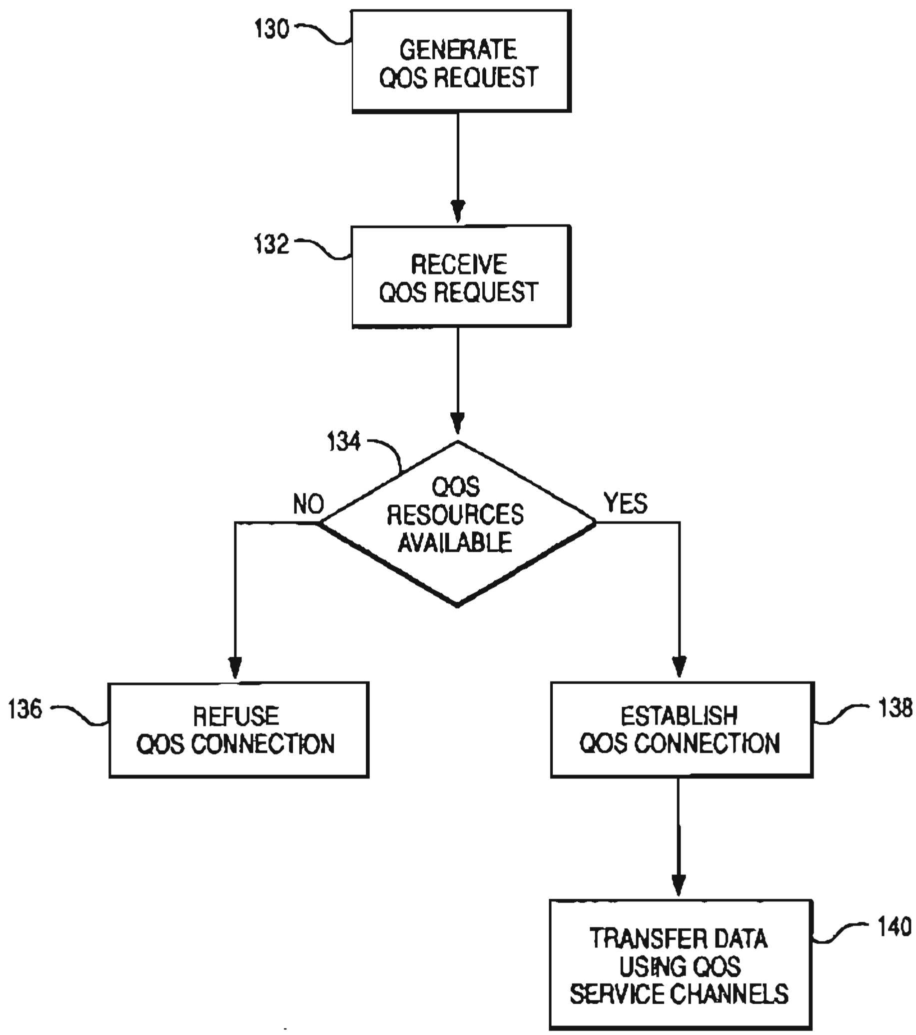

130
GENERATE QOS REQUEST
132
RECEIVE QOS REQUEST
134
QOS RESOURCES AVAILABLE
NO
YES
136
REFUSE QOS CONNECTION
138
ESTABLISH QOS CONNECTION
140
TRANSFER DATA USING QOS SERVICE CHANNELS

FIG. 8

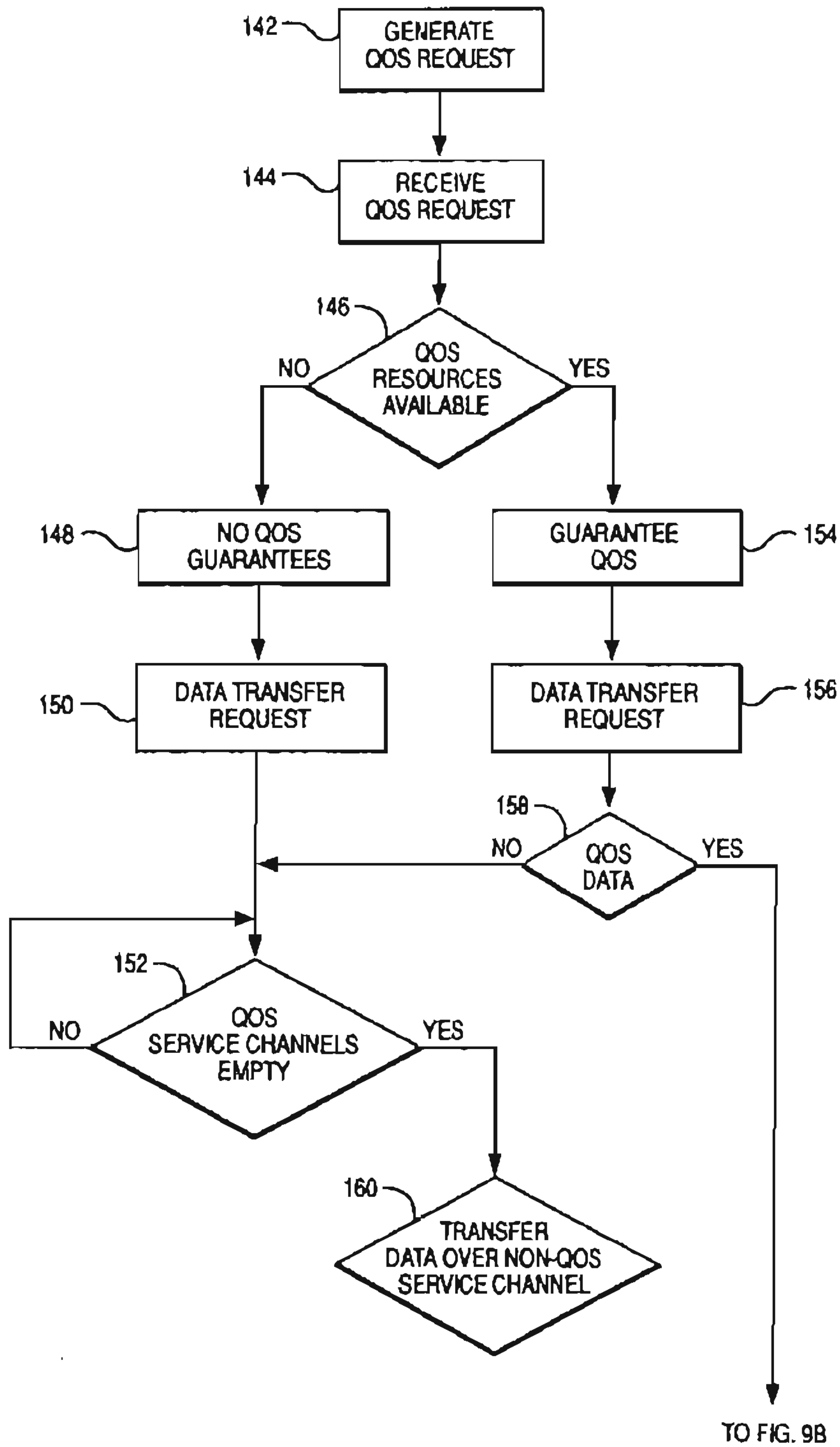

142
GENERATE
QOS REQUEST
144
RECEIVE
QOS REQUEST
146
QOS
RESOURCES
AVAILABLE
NO
YES
148
NO QOS
GUARANTEES
154
GUARANTEE
QOS
150
DATA TRANSFER
REQUEST
156
DATA TRANSFER
REQUEST
158
QOS
DATA
NO
YES
152
QOS
SERVICE CHANNELS
EMPTY
NO
YES
160
TRANSFER
DATA OVER NON-QOS
SERVICE CHANNEL
TO FIG. 9B

FIG. 9A

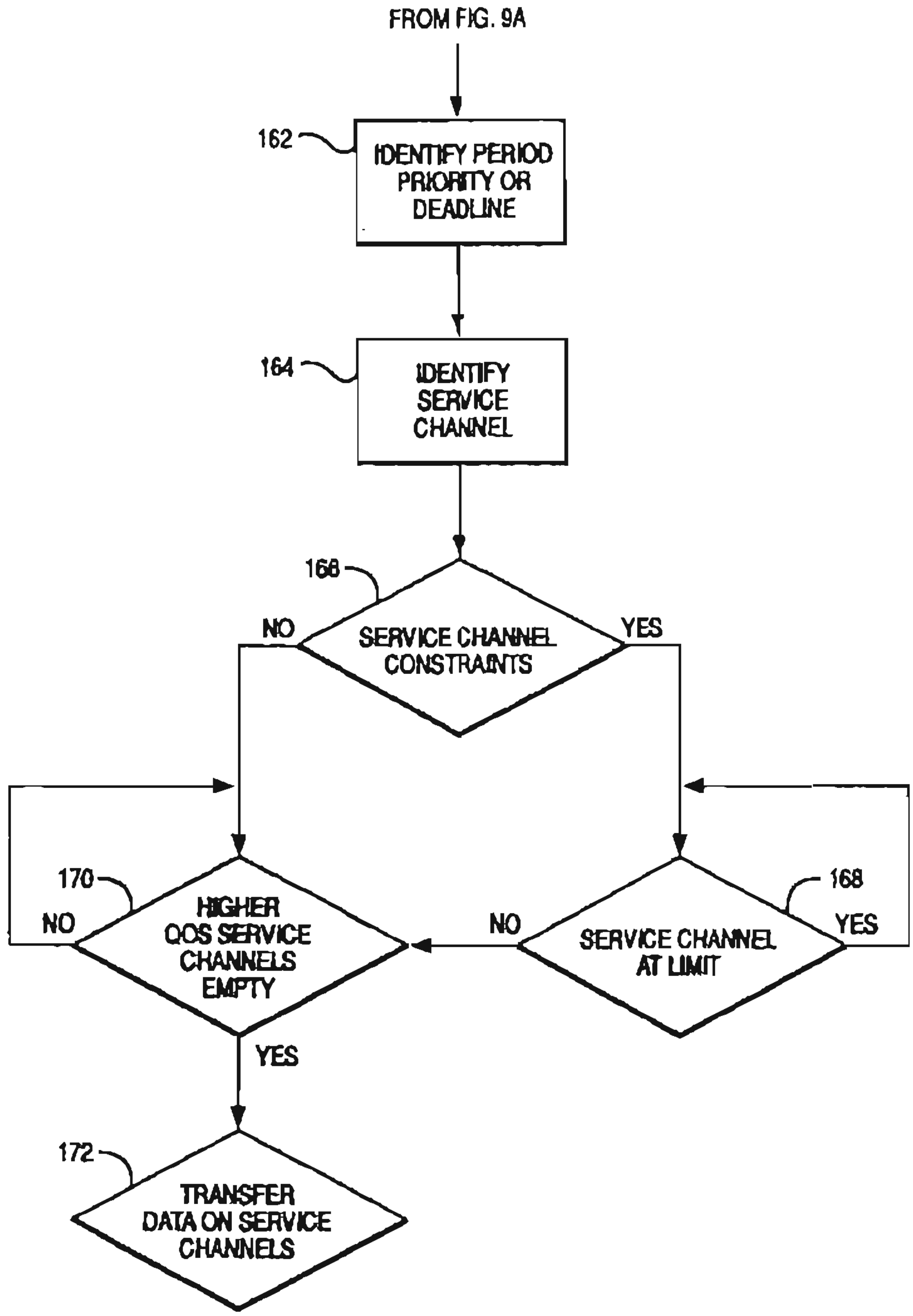

FROM FIG. 9A
162
IDENTIFY PERIOD PRIORITY OR DEADLINE
164
IDENTIFY SERVICE CHANNEL
166
SERVICE CHANNEL CONSTRAINTS
NO
YES
170
HIGHER QOS SERVICE CHANNELS EMPTY
NO
YES
168
SERVICE CHANNEL AT LIMIT
NO
YES
172
TRANSFER DATA ON SERVICE CHANNELS

FIG. 9B

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : 0 648 038 A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number : 94306326.3

(51) Int. Cl.[6] : H04L 29/06

(22) Date of filing : 26.08.94

(30) Priority : 11.09.93 GB 9318877

(43) Date of publication of application : 12.04.95 Bulletin 95/15

(84) Designated Contracting States : DE FR GB

(71) Applicant : International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor : Wallis, Graham Derek
75 Raley Road
Locks Heath, Hampshire, S03 6PB (GB)

Inventor : Taylor, Michael George
St. Elmo,
Duncan Road,
Park Gate
Southampton, Hampshire, S03 7BD (GB)

Inventor : Platt, Michael
Avoca, Chidden Holt,
Hambledon
Hampshire, P07 6TG (GB)

Inventor : Stanford-Clark, Andrew James
68 Allbrook Knoll,
Allbrook
Eastleigh, Hampshire, S03 4RY (GB)

(74) Representative : Burt, Roger James, Dr.
IBM United Kingdom Limited
Intellectual Property Department
Hursley Park
Winchester Hampshire SO21 2JN (GB)

(54) A data processing system for providing user load levelling in a network.

(57) The present invention provides a data processing system, and method of operating such a system, for facilitating a connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources. The data processing system, the client computer, and the server computers all reside in a logical network. The data processing system has an input means for receiving a request from the client computer for a machine address of a server computer identified by a server computer name sent with the request, such a machine address enabling a connection to be made from the client computer to that server computer via the network. A storage device is provided by the system for storing a list identifying server computer names with machine addresses of the server computers. A conversion means in the system uses the list to convert the server computer name received by the input means into the machine address of the server computer, and then an output means sends the machine address from the conversion means to the client computer. The system is characterised by decision logic for studying the server computers at predetermined intervals having regard to a predetermined test criteria, in order to select one of the server computers; and writing means for updating the list by associating the machine address for the server computer selected by the decision logic with a particular server computer name contained as a generic server computer name in the list. Using this technique, when a client computer specifies the generic server computer name, it receives the machine address of the server computer identified by the decision logic.

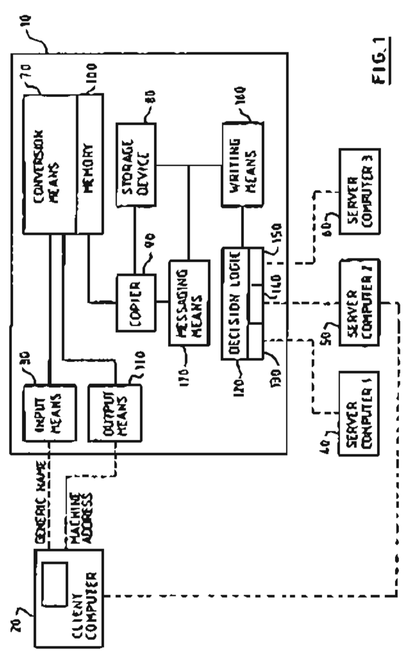


FIG. 1

EP 0 648 038 A2

Jouve, 18, rue Saint-Denis, 75001 PARIS

The present invention relates to a data processing system for facilitating the connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources.

The data processing system, the client computer, and the server computers are all resident on a network. This network need not be one physical network such as a Local Area Network (LAN); for instance it may consist of a number of LANS or WANS (wide area networks) connected together (eg. via 'bridges') to form a single logical network. However the same network protocol will be employed throughout the network, a typical example of such a protocol being TCP/IP which will be familiar to those skilled in the art.

In many environments it is increasingly found that a number of server computers are connected together using some form of network, eg. a LAN. It is often the case that several users of client computers will be connected to one such server computer, whilst other server computers stand idle. An example of this is when such computers are situated in workers' offices, connected by, for example, a token ring LAN using the standard TCP/IP network protocol. When workers are away from their offices, their computers will usually stand idle.

In such situations it is commonly the case that a few of the computers in the network are heavily loaded, whilst other computers in the network are very lightly loaded, giving poor response and performance for the client computers using the heavily loaded server computers. Hence there is a problem of how to enable the client user load to be spread more evenly across the available computing resources of the server in a manner which is transparent to the client computer and its programs. If transparency is to be achieved, standard protocols need to be observed in order that client computers can use a variety of connection methods without modification of any programs being required.

A prior art technique which has been developed to provide some sort of load spreading is called "Static load levelling". With this technique each application on each client computer has a designated server to which it always connects. Hence, for example, if there are 200 potential clients of a server having five server computers, a pre-specified group of, say, 40 of the clients will be told (or configured) to always connect to machine 1, etc. On average it may be argued that this will give a reasonably even load across all of the server computers. However in practice it is often the case that, using this technique, a large number of users of client computers are connected to one server computer, while an adjacent server computer stands completely idle (eg. given the above example there could easily be 40 users on one server computer whilst the other 4 server computers stand idle). Hence in situations where the user loading changes from time to time, the prior art static load levelling technique is not particularly satisfactory. What is required in such instances is a more 'dynamic' technique which can respond to changing user loads, and thus direct new users to the most suitable (eg. least heavily loaded) server computers in the server.

Other prior art techniques can be found in other environments, such as those where job allocation is an issue. For instance in "batch processing", a client computer submits an encapsulated task to a central server, which determines which one of several possible servers is quiet enough to be able to handle the task. The task is sent to that server, is processed, and the results are then sent back to the client (e.g. as a results file, or by electronic mail). With a batch processing system, there is a brief connection to the central server while the job is transferred from the client to the server allocated by the central server. After this brief connection the client disconnects, and has no more interaction with the submitted task until it has been completed, and the results have been passed back to the client by some means.

However in the situation with which we are currently concerned, the dynamic load levelling technique that is required must be able to deal with "interactive" sessions. Rather like a phone conversation, the connection between the program on the client computer and a particular server computer will persist for the duration of the "conversation" session. Hence the batch processing concept is inappropriate in the present situation.

It is possible to write some specific code within a program on a client computer which contains internal message-passing systems to route work from that client program to a corresponding server program. Such systems are dedicated only to that particular client program, and the connection and load-levelling methods are not accessible to other client-server applications. Often, such systems operate by the client connecting to a specific "host" server computer, and from there the work will be sent to another server for processing. Clearly this technique can result in large bottlenecks arising at the "host" server computer.

Hence such a technique is not suitable in the present situation since it only supports one very specific type of client-server connection, whereas we need a technique that will allow any client-server connection method using the network protocol to be connected to a quiet server in a way that is completely transparent to the client program. Further the above technique relies on an initial connection to the 'central' host server computer, which then passes the request on to another server computer; as described above this can potentially create a serious bottleneck.

It is an object of the present invention to provide a technique which facilitates a connection between a client program and a server computer on a server in a way that takes into account the current status of the server computers forming the server. This technique must observe standard network protocols and should operate in a manner which is transparent to the client program requesting access.

Accordingly the present invention provides a data processing system for facilitating a connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources, the data processing system, the client computer, and the server computers residing in a network, the system comprising: input means for receiving a request from the client computer for a machine address of a server computer identified by a server computer name sent with the request, such a machine address enabling a connection to be made from the client computer to that server computer via the network; a storage device for storing a list identifying server computer names with machine addresses of the server computers; conversion means for using the list to convert the server computer name received by the input means into the machine address of the server computer; output means for sending the machine address from the conversion means to the client computer; the system being characterised by: decision logic for studying the server computers at predetermined intervals having regard to predetermined test criteria, in order to select one of the server computers; and writing means for updating the list by associating the machine address for the server computer selected by the decision logic with a particular server computer name contained as a generic server computer name in the list; whereby when a client computer specifies the generic server computer name, it receives the machine address of the server computer identified by the decision logic.

Typically the conversion means will access the list from a local piece of storage, the data processing system having a copier to copy the list from the storage device to that piece of memory. In preferred embodiments the data processing system further comprises a messaging means, responsive to the updating of the list by the writing means, for sending a message to the copier requesting the copier to copy the updated list into the piece of local memory.

Any manner of predetermined test criteria can be used in the data processing system of the invention, for example the amount of idle processor time, the number of processes running, the amount of free memory, the "load average", etc. However in preferred embodiments the predetermined test criteria are such that the decision logic identifies the server computer having the least number of client programs logged on to it.

In preferred embodiments the predetermined intervals are variable and will either be set by a user of the system, eg. the system administrator, or will be adjusted dynamically. The user will also set the predetermined test criteria to be used by the decision logic.

Further in preferred embodiments the user can limit the number of server computers which the decision logic studies. This may be useful if, for instance, some of the server computers have not got access to all of the resources that other server computers have access to, and so would not be suitable as server computers to be associated with the generic server computer name.

In some embodiments it may be advantageous to use a plurality of generic names. Each server name would then have a number of server computers whose machine addresses are associated with that generic name, the decision logic employing different sets of predetermined test criteria for each generic name. In such embodiments one or more of the server computers can be associated with a plurality of the generic names.

Viewed from a second aspect the present invention provides a method of operating a data processing system to facilitate a connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources, the data processing system, the client computer, and the server computers residing in a network, the method comprising the steps of: (a) receiving a request from the client computer for a machine address of a server computer identified by a server computer name sent with the request, such a machine address enabling a connection to be made from the client computer to that server computer via the network; (b) storing a list identifying server computer names with machine addresses of the server computers in a storage device; (c) converting, with reference to the list, the server computer name received at step (a) into the machine address of the server computer; (d) sending the machine address identified at step (c) to the client computer; the method being characterised by the steps of: (e) employing decision logic to study the server computers at predetermined intervals having regard to predetermined test criteria, in order to select one of the server computers; and (f) updating the list by associating the machine address for the server computer selected by the decision logic with a particular server computer name contained as a generic server computer name in the list; whereby when a client computer specifies the generic server computer name at step (a), it receives the machine address of the server computer identified by the decision logic.

The present invention will be described further, by way of example only, with reference to an embodiment therof as illustrated in the accompanying drawings, in which:

Figure 1 is a block diagram illustrating the data processing system of the preferred embodiment;

Figure 2 is a flow diagram illustrating how the decision logic in the data processing system of the preferred embodiment operates; and

Figure 3 illustrates a particular embodiment where two generic computer names are used.

In the preferred embodiment we will consider the situation where the server in question is a high performance database server which has its data distributed across a network of server computers, this server network hereafter being referred to as a cluster. Database applications being run by users on client computers are required to connect to one of the server computers in the cluster to enable them to access the data in the database server. By the nature of the database system, it does not matter which server computer the client connects to - all of the data is accessible from any server computer in the cluster. In the preferred embodiment the server computers and the client computers are all interconnected using TCP/IP on a token ring Local Area Network.

For a large number of users, it is highly desirable to have a number of users on each of the server computers in the server cluster, rather than all users connecting to (and hence overloading) just one or a few of the server computers. With a widely varying user workload profile for the database server, this problem can only be solved by providing some form of "load levelling" process, which will allocate new client application instances to server computers in the cluster that are most suitable for the client connection (eg. because they are more lightly loaded than other server computers). Clearly this process must be dynamic, able to respond to changing load conditions over time. Since the database applications on the client computers are typically complex and often are supplied only in object code form, it would be very difficult (or impossible) for the system administrator to alter them, and so it is essential that this allocation is done in a manner which is entirely transparent to the client application.

The manner in which the data processing system of the preferred embodiment solves the above problems will now be described with reference to Figure 1.

Each client computer in a network using the TCP/IP protocol (there will typically be many such client computers) will have been informed by the network administrator that it is to communicate with a particular computer when it wishes to convert a computer name of another computer in the network into a machine address. When utilising the present invention that computer will be the data processing system of the preferred embodiment.

Hence when a program running on a client computer 20 (for clarity, only one client computer is illustrated) wishes to obtain access to a server computer (40, 50, 60) in the cluster it will communicate with the data processing system 10 in order to obtain a full Internet machine address for the desired server (Internet addressing is part of the TCP/IP protocol). With the prior art technique the client computer would specify a server computer name in this communication that was specific to one particular server computer in the cluster. The input means 30 of the data processing system 10 would receive this server computer name and pass it to the conversion means 70.

In a storage device 80 of the data processing system a list is maintained which identifies server computer names with particular Internet addresses. When the conversion means is initiated the copier 90 copies this list from the storage device 80 into a piece of local memory 100 accessible by the conversion means 70. Hence the conversion means will access the list in memory 100 to find the Internet address of the computer associated with the server computer name passed to it by the input means 30. This Internet address will then be provided by the conversion means to the output means 110 for transmission back to the client computer 20.

Once the client computer has the Internet address it can then make direct contact with the server computer residing at the Internet address provided; in Figure 1 this is server computer 2. Since the TCP/IP protocol is used any of the access methods that use this protocol can be used to access the server computer.

When using the data processing system of the preferred embodiment the program running on the client computer 20 will not use the server computer name that it previously used. Instead a generic server computer name will be used. This generic server name will either have been placed in the program's configuration file, or alternatively the user of the program will specify the generic name when running the program.

Within the data processing system, decision logic 120 is provided which periodically studies the server computers in the cluster having regard to some predetermined test criteria, hereafter called the metric string. In the preferred embodiment the metric string is a list of questions which when answered by the various server computers will enable the decision logic to decide which server computer is most suitable for a client connection (the most suitable perhaps being the least heavily loaded server computer). The metric string can be altered as the system administrator deems appropriate, depending on what criteria the administrator wishes to be used to select a server computer.

In the preferred embodiment the decision logic actually sets up a number of child processes, each one being responsible for sending the metric string to a particular server computer and receiving the response from the server computer.

Once the responses have been received the decision logic will collate the responses, decide which server computer is most suitable, and then request the writing means 160 to pass the Internet address of that server computer to the storage device for association with the generic server computer name. If however the most suitable server computer is the same server computer as that identified in the last iteration of the process then there is no need to update the storage device and the writing means will not be activated.

Once any necessary update to the storage device 80 has been made the messaging means 170 will notify the copier 90 so that the copier updates the local memory 100 with the new list as stored in the storage device 80.

Hence when the client computer requests a machine address for the generic server computer name the conversion means 70 accesses the list in memory 100 and identifies a machine address just as it would if any other server computer name had been given. However in this instance the machine address actually relates to the server computer in the cluster which has been identified by the decision logic as the most suitable (eg. least heavily loaded). When this machine address is passed back to the client computer 20 via the output means 110, the client computer will proceed to automatically access the server which is most suitable.

By this approach it will be seen that a dynamic load levelling facility is provided which is completely transparent to the client program. As far as the program is concerned it is requesting a machine address as normal and is using one of the normal TCP/IP access methods to gain access to the server computer allocated to it.

In many of todays computing environments (eg Unix, AIX (Unix is a Trade Mark of Unix Systems Laboratories Inc)) an application is provided to perform the standard name resolution service (ie receipt of a computer name and conversion of that computer name into a full Internet address). This application is commonly known as a "nameserver" application, and is installed on one or more computers in the logical network. Every other computer in the network is told to communicate with a specified one of these 'nameserver' computers when it wishes to determine an Internet address for any other computer in the network. Hence a nameserver computer provides a resolution service to client computers by receiving from them a convenient name given to a particular computer (eg. abc.def.ghi.com), and converting it into a full Internet address (eg. 29.1.19.66). This Internet address is then used by the routing subsystem (TCP/IP) to allow a client user or application access to the physical computer (eg "abc" in this example).

In the above example of a computer name, "abc" is the physical machine, "def" is typically the site location, "ghi" the organisation, and "com" one of the Internet classes (three such classes are (com)mercial, (edu)cation, (mil)itary). Domains and sub-domains can also be added as part of this computer name. Basically the name takes a hierarchical form, with the finest resolution at the beginning and the coarsest resolution at the end; this type of naming structure will of course be well known to those skilled in the art.

All TCP/IP-based applications, including remote-login, remote-shell, telnet, ftp, and also client-server applications (such as database applications), are aware of the nameserver facility, and will automatically go to the designated nameserver computer to ask for resolution of a computer name into an Internet address before attempting to make a connection to another computer in the network.

If we consider Figure 1 again, the standard nameserver facility will include the following elements: the input means 30, the conversion means 70 with associated memory 100, the output means 110, the list stored in the storage device 80, and the copier 90.

The nameserver application is a "daemon" (background) process which runs on the data processing system; this data processing system may (but need not) be one of the server computers forming part of the cluster over which users are to be distributed. In Unix-type operating systems (eg. AIX by IBM Corporation, Ultrix by Digital Equipment Corporation, OSF/1 by the Open Software Foundation, and HP-UX by Hewlett Packard, etc) this daemon process is called "named" (name-daemon), and when it is initialised, it reads a special database file (named.data) stored on the storage device 80 to obtain details of the computer names about which it is expected to know (over which it has "authority"), and the corresponding Internet addresses ("dotted decimal", e.g. 29.1.19.66) for each computer name. Whilst the name daemon is operating, it can be forced to re-read the information from the named.data database file by the sending of an inter-process signal to the name daemon process telling it to update its internal tables 100 from the database file (named.data).

In the preferred embodiment of the present invention we provide a further facility which runs on the same computer as the nameserver application ("named"), and interfaces with it. A 'generic' computer name is introduced into the database file (named.data), which refers not to one specific computer, but to any one of a number of computers offering equivalent functionality. For example, the generic name might be "server.cluster.def.ghi.com"; a client program requesting a connection to 'server.cluster' is requesting connection to any one of the computers in the server cluster.

The further facility provided by the preferred embodiment will be referred to hereafter as the "User Load Leveller" (ULL) application. This application is responsible for deciding which server computer in the cluster is currently the least heavily loaded, according to some appropriate metric, and for conveying this information to the nameserver application. Then subsequent requests for resolution of the generic server computer name to an Internet address result in the nameserver application sending back to the client computer the Internet address of the server computer which has been deemed to be the most appropriate server computer for connection at that point in time.

The ULL application consists of the following elements from figure 1: the decision logic 120 with child processes 130, 140, 150; the writing means 160; and the messaging means 170. As described earlier with reference to Figure 1 the ULL application periodically (at a frequency which can varied (eg. tuned by a system administrator or dynamically adjusted)) polls the server computers in the cluster to determine how "busy" in some sense they are. The metric used may vary, depending on the type of work which is being handled by the cluster, but may for example include the number of login sessions, number of application instances running, number of idle cpu cycles since the last poll, etc. The metric can be altered to ensure that it is appropriate to a specific situation.

Based on the results of this polling, and taking into account the situation where a server computer in the cluster is too busy to respond to the status request within a certain number of seconds, the ULL application decides which machine is currently the least heavily loaded. The ULL application then modifies the database file (named.data) to associate the generic cluster machine name with the Internet address of this least heavily loaded machine, and sends the special inter-process signal via the messaging means 170 which tells the nameserver application to re-read its database file. The nameserver application will then, in response to a name resolution request from a client program, resolve the generic server computer name into the Internet address of the most appropriate server computer in the cluster for the client program to connect to.

The process carried out by the decision logic 120 of the preferred embodiment will now be described in more detail with reference to Figure 2. At step 200 the ULL application is initialised. A number of steps are carried out at initialisation. For example the application: checks for multiple copies of the ULL application in memory; cleans up from a previous run of the application (by freeing up system resources such as memory, locks and semaphores still held in the name of the previous instance of the ULL application); and locates the nameserver application (named) and its data file (named.data). The ULL application then parses its configuration file to read information defined by the system administrator, such as the metric strings, poll periods, identities of server computers in the cluster, etc. Further the ULL application generates a number of "child" processes - one per server computer - which are each responsible for polling the activity of one designated server.

Once the initialisation has been completed the process enters a main loop which executes until the ULL application is terminated. At steps 210, 220, 230 and 240 the child processes send a metric string (as defined by the system administrator) to each server computer in the cluster, await responses from those computers, and then wait for a trigger signal from the main ULL application.

Once the trigger signal has been sent by the main application the responses are sent by the child processes to the main application. The main application then collates the activity results received from the child processes (step 250), and based on predetermined test criteria identifies the most appropriate server computer (the "least busy" server computer) at step 260. At step 270 it is determined whether the server computer identified at step 260 differs from the current nominated server computer. If it does then the process advances to step 290, at which point the nameserver's data file (named.data) is modified. Further at step 300 a notification signal is sent to the nameserver application (named) to tell it to update its internal information from the data file.

The process then proceeds to step 280. If at step 270 it is determined that the server computer identified at step 260 is the same as the current nominated server computer then the process moves straight to step 280 without steps 290 and 300 being carried out. Writing to and reading from the data file are time consuming activities and so steps 290 and 300 should only be performed when necessary (ie when the "least busy" server computer changes).

At step 280 the process waits until the end of the "poll period". This period is the predetermined interval (as defined by the system administrator) between successive studies of the server computers by the ULL application. Once the poll period has expired the process loops back to steps 210-240 and the main loop is repeated.

Having discussed the preferred embodiment a few possible alterations will now be discussed. Firstly more than one generic server computer name can be added to the list in storage device 80 (the named.data file). Each generic name could be associated with a particular group of server computers, these groups being either completely separate or having a few server computers common to a plurality of the groups. Indeed one group may be a subset of another group. As an example consider Figure 3. A server cluster 410 comprises eight server computers 400. All eight server computers have access to a main body of data, but only four of them have access to some further (possibly more confidential) data.

In this situation two generic names could be generated, eg. "general.cluster" and "specific.cluster". Any one of the eight computers (enclosed by ring 430) can be associated with the former generic name, but only the four enclosed by ring 420 can be associated with the latter generic name, since only those four have access to the further (confidential) data.

The system administrator can then set up the metric string to be used when studying all eight server computers, or when studying only the four in ring 420; the metric string could be the same in both instances but need not be. If a client application needs access to the confidential information then it would request access to "specific.cluster", but if an application only needed access to the general information then it would request access to "general.cluster".

By this approach an application which only needs access to the general information will always be connected to the least busy server computer, whilst an application which needs access to the further (confidential) information will be given the machine address of the least busy server computer that can actually provide the necessary service; this may or may not be the least busy server computer in the network.

In preferred embodiments a further feature is provided to enable the decision logic to temporarily implement a "round-robin" metric instead of the above described 'studying' process. The round-robin principle will be familiar to those skilled in the art; basically when a client application requests access to a server computer it is assigned a particular server computer, and when the next request is received then that application is assigned the next server computer in the cluster, and so on. In this way the server computers are rotated so that each successive server access is made on a different server computer to the previous server access. Alternatively the server computers can be rotated at fixed time intervals rather than after each access request.

Although the round-robin technique does not have regard to the loading on any of the server computers, and so there is no determination of the least busy server computer, there are certain situations (eg. where there are lots of client applications which only take a short amount of database connection time) where a round-robin approach is acceptable. To implement the round robin approach the decision logic 120 would ask the writing means 160 to update the storage device 80 after each access request has been handled (or at fixed time intervals if the alternative approach is used), so that the generic name is always associated with successive server computers in the cluster in turn.

From the above description it will be clear that the system of the preferred embodiment has a number of advantages. Firstly the technique dynamically allocates new client users and applications to the server computer which is least heavily loaded at the time they make the connection, thus ensuring an even distribution of users and applications across all of the available server computers. The client computer only briefly contacts the data processing system of the preferred embodiment to resolve the generic computer name into a machine address. Completely standard access methods (eg. as provided by TCP/IP) are then used to make the connection, thus avoiding any proprietary protocols or any need to modify access methods or applications, and so providing fully transparent user load levelling.

Secondly the technique of the preferred embodiment does not involve any modification to the nameserver code - the User Load Leveller application interfaces with the standard code (eg. "named" as shipped with the unix/AIX operating system). It would be possible to provide similar functionality to that described here by producing a modified version of the nameserver code. However, avoiding this brings major advantages from both marketing and maintenance points of view.

Further the technique can be operated without requiring any modification to the server computers. They are accessed in a standard way after the generic server computer name has been used to provide the client computer with a machine address.

Another advantage is that the key parameters, such as the time interval between polls of the server computers in the cluster and the metric used to determine which server computer is least heavily loaded, can be altered and tuned by a local system administrator, allowing the system to be optimised for a particular situation.

The above described ULL application could be supplied as a separate tool to enhance the useability of parallel and distributed systems, or could be shipped with the nameserver application.

Claims

1. A data processing system for facilitating a connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources, the data processing system, the client computer, and the server computers residing in a network, the system comprising:
   input means for receiving a request from the client computer for a machine address of a server computer identified by a server computer name sent with the request, such a machine address enabling a connection to be made from the client computer to that server computer via the network;
   a storage device for storing a list identifying server computer names with machine addresses of the server computers;
   conversion means for using the list to convert the server computer name received by the input means into the machine address of the server computer;
   output means for sending the machine address from the conversion means to the client computer;

the system being characterised by:
decision logic for studying the server computers at predetermined intervals having regard to predetermined test criteria, in order to select one of the server computers; and
writing means for updating the list by associating the machine address for the server computer selected by the decision logic with a particular server computer name contained as a generic server computer name in the list;
whereby when a client computer specifies the generic server computer name, it receives the machine address of the server computer identified by the decision logic.

2. A system as claimed in Claim 1 further comprising:

a copier to copy the list from the storage device to a piece of memory accessible by the conversion means;

a messaging means, responsive to the updating of the list by the writing means, for sending a message to the copier requesting the copier to copy the updated list into the piece of local memory.

3. A system as claimed in Claim 1 or Claim 2, wherein the predetermined test criteria are such that the decision logic identifies the server computer having the least number of client programs logged on to it.

4. A system as claimed in any of claims 1 to 3, wherein the predetermined intervals are variable.

5. A system as claimed in any preceding claim, wherein the predetermined test criteria are set by a user of the system.

6. A system as claimed in any preceding claim, wherein the user can limit the number of server computers which the decision logic studies.

7. A system as claimed in any preceding claim wherein a plurality of generic names are used, each one having a number of server computers whose machine addresses are associated with that generic name, the decision logic employing different sets of predetermined test criteria for each generic name.

8. A system as claimed in Claim 7, wherein one or more of the server computers are associated with a plurality of the generic names.

9. A method of operating a data processing system to facilitate a connection of a program on a client computer to a server, the server consisting of a plurality of server computers with shared resources, the data processing system, the client computer, and the server computers residing in a network, the method comprising the steps of:

(a) receiving a request from the client computer for a machine address of a server computer identified by a server computer name sent with the request, such a machine address enabling a connection to be made from the client computer to that server computer via the network;

(b) storing a list identifying server computer names with machine addresses of the server computers in a storage device;

(c) converting, with reference to the list, the server computer name received at step (a) into the machine address of the server computer;

(d) sending the machine address identified at step (c) to the client computer;

the method being characterised by the steps of:

(e) employing decision logic to study the server computers at predetermined intervals having regard to predetermined test criteria, in order to select one of the server computers; and

(f) updating the list by associating the machine address for the server computer selected by the decision logic with a particular server computer name contained as a generic server computer name in the list;

whereby when a client computer specifies the generic server computer name at step (a), it receives the machine address of the server computer identified by the decision logic.

10. A method as claimed in Claim 9 further comprising the steps of:

copying the list from the storage device to a piece of memory accessible at the conversion step (c);

repeating, in response to the updating of the list at step (f), the copying step to ensure that the updated list is copied into the piece of local memory.

11. A method as as claimed in claim 9 or claim 10, wherein the predetermined intervals are set by a user of the system.

12. A method as claimed in any of claims 9 to 11, wherein the predetermined test criteria are set by a user of the system.

13. A method as claimed in any of claims 9 to 12, wherein the user can limit the number of server computers which the decision logic studies.

14. A method as claimed in any of claims 9 to 13, wherein a plurality of generic names are used, each one having a number of server computers whose machine addresses are associated with that generic name, the decision logic employing different sets of predetermined test criteria for each generic name.

15. A method as claimed in Claim 14, wherein one or more of the server computers are associated with a plurality of the generic names.

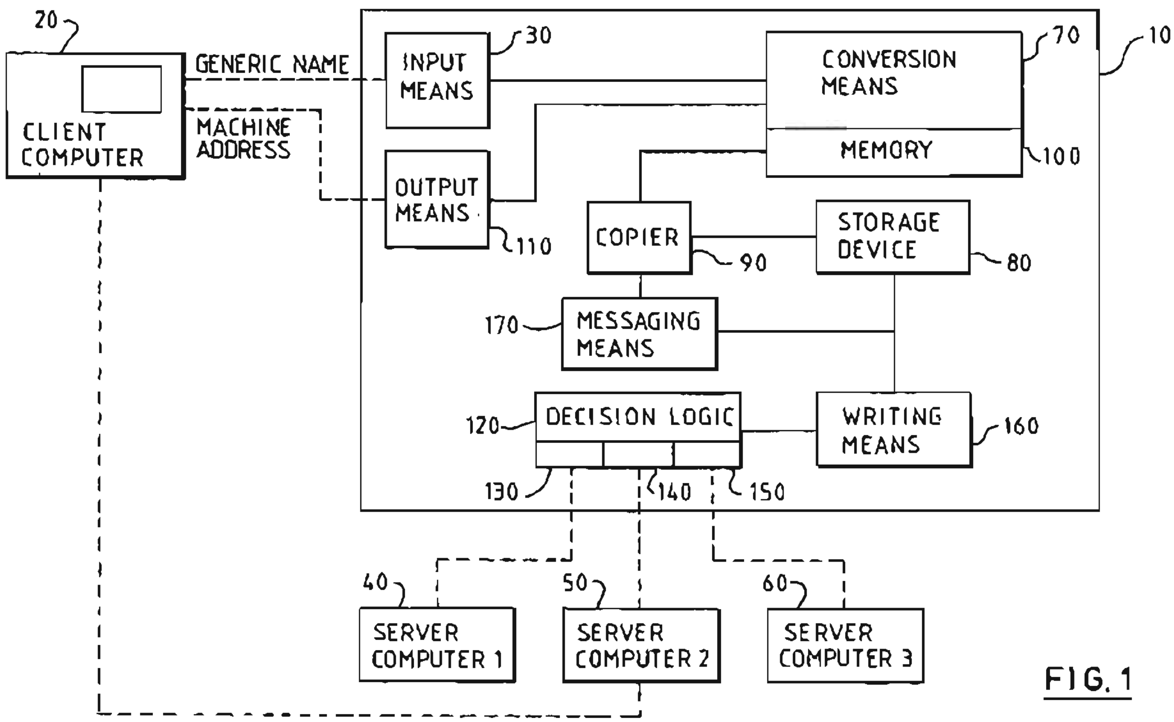

20
CLIENT COMPUTER
GENERIC NAME
MACHINE ADDRESS
INPUT MEANS
30
OUTPUT MEANS
110
CONVERSION MEANS
70
MEMORY
100
10
COPIER
90
STORAGE DEVICE
80
170
MESSAGING MEANS
120
DECISION LOGIC
WRITING MEANS
160
130
140
150
40
SERVER COMPUTER 1
50
SERVER COMPUTER 2
60
SERVER COMPUTER 3

FIG. 1

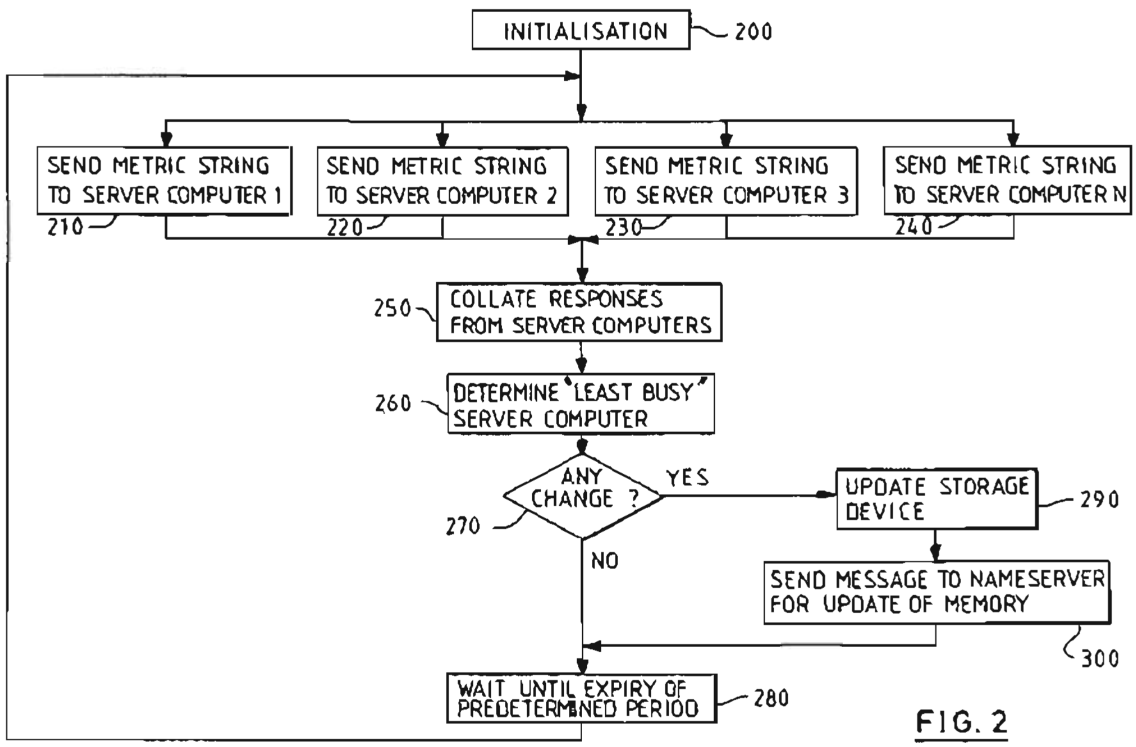

INITIALISATION
200
SEND METRIC STRING TO SERVER COMPUTER 1
210
SEND METRIC STRING TO SERVER COMPUTER 2
220
SEND METRIC STRING TO SERVER COMPUTER 3
230
SEND METRIC STRING TO SERVER COMPUTER N
240
COLLATE RESPONSES FROM SERVER COMPUTERS
250
DETERMINE "LEAST BUSY" SERVER COMPUTER
260
ANY CHANGE ?
270
YES
NO
UPDATE STORAGE DEVICE
290
SEND MESSAGE TO NAMESERVER FOR UPDATE OF MEMORY
300
WAIT UNTIL EXPIRY OF PREDETERMINED PERIOD
280

FIG. 2

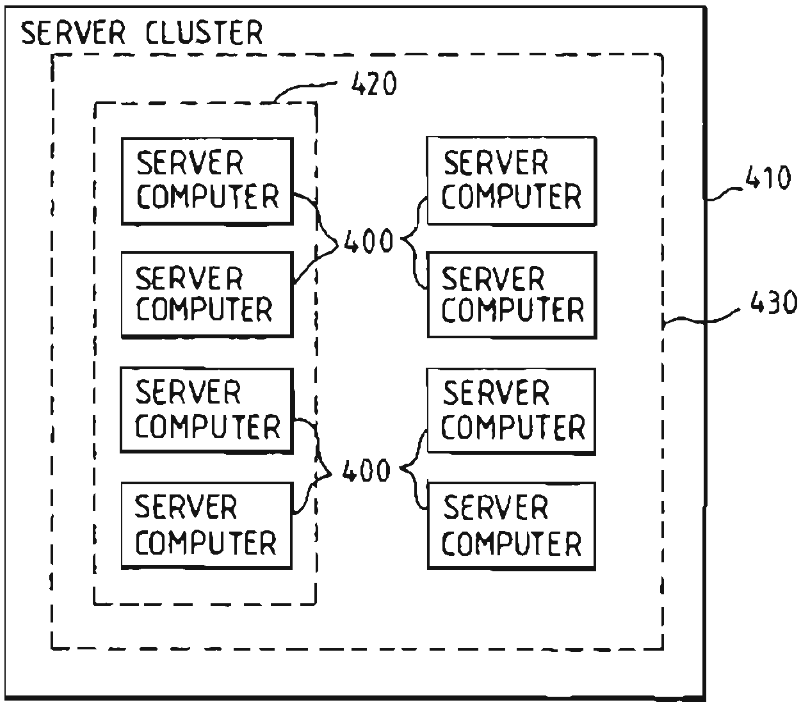

SERVER CLUSTER
420
SERVER COMPUTER
SERVER COMPUTER
SERVER COMPUTER
SERVER COMPUTER
400
400
SERVER COMPUTER
SERVER COMPUTER
SERVER COMPUTER
SERVER COMPUTER
410
430

FIG. 3

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 379 039 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
07.01.2004 Bulletin 2004/02

(51) Int Cl.7: H04L 12/58, H04L 29/06

(21) Application number: 03022287.1

(22) Date of filing: 25.09.1996

(84) Designated Contracting States:
CH DE ES FI FR GB IE IT LI NL SE

(30) Priority: 25.09.1995 US 533115

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
96933928.2 / 0 852 868

(71) Applicant: Netspeak Corporation
Boca Raton, FL 33487 (US)

(72) Inventor: Hutton, Glenn W.
Miami, FL 33196 (US)

(74) Representative: Kindermann, Manfred
Patentanwalt,
Sperberweg 29
71032 Böblingen (DE)

Remarks:
This application was filed on 02 - 10 - 2003 as a divisional application to the application mentioned under INID code 62.

(54) Point-to-point communication using e-mail to establish dynamic network addresses

(57) In a computer system having an audio transducer and a display device and being operatively coupled to other computers over a computer network (24), such as the Internet, means are included for establishing a point-to-point communication link between processes. The means provide for transmitting from a first processing unit (12) to the Internet an E-mail signal, including a first IP address assigned to the first processing unit, and for processing the E-mail signal through the Internet to deliver the E-mail signal to a second processing unit (22). Further means are provided for transmitting a second IP address to the first processing unit for establishing a point-to-point communication link between the first and second processing units through the Internet.

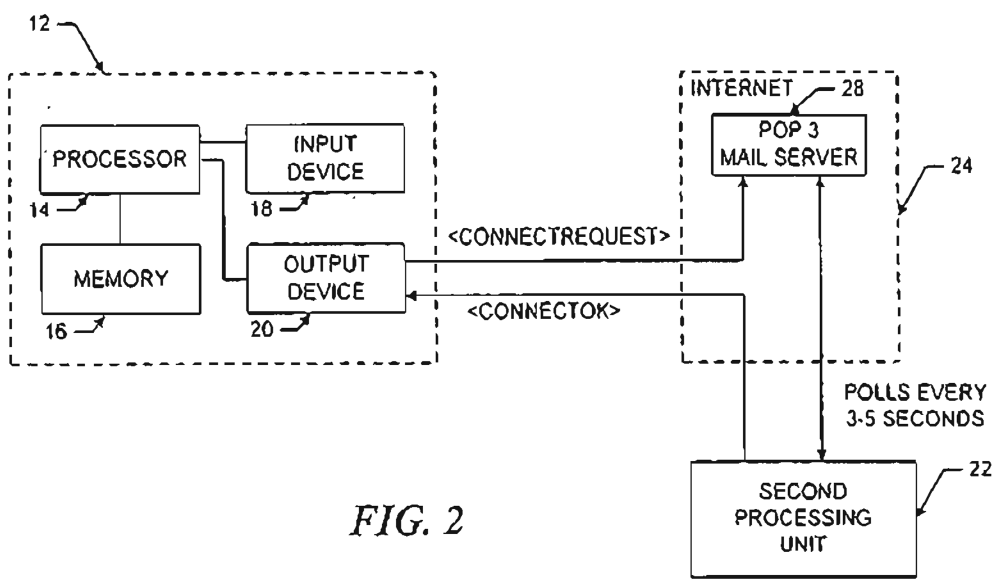


FIG. 2

EP 1 379 039 A2

Printed by Jouve, 75001 PARIS (FR)

Description

FIELD OF THE INVENTION

[0001] The present invention relates in general to data processing systems, and more specifically, to an apparatus, a method and a computer program product for facilitating audio communications over computer networks.

BACKGROUND OF THE INVENTION

[0002] The increased popularity of on-line services such as AMERICA ONLINE (TM), COMPUSERVE (R), and other services such as Internet gateways have spurred applications to provide multimedia contents, including video and voice clips, to online users. An example of an online voice clip application is VOICE E-MAIL FOR WINCIM and VOICE E-MAIL FOR AMERICA ONLINE (TM), available from Bonzi Software, as described in "Simple Utilities Send Voice E-Mail Online", MULTIMEDIA WORLD, VOL. 2, NO. 9, August 1995, p. 52. Using such Voice E-Mail software, a user may create an audio message to be sent to a predetermined E-mail address specified by the user.

[0003] Generally, devices interfacing with the Internet and other online services may communicate with each other upon establishing respective device addresses. One type of device address is the Internet Protocol (IP) address, which acts as a pointer to the device associated with the IP address. A typical device may have a Serial Line Internet Protocol or Point-to-Point Protocol (SLIP/PPP) account with a permanent IP address for receiving E-mail, voicemail, and the like over the Internet. E-mail and voicemail is generally intended to convey text, audio, etc., with any routing information such as an IP address and routing headers generally being considered an artifact of the communication, or even gibberish to the recipient.

[0004] Devices such as a host computer or server of a company may include multiple modems for connection of users to the Internet, with a temporary IP address allocated to each user. For example, the host computer may have a general IP address "XXX.XXX.XXX", and each user may be allocated a successive IP address of XXX.XXX.XXX.10, XXX.XXX.XXX.11, XXX.XXX.XXX.12, etc. Such temporary IP addresses may be reassigned or recycled to the users, for example, as each user is successively connected to an outside party. For example, a host computer of a company may support a maximum of 254 IP addresses which are pooled and shared between devices connected to the host computer.

[0005] Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet. For example, global real-time video conferencing has been implemented using dedicated IP addresses and mechanisms known as reflectors.

[0006] A technique for matching domain names to Internet Protocol addresses is described in the text entitled "Internetworking With TCP/IP", 2nd Edition, by Douglas E. Comer, November 1992, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A. Comer describes a domain name system and cooperative systems of name servers for matching domain names to network addresses. Each name server is a server program that supplies mapping of domain names to IP addresses. The system described in Comer, however, is not designed for use with network nodes whose network names or name to address bindings change frequently.

[0007] International Publication WO 92/19054 discloses a network monitoring system including an address tracking module which uses passive monitoring of all packet communications over a local area network to maintain a name table of IP address mappings. The disclosed address tracking module is capable of monitoring only a small number of nodes on a local area network and is not suitable for use with a multitude of nodes over a wide area network.

[0008] Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in real-time of voice and video have been generally difficult to attain.

SUMMARY OF THE INVENTION

[0009] In a system for enabling point-to-point communications between a plurality of processing units over the Internet, means are provided for establishing a point-to-point communication link between a first processing unit and a second processing unit. The invention, as defined in the claims, comprises (a) means for transmitting from a first processing unit to the Internet an E-mail signal, including a first IP address assigned to the first processing unit, (b) means for processing the E-mail signal through the Internet to deliver the E-mail signal to a second processing unit and (c) means for transmitting a second IP address to the first processing unit for establishing a point-to-point communication link between the first and second processing units through the Internet.

[0010] The invention is in particular suitable for being used in connection with computer networks, such as the Internet, wherein the processing unit does not have a fixed or predetermined network protocol address. The invention thus provides for a protocol wherein a calling processing unit transmits by E-mail its dynamically assigned network protocol address, or IP address, directly to the called processing unit. The called processing unit then transmits its dynamically assigned IP address to the calling processing unit also via E-mail message.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The features of the invention will become more readily apparent and may be better understood by referring to the following detailed description of an illustrative embodiment of the present invention, taken in conjunction with the accompanying drawings, in which:

FIG. 1 illustrates, in block diagram format, a system for the disclosed point-to-point Internet protocol;
FIG. 2 illustrates, in block diagram format, the system using a secondary point-to-point Internet protocol;
FIG. 3 illustrates, in block diagram format, the system of FIGS. 1-2 with the point-to-point Internet protocol established;
FIG. 4 is another block diagram of the system of FIGS 1-2 with audio communications being conducted;
FIG. 5 illustrates a display screen for a processing unit;
FIG. 6 illustrates another display screen for a processing unit;
FIG. 7 illustrates a flowchart of the initiation of the point-to-point Internet protocols;
FIG. 8 illustrates a flowchart of the performance of the primary point-to-point Internet protocols; and
FIG. 9 illustrates a flowchart of the performance of the secondary point-to-point Internet protocol.

DETAILED DESCRIPTION

[0012] Referring now in specific detail to the drawings, with like reference numerals identifying similar or identical elements, as shown in FIG. 1, the present disclosure describes a point-to-point network protocol and system 10 for using such a protocol.

[0013] In an exemplary embodiment, the system 10 includes a first processing unit 12 for sending at least a voice signal from a first user to a second user. The first processing unit 12 includes a processor 14, a memory 16, an input device 18, and an output device 20. The output device 20 includes at least one modem capable of, for example, 14.4 kbaud communications and operatively connected via wired and/or wireless communication connections to the Internet or other computer networks such as an Intranet, i.e., a private computer network. One skilled in the art would understand that the input device 18 may be implemented at least in part by the modem of the output device 20 to allow input signals from the communication connections to be received. The second processing unit 22 may have a processor, memory, and input and output devices, including at least one modem and associated communication connections, as described above for the first processing unit 12. In an exemplary embodiment, each of the processing units 12, 22 may execute the WEBPHONE™ Internet telephony application available from NetSpeak Corporation, Boca Raton, FL, which is capable of performing the disclosed point-to-point Internet protocol and system 10, as described herein.

[0014] The first processing unit 12 and the second processing unit 22 are operatively connected to the Internet 24 by communication devices and software known in the art, such as an Internet Service Provider (ISP) or an Internet gateway. The processing units 12, 22 may be operatively interconnected through the Internet 24 to a connection server 26, and may also be operatively connected to a mail server 28 associated with the Internet 24.

[0015] The connection server 26 includes a processor 30, a timer 32 for generating time stamps, and a memory such as a database 34 for storing, for example, E-mail and Internet Protocol (IP) addresses of logged-in units. In an exemplary embodiment, the connection server 26 may be a SPARC 5 server or a SPARC 20 server, available from SUN MICROSYSTEMS, INC., Mountain View, CA, having a central processing unit (CPU) as processor 30, an operating system (OS) such as UNIX, for providing timing operations such as maintaining the timer 32, a hard drive or fixed drive, as well as dynamic random access memory (DRAM) for storing the database 34, and a keyboard and display and/or other input and output devices (not shown in FIG. 1). The database 34 may be an SQL database available from ORACLE or INFORMIX.

[0016] In an exemplary embodiment, the mail server 28 may be a Post Office Protocol (POP) Version 3 mail server including a processor, memory, and stored programs operating in a UNIX environment, or, alternatively, another OS, to process E-mail capabilities between processing units and devices over the Internet 24.

[0017] The first processing unit 12 may operate the disclosed point-to-point Internet protocol by a computer program described hereinbelow in conjunction with FIG. 6, which may be implemented from compiled and /or interpreted source code in the C++ programming language and which may be downloaded to the first processing unit 12 from an external computer. The operating computer program may be stored in the memory 16, which may include about 8 MB RAM and/or a hard or fixed drive having about 8 MB. Alternatively, the source code may be implemented in the first processing unit 12 as firmware, as an erasable read only memory (EPROM), etc. It is understood that one skilled in the art would be able to use programming languages other than C++ to implement the disclosed point-to-point network protocol and system 10.

[0018] The processor 14 receives input commands and data from a first user associated with the first processing unit 12 though the input device 18, which may be an input port connected by a wired, optical, or a wireless connection for electromagnetic transmissions, or alternatively may be transferable storage media, such as floppy disks, magnetic tapes, compact disks, or other storage media including the input data from the first user.

[0019] The input device 18 may include a user interface (not shown) having, for example, at least one button actuated by the user to input commands to select from a plurality of operating modes to operate the first processing unit 12. In alternative embodiments, the input device 18 may include a keyboard, a mouse, a touch screen, and/or a data reading device such as a disk drive for receiving the input data from input data files stored in storage media such as a floppy disk or, for example, an 8 mm storage tape. The input device 18 may alternatively include connections to other computer systems to receive the input commands and data therefrom.

[0020] The first processing unit 12 may include a visual interface for use in conjunction with the input device 18 and output device 20 similar to those screens illustrated in FIGS. 5-6, discussed below. It is also understood that alternative devices may be used to receive commands and data from the user, such as keyboards, mouse devices, and graphical user interfaces (GUI) such as WINDOWS™ 3.1 available form MICROSOFT Corporation, Redmond, WA., and other operating systems and GUIs, such as OS/2 and OS/2 WARP, available from IBM CORPORATION, Boca Raton, FL. Processing unit 12 may also include microphones and/or telephone handsets for receiving audio voice data and commands, speech or voice recognition devices, dual tone multifrequency (DTMF) based devices, and/or software known in the art to accept voice data and commands and to operate the first processing unit 12.

[0021] In addition, either of the first processing unit 12 and the second processing unit 22 may be implemented in a personal digital assistant (PDA) providing modem and E-mail capabilities and Internet access, with the PDA providing the input/output screens for mouse interactions or for touchscreen activation as shown, for example, in FIGS. 5-6, as a combination of the input device 18 and output device 20.

[0022] For clarity of explanation, the illustrative embodiment of the disclosed point-to-point Internet protocol and system 10 is presented as having individual functional blocks, which may include functional blocks labeled as "processor" and "processing unit". The functions represented by these blocks may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. For example, the functions of each of the processors and processing units presented herein may be provided by a shared processor or by a plurality of individual processors. Moreover, the use of the functional blocks with accompanying labels herein is not to be construed to refer exclusively to hardware capable of executing software. Illustrative embodiments may include digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, read-only memory (ROM) for storing software performing the operations discussed below, and random access memory (RAM) for storing DSP results. Very large scale integration (VLSI) hardware embodiments, as well as custom VLSI circuitry in combination with a general purpose DSP circuit, may also be provided. Any and all of these embodiments may be deemed to fall within the meaning of the labels for the functional blocks as used herein.

[0023] The processing units 12, 22 are capable of placing calls and connecting to other processing units connected to the Internet 24, for example, via dialup SLIP/PPP lines. In an exemplary embodiment, each processing unit assigns an unsigned long session number, for example, a 32- bit long sequence in a *.ini file for each call. Each call may be assigned a successive session number in sequence, which may be used by the respective processing unit to associate the call with one of the SLIP/PPP lines, to associate a <ConnectOK> response signal with a <Connect Request> signal, and to allow for multiplexing and demultiplexing of inbound and outbound conversations on conference lines, as explained hereinafter.

[0024] For callee (or called) processing units with fixed IP addresses, the caller (or calling) processing unit may open a "socket", i.e. a file handle or address indicating where data is to be sent, and transmit a <Call> command to establish communication with the callee utilizing, for example, datagram services such as Internet Standard network layering as well as transport layering, which may include a Transport Control Protocol (TCP) or a User Datagram Protocol (UDP) on top of the IP. Typically, a processing unit having a fixed IP address may maintain at least one open socket and a called processing unit waits for a <Call> command to assign the open socket to the incoming signal. If all lines are in use, the callee processing unit sends a BUSY signal or message to the callee processing unit. As shown in FIG. 1, the disclosed point-to-point Internet protocol and system 10 operate when a callee processing unit does not have a fixed or predetermined IP address. In the exemplary embodiment and without loss of generality, the first processing unit 12 is the caller processing unit and the second processing unit 22 is the called processing unit. When either of processing units 12, 22 logs on to the Internet via a dial-up connection, the respective unit is provided a dynamically allocated IP address by the a connection service provider.

[0025] Upon the first user initiating the point-to-point Internet protocol when the first user is logged on to the Internet 24, the first processing unit 12 automatically transmits its associated E-mail address and its dynamically allocated IP address to the connection server 26. The connection server 26 then stores these addresses in the database 34 and time stamps the stored addresses using timer 32. The first user operating the first processing unit 12 is thus established in the database 34 as an active on-line party available for communication using the disclosed point-to-point Internet protocol. Similarly, a second user operating the second processing unit 22, upon connection to the Internet 24 through the a connection service provider, is processed by the connection server 26 to be established in the database 34 as an active on-line party.

[0026] The connection server 26 may use the time stamps to update the status of each processing unit; for example, after 2 hours, so that the on-line status information stored in the database 34 is relatively current. Other predetermined time periods, such as a default value of 24 hours, may be configured by a systems operator.

[0027] The first user with the first processing unit 12 initiates a call using, for example, a Send command and/or a command to speeddial an N^{TH} stored number, which may be labeled [SND] and [SPD] [N], respectively, by the input device 18 and/or the output device 20, such as shown in FIGS. 5-6. In response to either the Send or speeddial commands, the first processing unit 12 retrieves from memory 16 a stored E-mail address of the callee corresponding to the N^{TH} stored number. Alternatively, the first user may directly enter the E-mail address of the callee.

[0028] The first processing unit 12 then sends a query, including the E-mail address of the callee, to the connection server 26. The connection server 26 then searches the database 34 to determine whether the callee is logged-in by finding any stored information corresponding to the callee's E-mail address indicating that the callee is active and on-line. If the callee is active and on-line, the connection server 26 then performs the primary point-to-point Internet protocol; i.e. the IP address of the callee is retrieved from the database 34 and sent to the first processing unit 12. The first processing unit 12 may then directly establish the point-to-point Internet communications with the callee using the IP address of the callee.

[0029] If the callee is not on-line when the connection server 26 determines the callee's status, the connection server 26 sends an OFFLINE signal or message to the first processing unit 12. The first processing unit 12 may also display a message such as "Called Party Off-Line" to the first user.

[0030] When a user logs off or goes off-line from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user's information, or by flagging the user as being off-line. The connection server 26 may be instructed to update the user's information in the database 34 by an off-line message, such as a data packet, sent automatically from the processing unit of the user prior to being disconnected from the connection server 26. Accordingly, an off-line user is effectively disabled from making and/or receiving point-to-point Internet communications.

[0031] As shown in FIGS. 2-4, the disclosed secondary point-to-point Internet protocol may be used as an alternative to the primary point-to-point Internet protocol described above, for example, if the connection server 26 is non-responsive, inoperative, and/or unable to perform the primary point-to-point Internet protocol, as a non-responsive condition. Alternatively, the disclosed secondary point-to-point Internet protocol may be used independent of the primary point-to-point Internet protocol. In the disclosed secondary point-to-point Internet protocol, the first processing unit 12 sends a <ConnectRequest> message via E-mail over the Internet 24 to the mail server 28. The E-mail including the <ConnectRequest> message may have, for example, the subject

["wp#XXXXXXXX#nnn.nnn.nnn.#emailAddr]

where nnn.nnn.nnn.nnn. is the current (i.e. temporary or permanent) IP address of the first user, and XXXXXXXX is a session number, which may be unique and associated with the request of the first user to initiate point-to-point communication with the second user.

[0032] As described above, the first processing unit 12 may send the <ConnectRequest> message in response to an unsuccessful attempt to perform the primary point-to-point Internet protocol. Alternatively, the first processing unit 12 may send the <ConnectRequest> message in response to the first user initiating a SEND command or the like.

[0033] After the <ConnectRequest> message via E-mail is sent, the first processing unit 12 opens a socket and waits to detect a response from the second processing unit 22. A timeout timer, such as timer 32, may be set by the first processing unit 12, in a manner known in the art, to wait for a predetermined duration to receive a <ConnectOK> signal. The processor 14 of the first processing unit 12 may cause the output device 20 to output a Ring signal to the user, such as an audible ringing sound, about every 3 seconds. For example, the processor 14 may output a *.wav file, which may be labeled RING.WAV, which is processed by the output device 20 to output an audible ringing sound.

[0034] The mail server 28 then polls the second processing unit 22, for example, every 3-5 seconds, to deliver the E-mail. Generally, the second processing unit 22 checks the incoming lines, for example, at regular intervals to wait for and to detect incoming E-mail from the mail server 28 through the Internet 24.

[0035] Typically, for sending E-mail to users having associated processing units operatively connected to a host computer or server operating an Internet gateway, E-mail for a specific user may be sent over the Internet 24 and directed to the permanent IP address or the SLIP/PPP account designation of the host computer, which then assigns a temporary IP address to the processing unit of the specified user for properly routing the E-mail. The E-mail signal may include a name or other designation such as a user name which identifies the specific user regardless of the processing unit assigned to the user; that is, the host computer may track and store the specific device where a specific user is assigned or logged on, independent of the IP address system, and so the host computer may switch the E-mail signal to the device of the specific user. At that time, a temporary IP address may be generated or assigned to the specific user and device.

[0036] Upon detecting and/or receiving the incoming E-mail signal from the first processing unit 12, the second processing unit 22 may assign or may be assigned a temporary IP address. Therefore, the delivery of the E-mail through the Internet 24 provides the second processing unit 22 with a session number as well as IP addresses of both the first processing unit 12 and the second processing unit 22.

[0037] Point-to-point communication may then be established by the processing unit 22 processing the E-mail signal to extract the <ConnectRequest> message, including the IP address of the first processing unit 12 and the session number. The second processing unit 22 may then open a socket and generate a <ConnectOK> response signal, which includes the temporary IP address of the second processing unit 22 as well as the session number of the first processing unit.

[0038] The second processing unit 22 sends the <ConnectOK> signal directly over the Internet 24 to the IP address of the first processing unit 12 without processing by the mail server 28, and a timeout timer of the second processing unit 22 may be set to wait and detect a <Call> signal expected from the first processing unit 12.

[0039] Realtime point-to-point communication of audio signals over the Internet 24, as well as video and voicemail, may thus be established and supported without requiring permanent IP addresses to be assigned to either of the users or processing units 12, 22. For the duration of the realtime point-to-point link, the relative permanence of the current IP addresses of the processing units 12, 22 is sufficient, whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).

[0040] In the exemplary embodiment, a first user operating the first processing unit 12 is not required to be notified by the first processing unit 12 that an E-mail is being generated and sent to establish the point-to-point link with the second user at the second processing unit 22. Similarly, the second user is not required to be notified by the second processing unit 22 that an E-mail has been received and/or a temporary IP address is associated with the second processing unit 22. The processing units 12, 22 may perform the disclosed point-to-point Internet protocol automatically upon initiation of the point-to-point communication command by the first user without displaying the E-mail interactions to either user. Accordingly, the disclosed point-to-point Internet protocol may be transparent to the users. Alternatively, either of the first and second users may receive, for example, a brief message of "CONNECTION IN PROGRESS" or the like on a display of the respective output device of the processing units 12, 22.

[0041] After the initiation of either the primary or the secondary point-to-point Internet protocols described above in conjunction with FIGS. 1-2, the point-to-point communication link over the Internet 24 may be established as shown in FIGS. 3-4 in a manner known in the art. For example, referring to FIG. 3, upon receiving the <ConnectorOK> signal from the second processing unit 22, the first processing unit 12 extracts the IP address of the second processing unit 22 and the session number, and the session number sent from the second processing unit 22 is then checked with the session number originally sent from the first processing unit 12 in the <ConnectRequest> message as E-mail. If the session numbers sent and received by the processing unit 12 match, then the first processing unit 12 sends a <Call> signal directly over the Internet 24 to the second processing unit 22; i.e. using the IP address of the second processing unit 22 provided to the first processing unit 12 in the <ConnectOK> signal.

[0042] Upon receiving the <Call> signal, the second processing unit 22 may then begin a ring sequence, for example, by indicating or annunciating to the second user that an incoming call is being received. For example, the word "CALL" may be displayed on the output device of the second processing unit 22. The second user may then activate the second processing unit 22 to receive the incoming call.

[0043] Referring to FIG. 4, after the second processing unit 22 receives the incoming call, realtime audio and/or video conversations may be conducted in a manner known in the art between the first and second users through the Internet 24, for example, by compressed digital audio signals. Each of the processing units 12, 22 also display to each respective user the words "IN USE" to indicate that the point-to-point communication link is established and audio or video signals are being transmitted.

[0044] In addition, either user may terminate the point-to-point communication link by, for example, activating a termination command, such as by activating an [END] button or icon on a respective processing unit, causing the respective processing unit to send an <End> signal which causes both processing units to terminate the respective sockets, as well as to perform other cleanup commands and functions known in the art.

[0045] FIGS. 5-6 illustrate examples of display screens 36 which may be output by a respective output device of each processing unit 12, 22 of FIGS. 1-4 for providing the disclosed point-to-point Internet protocol and system 10. Such display screens may be displayed on a display of a personal computer (PC) or a PDA in a manner known in the art.

[0046] As shown in FIG. 5, a first display screen 36 includes a status area 38 for indicating, for example, a called user by name and/or by IP address or telephone number; a current function such as C2; a current time; a current operating status such as "IN USE", and other control icons such as a down arrow icon 40 for scrolling down a list of parties on a current conference line. The operating status may include such annunciators as "IN USE," "IDLE," "BUSY," "NO ANSWER," "OFFLINE," "CALL," "DIALING," "MESSAGES," and "SPEEDDIAL."

[0047] Other areas of the display screen 36 may include activation areas or icons for actuating commands or entering data. For example, the display screen 36 may include a set of icons 42 arranged in columns and rows including digits 0-9 and commands such as END, SND, HLD, etc. For example, the END and SND commands may be initiated as described above, and the HLD icon 44 may be actuated to place a current line on hold. Such icons may also be configured to substantially simulate a telephone handset or a cellular telephone interface to facilitate ease of use, as well as to simulate function keys of a keyboard. For example, icons labeled L1-L4 may be mapped to function keys F1-F4 on standard PC keyboards, and icons C1-C3 may be mapped to perform as combinations of function keys, such as CTRL-F1, CTRL-F2, and CTRL-F3, respectively. In addition, the icons labeled L1-L4 and C1-C3 may include circular regions which may simulate light emitting diodes (LEDs) which indicate that the function or element represented by the respective icon is active or being performed.

[0048] Icons L1-L4 may represent each of 4 lines available to the caller, and icons C1-C3 may represent conference calls using at least one line to connect, for example, two or more parties in a conference call. The icons L1-L4 and C1-C3 may indicate the activity of each respective line or conference line. For example, as illustrated in FIG. 5, icons L1-L2 may have lightly shaded or colored circles, such as a green circle, indicating that each of lines 1 and 2 are in use, while icons L3-L4 may have darkly shaded or color circles, such as a red or black circle, indicating that each of lines 3 and 4 are not in use. Similarly, the lightly shaded circle of the icon labeled C2 indicates that the function corresponding to C2 is active, as additionally indicated in the status are 38, while darkly shaded circles of icons labeled C1 and C3 indicate that such corresponding functions are not active.

[0049] The icons 42 are used in conjunction with the status area 38. For example, using a mouse for input, a line that is in use, as indicated by the lightly colored circle of the icon, may be activated to indicate a party's name by clicking a right mouse button for 5 seconds until another mouse click is actuated or the [ESC] key or icon is actuated. Thus, the user may switch between multiple calls in progress on respective lines.

[0050] Using the icons as well as an input device such as a mouse, a user may enter the name or alias or IP address, if known, of a party to be called by either manually entering the name, by using the speeddial feature, or by double clicking on an entry in a directory stored in the memory, such as the memory 16 of the first processing unit 12, where the directory entries may be scrolled using the status area 38 and the down arrow icon 40.

[0051] Once a called party is listed in the status area 38 as being active on a line, the user may transfer the called party to another line or a conference line by clicking and dragging the status area 38, which is represented by a reduced icon 46. Dragging the reduced icon 46 to any one of line icons L1-L4 transfers the called party in use to the selected line, and dragging the reduced icon 46 to any one of conference line icons C1-C3 adds the called party to the selected conference call.

[0052] Other features may be supported, such as icons 48-52, where icon 48 corresponds to, for example, an ALT-X command to exit the communication facility of a processing unit, and icon 50 corresponds to, for example, an ALT-M command to minimize or maximize the display screen 36 by the output device of the processing unit. Icon 52 corresponds to an OPEN command, which may, for example, correspond to pressing the O key on a keyboard, to expand or contract the display screen 36 to represent the opening and closing of a cellular telephone. An "opened" configuration is shown in FIG. 5, and a "closed" configuration is shown in FIG. 6. In the "opened" configuration, additional features such as output volume (VOL) controls, input microphone (MIC) controls, waveform (WAV) sound controls, etc.

[0053] The use of display screens such as those shown in FIGS. 5-6 provided flexibility in implementing various features available to the user. It is to be understood that additional features such as those known in the art may be supported by the processing units 12, 22.

[0054] Alternatively, it is to be understood that one skilled in the art may implement the processing units 12, 22 to have the features of the display screens in FIGS. 5-6 in hardware; i.e. a wired telephone or wireless cellular telephone may include various keys, LEDs, liquid crystal displays (LCDs), and touchscreen actuators corresponding to the icons and features shown in FIGS. 5-6. In addition, a PC may have the keys of a keyboard and mouse mapped to the icons and features shown in FIGS. 5-6.

[0055] Referring to FIG. 7, the disclosed point-to-point Internet protocol and system 10 is illustrated. First processing unit 12 initiates the point-to-point Internet protocol in step 56 by sending a query from the first processing unit 12 to the connection server 26. If connection server 26 is operative to perform the point-to-point Internet protocol, in step 58, first processing unit 12 receives an on-line status signal from the connection server 26, such signal may include the IP address of the callee or a "Callee Off-Line" message. Next, first processing unit 12 performs the primary point-to-point Internet protocol in step 60, which may include receiving, at the first processing unit 12, the IP address of the callee if the callee is active and on-line. Alternatively, processing unit 60 may initiate and perform the secondary point-to-point Internet protocol in step 62, if the called party is not active and/or on-line.

[0056] Referring to FIG. 8, in conjunction with FIGS. 1 and 3-4, the disclosed point-to-point Internet protocol and system 10 is illustrated. Connection server 26 starts the point-to-point Internet protocol, in step 64, and timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66. Connection server 26 receives a query from a first processing unit 12 in step 68 to determine whether a second user or second processing unit 22 is logged-in to the Internet 24, with the second user being specified, for example, by an E-mail address. Connection server 26 retrieves the IP address of the specified user from the database 34 in step 70, if the specified user is logged-in to the Internet, and sends the retrieved IP address to the first processing unit 12 in step 72 to enable first processing unit 12 to establish point-to-point communications with the specified second user.

[0057] The disclosed secondary point-to-point Internet protocol operates as shown in FIG. 9. First processing unit 12 generates an E-mail signal, including a session number and a first IP address corresponding to a first processing unit in step 76. First processing unit 12 transmits the E-mail signal as a <ConnectRequest> signal to the Internet 24 in step 78. The E-mail signal is delivered through the Internet 24 using a mail server 28 to the second processing unit 22 in step 80. Second processing unit 22 extracts the session number and the first IP address from the E-mail signal in step 82 and transmits or sends the session number and a second IP address corresponding to the second processing unit 22, back to the first processing unit 12 through the Internet 24, in step 84. First processing unit 12 verifies the session number received from the second processing unit 22 in step 86, and establishes a point-to-point Internet communication link between the first processing unit 12 and second processing unit 22 using the first and second IP addresses in step 88.

[0058] While the disclosed point-to-point Internet protocols and system have been particularly shown and described with reference to the preferred embodiments, it is understood by those skilled in the art that various modifications in form and detail may be made therein without departing from the scope and spirit of the invention. Accordingly, modifications such as those suggested above, but not limited thereto, are to be considered within the scope of the invention.

Claims

1. A method for establishing point-to-point Internet communication between a plurality of processing units **characterized by** the steps:

   a) transmitting an E-mail signal, including a first IP address assigned to a first processing unit (12);
   b) processing the E-mail signal through the Internet to deliver the E-mail signal to a second processing unit (22); and
   c) transmitting a second IP address to the first processing unit for establishing a point-to-point communication link between the first and second processing units through the Internet.

2. The method of claim 1, further **characterized by**:

   a1) generating the E-mail signal from the first IP address corresponding to the first processing unit before the step (a) of transmitting the E-mail signal.

3. The method of claim 1, further **characterized by**:

   a1) generating the E-mail signal from a session number before the step (a) of transmitting the E-mail signal.

4. The method of claim 1, **characterized** in that the step of processing the E-mail signal further comprises the step of:

   b1) processing the E-mail signal using a mail server operatively connected to the second processing unit.

5. The method of claim 1, further **characterized by**:

   b1) generating a connection signal (CONNECTOK) including the second IP address at the second processing unit before the step (c) of transmitting the second IP address to the first processing unit, and

   wherein the step (c) of transmitting the second IP address includes the step of transmitting the connection signal from the second processing unit to the first processing unit.

6. A system for enabling point-to-point communications between a first and a second processing unit over the Internet, **characterized by**:

   a) means for transmitting from the first processing unit (12) to the Internet an E-mail signal, including a first IP address assigned to the first processing unit;
   b) means (28) for processing the E-mail signal through the Internet to deliver the E-mail signal to the second processing (22) unit; and
   c) means for transmitting a second IP address to the first processing unit for establishing a point-to-point communication link between the first and second processing units through the Internet.

7. The system of claim 6 comprising a server which is **characterized by**:

   a processor;
   a memory operatively coupled to the processor;
   a network interface logic operatively coupled to the processor and the memory and configured to connect the server to a computer network; and

mail processing logic responsive to an E-mail signal from the first processing unit and configured to provide the E-mail signal to the second processing unit, the E-mail signal comprising the network protocol address of the first processing unit.

8. The system of claim 6, further **characterized by**:

b1) means for generating a connection signal (CONNECTOK) including the second IP address in the second processing unit, and for transmitting the connection signal from the second processing unit to the first processing unit.

9. The system of claim 6, wherein the first processing unit comprises a processor for executing the point-to-point Internet protocol, **characterized by** means for generating an E-mail signal, including a first IP address, and by means for transmitting the E-mail signal through the Internet to the second processing unit for establishing a point-to-point communication link to the first processing unit.

10. The system of claim 9, **characterized in that** the processor is adapted to generate the E-mail signal from the first IP address corresponding to the first processing unit.

11. The apparatus of claim 9, **characterized in that** the processor is adapted to wait to detect a response from the second processing unit.

12. The system according to one of the claims 6-11, **characterized in that** the first IP address is dynamically assigned to the first processing unit.

13. The system according to one of the claims 6-12, comprising a mail server (28) for processing the E-mail signal through the Internet to deliver the E-mail to the second processing unit (22) for establishing a point-to-point communication link between the first and second processing unit through the Internet.

14. The system of claim 13, wherein the second processing unit comprises a processor adapted for receiving the E-mail signal from the mail server and for generating a connection signal (CONNECTOK) including a second IP address and for transmitting the connection signal to the first processing unit for establishing the point-to-point communication link to the first processing unit.

15. A computer program product for establishing point-to-point Internet communication between a plurality of processing units, the computer program product having a computer usable medium containing computer readable program code, comprising:

a) program code for transmitting an E-mail signal including a first IP address assigned to a first processing unit (12);
b) program code for processing the E-mail signal through the Internet to deliver the E-mail signal to a second processing unit (22);
c) program code for transmitting a second IP address to the first processing unit; and
d) program code for establishing in response to receiving the second IP address in the first processing unit a point-to-point communication link between the first and second processing units through the Internet.

16. The computer program product of claim 15, further **characterized by**:

b1) program code for generating a connection signal (CONNECTOK) including the second IP address at the second processing unit; and
c1) program code for transmitting the connection signal from the second processing unit to the first processing unit.

17. The computer program product of claim 15, further comprising in a memory (16) of the first processing unit:

program code for performing a point-to-point Internet protocol;
program code for generating an E-mail signal, including a first IP address; and
program code for use of a mail server (28) for processing the E-mail signal through the Internet to deliver the E-mail to the second processing unit for establishing a point-to-point communication link between the first and second processing unit.

18. The computer program product of claim 15, further comprising in a memory of the second processing unit:

second program code for performing a point-to-point Internet protocol; program code for receiving the E-mail signal from a mail server; and program code for generating a connection signal including a second IP address for establishing the point-to-point communication link to the first processing unit.

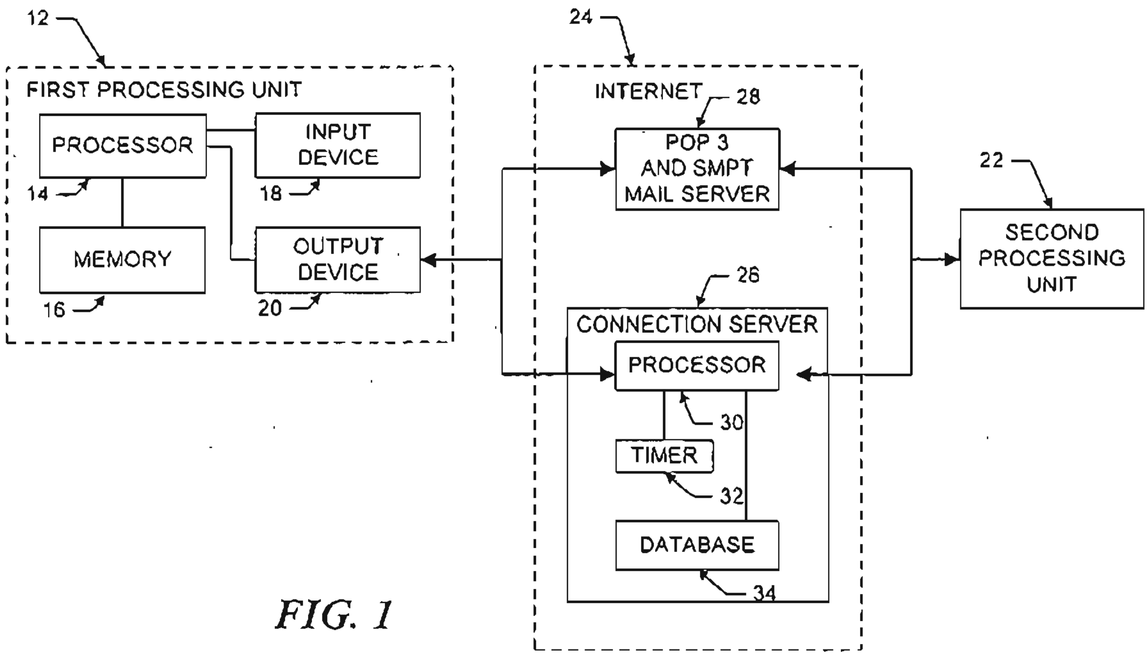

12
FIRST PROCESSING UNIT
PROCESSOR
INPUT DEVICE
14
18
MEMORY
OUTPUT DEVICE
16
20
24
INTERNET
28
POP 3 AND SMPT MAIL SERVER
26
CONNECTION SERVER
PROCESSOR
30
TIMER
32
DATABASE
34
22
SECOND PROCESSING UNIT

FIG. 1

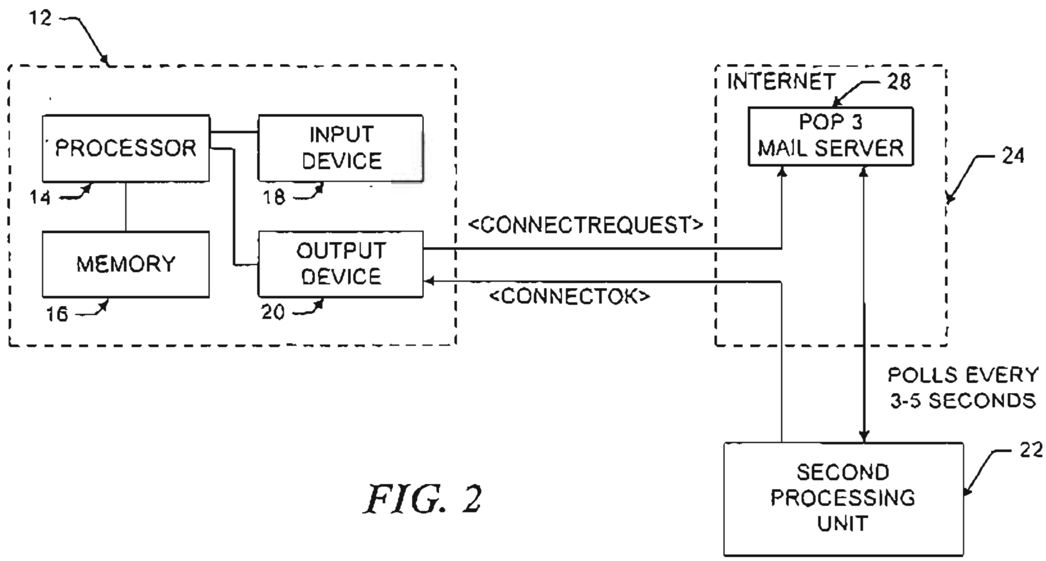


FIG. 2

EP 1 379 039 A2

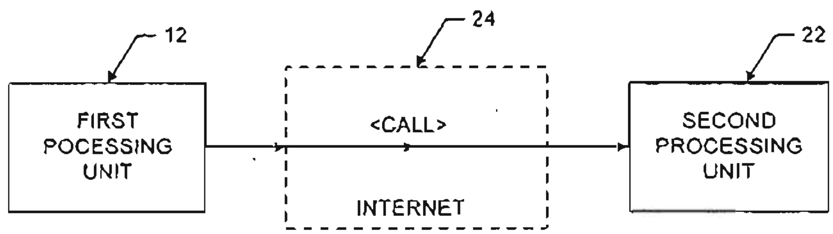

12
24
22
FIRST
POCESSING
UNIT
<CALL>
INTERNET
SECOND
PROCESSING
UNIT

FIG. 3

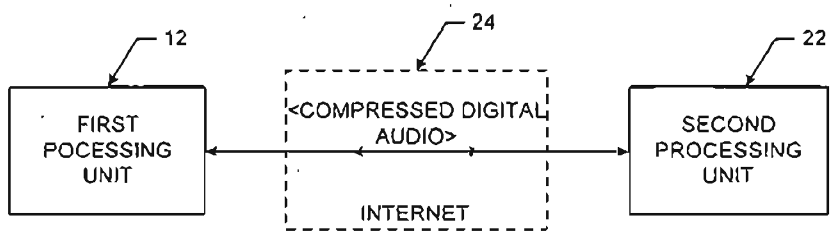

12
24
22
FIRST
POCESSING
UNIT
<COMPRESSED DIGITAL
AUDIO>
INTERNET
SECOND
PROCESSING
UNIT

FIG. 4

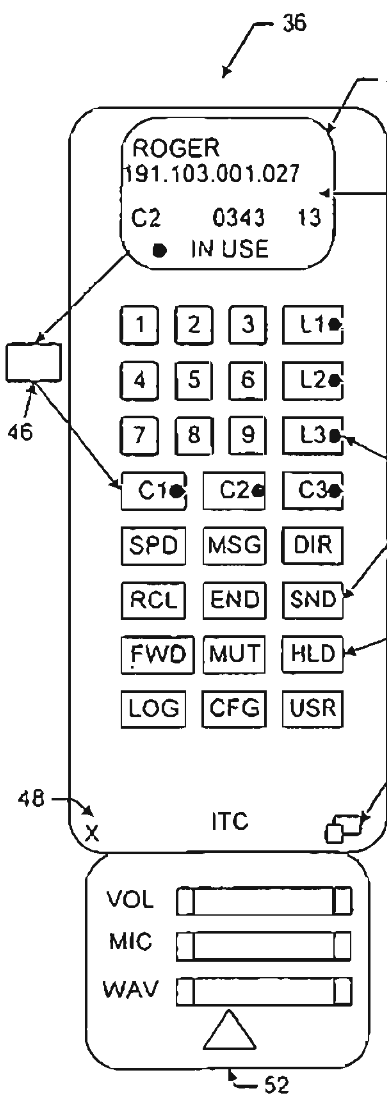

36
40
ROGER
191.103.001.027
38
C2 0343 13
IN USE
1
2
3
L1
4
5
6
L2
7
8
9
L3
46
C1
C2
C3
42
SPD
MSG
DIR
RCL
END
SND
FWD
MUT
HLD
44
LOG
CFG
USR
48
50
X
ITC
VOL
MIC
WAV
52

FIG. 5

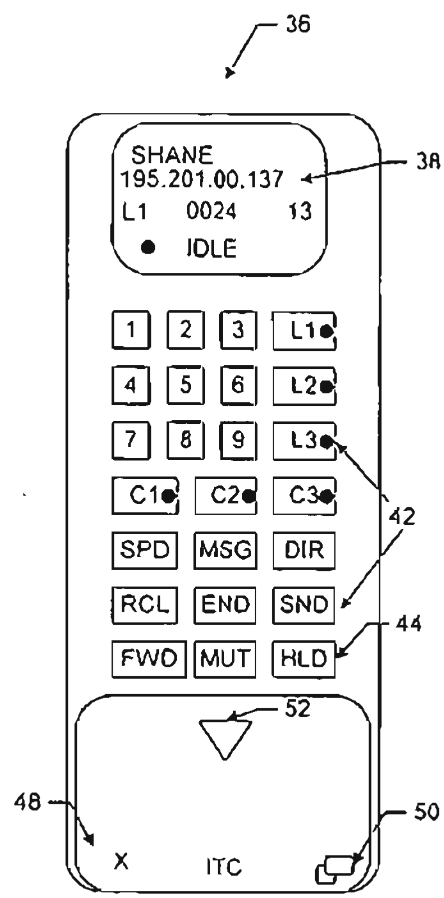

36
SHANE
195.201.00.137
38
L1 0024 13
IDLE
1
2
3
L1
4
5
6
L2
7
8
9
L3
C1
C2
C3
42
SPD
MSG
DIR
RCL
END
SND
FWD
MUT
HLD
44
52
48
50
X
ITC

FIG. 6

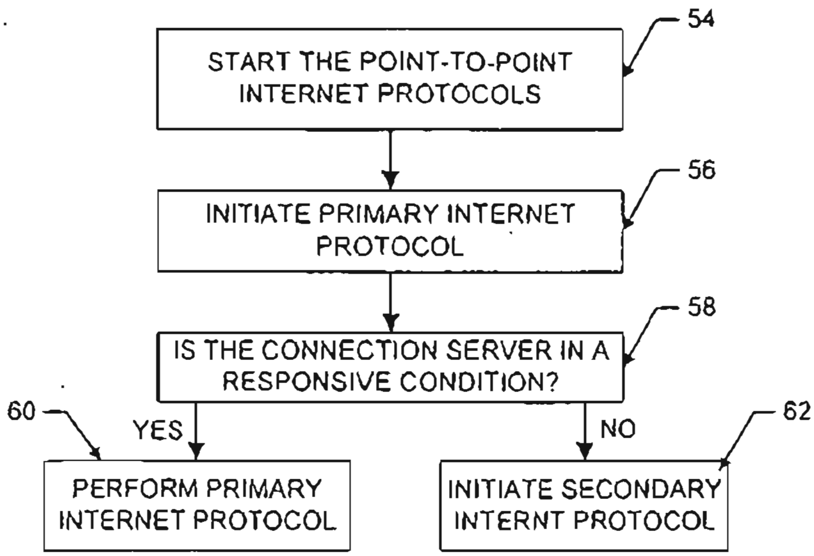

54
START THE POINT-TO-POINT INTERNET PROTOCOLS
56
INITIATE PRIMARY INTERNET PROTOCOL
58
IS THE CONNECTION SERVER IN A RESPONSIVE CONDITION?
60
YES
NO
62
PERFORM PRIMARY INTERNET PROTOCOL
INITIATE SECONDARY INTERNT PROTOCOL

FIG. 7

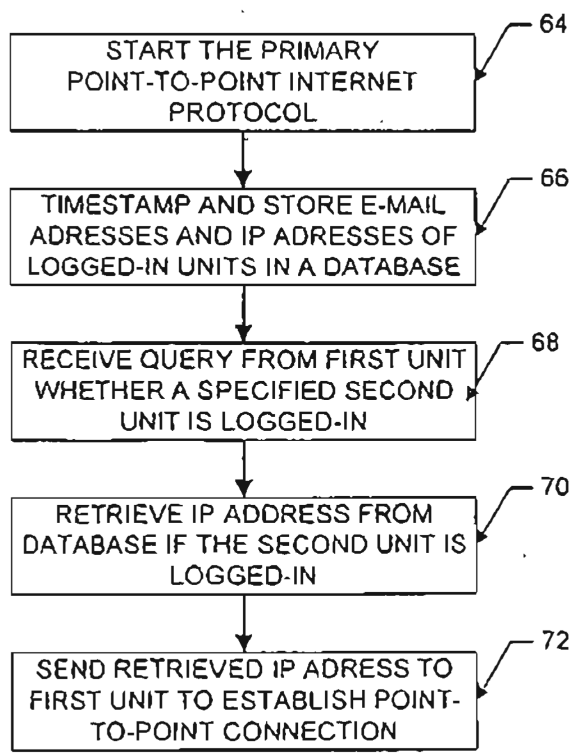

64
START THE PRIMARY POINT-TO-POINT INTERNET PROTOCOL
66
TIMESTAMP AND STORE E-MAIL ADRESSES AND IP ADRESSES OF LOGGED-IN UNITS IN A DATABASE
68
RECEIVE QUERY FROM FIRST UNIT WHETHER A SPECIFIED SECOND UNIT IS LOGGED-IN
70
RETRIEVE IP ADDRESS FROM DATABASE IF THE SECOND UNIT IS LOGGED-IN
72
SEND RETRIEVED IP ADRESS TO FIRST UNIT TO ESTABLISH POINT-TO-POINT CONNECTION

FIG. 8

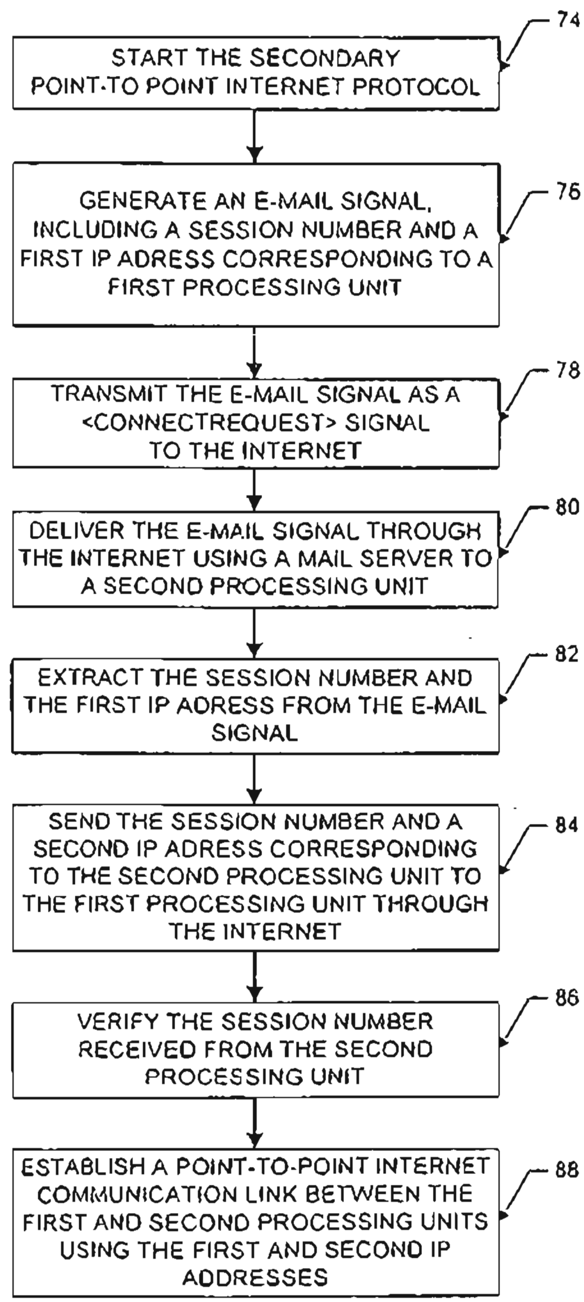

START THE SECONDARY POINT-TO POINT INTERNET PROTOCOL
74
GENERATE AN E-MAIL SIGNAL, INCLUDING A SESSION NUMBER AND A FIRST IP ADRESS CORRESPONDING TO A FIRST PROCESSING UNIT
76
TRANSMIT THE E-MAIL SIGNAL AS A <CONNECTREQUEST> SIGNAL TO THE INTERNET
78
DELIVER THE E-MAIL SIGNAL THROUGH THE INTERNET USING A MAIL SERVER TO A SECOND PROCESSING UNIT
80
EXTRACT THE SESSION NUMBER AND THE FIRST IP ADRESS FROM THE E-MAIL SIGNAL
82
SEND THE SESSION NUMBER AND A SECOND IP ADRESS CORRESPONDING TO THE SECOND PROCESSING UNIT TO THE FIRST PROCESSING UNIT THROUGH THE INTERNET
84
VERIFY THE SESSION NUMBER RECEIVED FROM THE SECOND PROCESSING UNIT
86
ESTABLISH A POINT-TO-POINT INTERNET COMMUNICATION LINK BETWEEN THE FIRST AND SECOND PROCESSING UNITS USING THE FIRST AND SECOND IP ADDRESSES
88

FIG. 9

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 379 039 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
10.03.2004 Bulletin 2004/11

(43) Date of publication A2:
07.01.2004 Bulletin 2004/02

(21) Application number: 03022287.1

(22) Date of filing: 25.09.1996

(51) Int Cl.7: H04L 12/58, H04L 29/06, H04L 29/12

(84) Designated Contracting States:
CH DE ES FI FR GB IE IT LI NL SE

(30) Priority: 25.09.1995 US 533115

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
96933928.2 / 0 852 868

(71) Applicant: Netspeak Corporation
Boca Raton, FL 33487 (US)

(72) Inventor: Hutton, Glenn W.
Miami, FL 33196 (US)

(74) Representative: Kindermann, Manfred
Patentanwalt,
Sperberweg 29
71032 Böblingen (DE)

(54) **Point-to-point communication using e-mail to establish dynamic network addresses**

(57) In a computer system having an audio transducer and a display device and being operatively coupled to other computers over a computer network (24), such as the Internet, means are included for establishing a point-to-point communication link between processes. The means provide for transmitting from a first processing unit (12) to the Internet an E-mail signal, including a first IP address assigned to the first processing unit, and for processing the E-mail signal through the Internet to deliver the E-mail signal to a second processing unit (22). Further means are provided for transmitting a second IP address to the first processing unit for establishing a point-to-point communication link between the first and second processing units through the Internet.

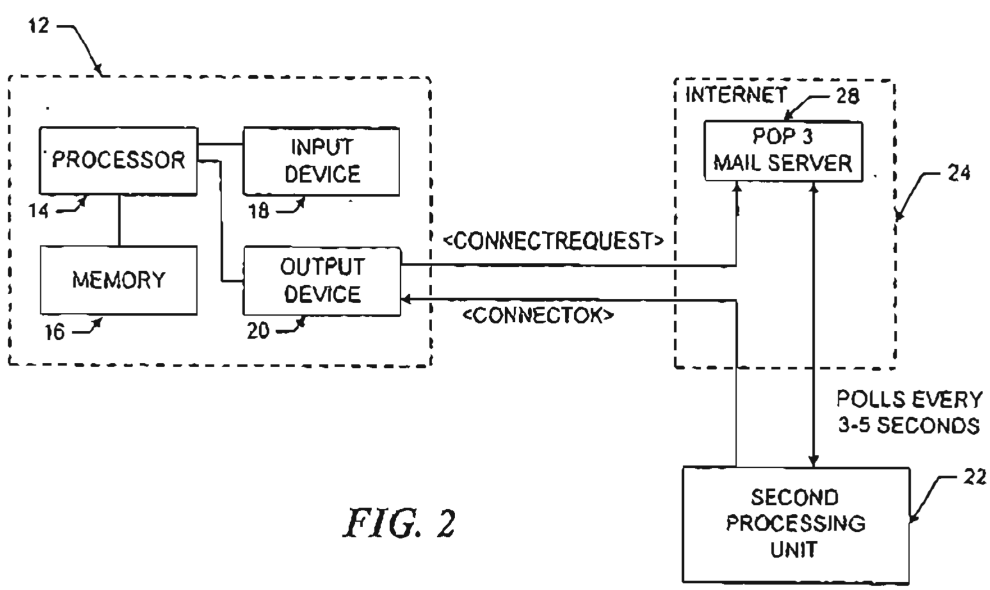


FIG. 2

EP 1 379 039 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 03 02 2287

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	EP 0 581 722 A (YEDA RES & DEV) 2 February 1994 (1994-02-02) * page 2, line 3 - page 3, line 56 * * page 5, line 33 - page 6, line 13 * * page 7, line 10 - line 17 * * page 8, line 19 - line 24 * ---	1-18	H04L12/58 H04L29/06 H04L29/12
A	US 5 440 547 A (ESAKI HIROSHI ET AL) 8 August 1995 (1995-08-08) * column 1, line 9 - column 4, line 48 * * column 15, line 67 - column 17, line 6 * ---	1-18	
A	GB 2 283 645 A (DIGITAL EQUIPMENT INT) 10 May 1995 (1995-05-10) * page 1, line 3 - page 3, line 22 * * page 5, line 22 - line 26 * * page 6, line 4 - page 7, line 28 * * page 11, line 17 - page 12, line 21 * ---	1,6,15	
A	US 5 287 103 A (KASPRZYK MARLON Z ET AL) 15 February 1994 (1994-02-15) * the whole document * ---	1,6,15	TECHNICAL FIELDS SEARCHED (Int.Cl.7) H04L
A	US 5 410 754 A (FAVREAU KEITH ET AL) 25 April 1995 (1995-04-25) * column 1, line 6 - column 3, line 41 * * column 15, line 13 - line 16 * -----	1,6,15	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	12 January 2004	Vaskimo, K

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 03 02 2287

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

12-01-2004

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
EP 0581722	A	02-02-1994	EP	0581722 A1	02-02-1994
			JP	6216937 A	05-08-1994
US 5440547	A	08-08-1995	JP	6318951 A	15-11-1994
			JP	6261094 A	16-09-1994
GB 2283645	A	10-05-1995	NONE		
US 5287103	A	15-02-1994	NONE		
US 5410754	A	25-04-1995	US	5796742 A	18-08-1998

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 379 050 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
07.01.2004 Bulletin 2004/02

(51) Int Cl.7: H04L 29/06, H04L 12/58

(21) Application number: 03022288.9

(22) Date of filing: 25.09.1996

(84) Designated Contracting States:
CH DE ES FI FR GB IE IT LI NL SE

(30) Priority: 25.09.1995 US 533115

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
96933928.2 / 0 852 868

(71) Applicant: Netspeak Corporation
Boca Raton, FL 33487 (US)

(72) Inventor: Hutton, Glenn W.
Miami, FL 33196 (US)

(74) Representative: Kindermann, Manfred
Patentanwalt,
Sperberweg 29
71032 Böblingen (DE)

Remarks:
This application was filed on 02 - 10 - 2003 as a divisional application to the application mentioned under INID code 62.

(54) Server mediated point-to-point communication over a computer network

(57) In a computer system (12) having an audio transducer and a display device and being operatively coupled to other computers (22) and a server (26) over a computer network (24), means are described for establishing a point-to-point communication link between computer systems. The means provide for transmitting from a first process to the server a query as to whether a second process is connected to the computer network and for receiving a network protocol address of the second process from the server when the second process is connected to the computer network. In response to the received network protocol of the second process a point-to-point communication link is established between the first process and the second process over the computer network.

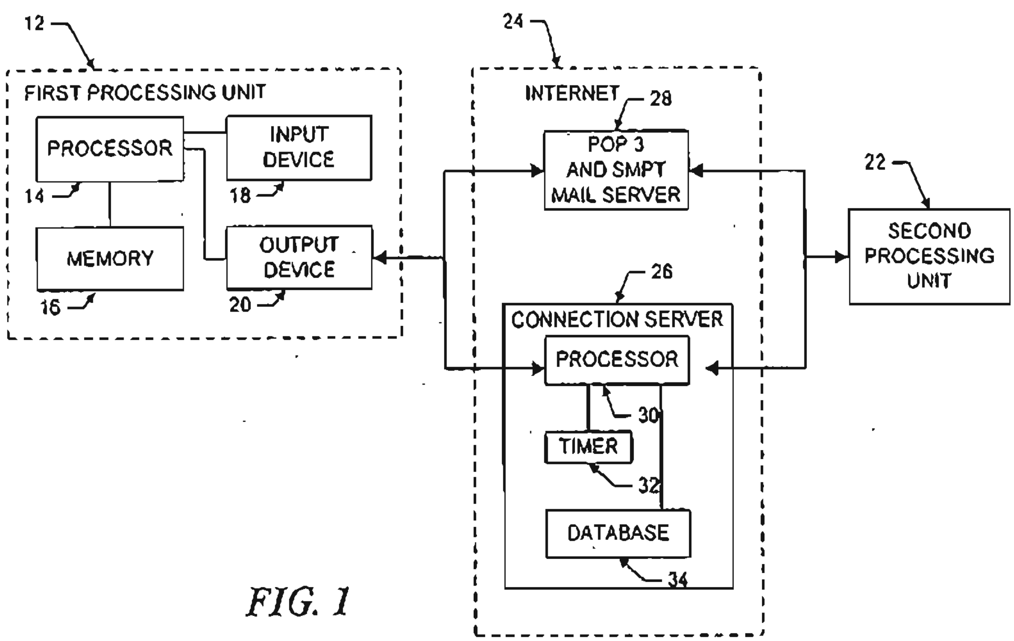


FIG. 1

EP 1 379 050 A2

Description

FIELD OF THE INVENTION

[0001] The present invention relates in general to data processing systems, and more specifically, to an apparatus, a method and a computer program product for facilitating audio communications over computer networks.

BACKGROUND OF THE INVENTION

[0002] The increased popularity of on-line services such as AMERICA ONLINE (TM), COMPUSERVE (R), and other services such as Internet gateways have spurred applications to provide multimedia contents, including video and voice clips, to online users. An example of an online voice clip application is VOICE E-MAIL FOR WINCIM and VOICE E-MAIL FOR AMERICA ONLINE (TM), available from Bonzi Software, as described in "Simple Utilities Send Voice E-Mail Online", MULTIMEDIA WORLD, VOL. 2, NO. 9, August 1995, p. 52. Using such Voice E-Mail software, a user may create an audio message to be sent to a predetermined E-mail address specified by the user.

[0003] Generally, devices interfacing with the Internet and other online services may communicate with each other upon establishing respective device addresses. One type of device address is the Internet Protocol (IP) address, which acts as a pointer to the device associated with the IP address. A typical device may have a Serial Line Internet Protocol or Point-to-Point Protocol (SLIP/PPP) account with a permanent IP address for receiving E-mail, voicemail, and the like over the Internet. E-mail and voicemail is generally intended to convey text, audio, etc., with any routing information such as an IP address and routing headers generally being considered an artifact of the communication, or even gibberish to the recipient.

[0004] Devices such as a host computer or server of a company may include multiple modems for connection of users to the Internet, with a temporary IP address allocated to each user. For example, the host computer may have a general IP address "XXX.XXX.XXX", and each user may be allocated a successive IP address of XXX.XXX.XXX.10, XXX.XXX.XXX.11, XXX.XXX.XXX.12, etc. Such temporary IP addresses may be reassigned or recycled to the users, for example, as each user is successively connected to an outside party. For example, a host computer of a company may support a maximum of 254 IP addresses which are pooled and shared between devices connected to the host computer.

[0005] Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet. For example, global real-time video conferencing has been implemented using dedicated IP addresses and mechanisms known as reflectors.

[0006] A technique for matching domain names to Internet Protocol addresses is described in the text entitled "Internetworking With TCP/IP", 2nd Edition, by Douglas E. Comer, November 1992, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A. Comer describes a domain name system and cooperative systems of name servers for matching domain names to network addresses. Each name server is a server program that supplies mapping of domain names to IP addresses. The system described in Comer, however, is not designed for use with network nodes whose network names or name to address bindings change frequently.

[0007] International Publication WO 92/19054 discloses a network monitoring system including an address tracking module which uses passive monitoring of all packet communications over a local area network to maintain a name table of IP address mappings. The disclosed address tracking module is capable of monitoring only a small number of nodes on a local area network and is not suitable for use with a multitude of nodes over a wide area network.

[0008] Due to the dynamic nature of temporary IP addresses of some devices accessing the Internet, point-to-point communications in realtime of voice and video have been generally difficult to attain.

SUMMARY OF THE INVENTION

[0009] In a computer system having an audio transducer and a display device and being operatively coupled to other computers and a server over a computer network, means are provided for establishing a point-to-point communication link between the computer system and a second computer system over the computer network. The invention, as defined in the claims, comprises (a) means for transmitting from the first process to a server a query as to whether a second process is connected to the computer network; (b) means for receiving a network protocol address of the second process from the server when the second process is connected to the computer network, and (c) means, responsive to the network protocol of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.

[0010] The Invention is in particular suitable for being used in connection with computer networks, such as the Internet, wherein the processing unit does not have a fixed or predetermined network protocol address. The invention thus provides for a protocol by which the processing units report their dynamically assigned network protocol address to a server once they are logging on the computer network. The server maintains and retrieves such information upon request from a calling processing unit.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The features of the invention will become more readily apparent and may be better understood by referring to the following detailed description of an illustrative embodiment of the present invention, taken in conjunction with the accompanying drawings, in which:

FIG. 1 illustrates, in block diagram format, a system for the disclosed point-to-point Internet protocol;
FIG. 2 illustrates, in block diagram format, the system using a secondary point-to-point Internet protocol;
FIG. 3 illustrates, in block diagram format, the system of FIGS. 1-2 with the point-to-point Internet protocol established;
FIG. 4 is another block diagram of the system of FIGS 1-2 with audio communications being conducted;
FIG. 5 illustrates a display screen for a processing unit;
FIG. 6 illustrates another display screen for a processing unit;
FIG. 7 illustrates a flowchart of the initiation of the point-to-point Internet protocols;
FIG. 8 illustrates a flowchart of the performance of the primary point-to-point Internet protocols; and
FIG. 9 illustrates a flowchart of the performance of the secondary point-to-point Internet protocol.

DETAILED DESCRIPTION

[0012] Referring now in specific detail to the drawings, with like reference numerals identifying similar or identical elements, as shown in FIG. 1, the present disclosure describes a point-to-point network protocol and system 10 for using such a protocol.

[0013] In an exemplary embodiment, the system 10 includes a first processing unit 12 for sending at least a voice signal from a first user to a second user. The first processing unit 12 includes a processor 14, a memory 16, an input device 18, and an output device 20. The output device 20 includes at least one modem capable of, for example, 14.4 kbaud communications and operatively connected via wired and/or wireless communication connections to the Internet or other computer networks such as an Intranet, i.e., a private computer network. One skilled in the art would understand that the input device 18 may be implemented at least in part by the modem of the output device 20 to allow input signals from the communication connections to be received. The second processing unit 22 may have a processor, memory, and input and output devices, including at least one modem and associated communication connections, as described above for the first processing unit 12. In an exemplary embodiment, each of the processing units 12, 22 may execute the WEBPHONE™ Internet telephony application available from NetSpeak Corporation, Boca Raton, FL, which is capable of performing the disclosed point-to-point Internet protocol and system 10, as described herein.

[0014] The first processing unit 12 and the second processing unit 22 are operatively connected to the Internet 24 by communication devices and software known in the art, such as an Internet Service Provider (ISP) or an Internet gateway. The processing units 12, 22 may be operatively interconnected through the Internet 24 to a connection server 26, and may also be operatively connected to a mail server 28 associated with the Internet 24.

[0015] The connection server 26 includes a processor 30, a timer 32 for generating time stamps, and a memory such as a database 34 for storing, for example, E-mail and Internet Protocol (IP) addresses of logged-in units. In an exemplary embodiment, the connection server 26 may be a SPARC 5 server or a SPARC 20 server, available from SUN MICROSYSTEMS, INC., Mountain View, CA, having a central processing unit (CPU) as processor 30, an operating system (OS) such as UNIX, for providing timing operations such as maintaining the timer 32, a hard drive or fixed drive, as well as dynamic random access memory (DRAM) for storing the database 34, and a keyboard and display and/or other input and output devices (not shown in FIG. 1). The database 34 may be an SQL database available from ORACLE or INFORMIX.

[0016] In an exemplary embodiment, the mail server 28 may be a Post Office Protocol (POP) Version 3 mail server including a processor, memory, and stored programs operating in a UNIX environment, or, alternatively, another OS, to process E-mail capabilities between processing units and devices over the Internet 24.

[0017] The first processing unit 12 may operate the disclosed point-to-point Internet protocol by a computer program described hereinbelow in conjunction with FIG. 6, which may be implemented from compiled and /or interpreted source code in the C++ programming language and which may be downloaded to the first processing unit 12 from an external computer. The operating computer program may be stored in the memory 16, which may include about 8 MB RAM and/or a hard or fixed drive having about 8 MB. Alternatively, the source code may be implemented in the first processing unit 12 as firmware, as an erasable read only memory (EPROM), etc. It is understood that one skilled in the art would be able to use programming languages other than C++ to implement the disclosed point-to-point network protocol and system 10.

[0018] The processor 14 receives input commands and data from a first user associated with the first processing unit 12 though the input device 18, which may be an input port connected by a wired, optical, or a wireless connection for electromagnetic transmissions, or alternatively may be transferable storage media, such as floppy disks, magnetic tapes, compact disks, or other storage media including the input data from the first user.

[0019] The input device 18 may include a user interface (not shown) having, for example, at least one button actuated by the user to input commands to select from a plurality of operating modes to operate the first processing unit 12. In alternative embodiments, the input device 18 may include a keyboard, a mouse, a touch screen, and/or a data reading device such as a disk drive for receiving the input data from input data files stored in storage media such as a floppy disk or, for example, an 8 mm storage tape. The input device 18 may alternatively include connections to other computer systems to receive the input commands and data therefrom.

[0020] The first processing unit 12 may include a visual interface for use in conjunction with the input device 18 and output device 20 similar to those screens illustrated in FIGS. 5-6, discussed below. It is also understood that alternative devices may be used to receive commands and data from the user, such as keyboards, mouse devices, and graphical user interfaces (GUI) such as WINDOWS™ 3.1 available form MICROSOFT Corporation, Redmond, WA., and other operating systems and GUIs, such as OS/2 and OS/2 WARP, available from IBM CORPORATION, Boca Raton, FL. Processing unit 12 may also include microphones and/or telephone handsets for receiving audio voice data and commands, speech or voice recognition devices, dual tone multifrequency (DTMF) based devices, and/or software known in the art to accept voice data and commands and to operate the first processing unit 12.

[0021] In addition, either of the first processing unit 12 and the second processing unit 22 may be implemented in a personal digital assistant (PDA) providing modem and E-mail capabilities and Internet access, with the PDA providing the input/output screens for mouse interactions or for touchscreen activation as shown, for example, in FIGS. 5-6, as a combination of the input device 18 and output device 20.

[0022] For clarity of explanation, the illustrative embodiment of the disclosed point-to-point Internet protocol and system 10 is presented as having individual functional blocks, which may include functional blocks labeled as "processor" and "processing unit". The functions represented by these blocks may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. For example, the functions of each of the processors and processing units presented herein may be provided by a shared processor or by a plurality of individual processors. Moreover, the use of the functional blocks with accompanying labels herein is not to be construed to refer exclusively to hardware capable of executing software. Illustrative embodiments may include digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, read-only memory (ROM) for storing software performing the operations discussed below, and random access memory (RAM) for storing DSP results. Very large scale integration (VLSI) hardware embodiments, as well as custom VLSI circuitry in combination with a general purpose DSP circuit, may also be provided. Any and all of these embodiments may be deemed to fall within the meaning of the labels for the functional blocks as used herein.

[0023] The processing units 12, 22 are capable of placing calls and connecting to other processing units connected to the Internet 24, for example, via dialup SLIP/PPP lines. In an exemplary embodiment, each processing unit assigns an unsigned long session number, for example, a 32- bit long sequence in a *.ini file for each call. Each call may be assigned a successive session number in sequence, which may be used by the respective processing unit to associate the call with one of the SLIP/PPP lines, to associate a <ConnectOK> response signal with a <Connect Request> signal, and to allow for multiplexing and demultiplexing of inbound and outbound conversations on conference lines, as explained hereinafter.

[0024] For callee (or called) processing units with fixed IP addresses, the caller (or calling) processing unit may open a "socket", i.e. a file handle or address indicating where data is to be sent, and transmit a <Call> command to establish communication with the callee utilizing, for example, datagram services such as Internet Standard network layering as well as transport layering, which may include a Transport Control Protocol (TCP) or a User Datagram Protocol (UDP) on top of the IP. Typically, a processing unit having a fixed IP address may maintain at least one open socket and a called processing unit waits for a <Call> command to assign the open socket to the incoming signal. If all lines are in use, the callee processing unit sends a BUSY signal or message to the callee processing unit. As shown in FIG. 1, the disclosed point-to-point Internet protocol and system 10 operate when a callee processing unit does not have a fixed or predetermined IP address. In the exemplary embodiment and without loss of generality, the first processing unit 12 is the caller processing unit and the second processing unit 22 is the called processing unit. When either of processing units 12, 22 logs on to the Internet via a dial-up connection, the respective unit is provided a dynamically allocated IP address by the a connection service provider.

[0025] Upon the first user initiating the point-to-point Internet protocol when the first user is logged on to the Internet 24, the first processing unit 12 automatically transmits its associated E-mail address and its dynamically allocated IP address to the connection server 26. The connection server 26 then stores these addresses in the database 34 and time stamps the stored addresses using timer 32. The first user operating the first processing unit 12 is thus established in the database 34 as an active on-line party available for communication using the disclosed point-to-point Internet protocol. Similarly, a second user operating the second processing unit 22, upon connection to the Internet 24 through the a connection service provider, is processed by the connection server 26 to be established in the database 34 as an active on-line party.

[0026] The connection server 26 may use the time stamps to update the status of each processing unit; for example, after 2 hours, so that the on-line status information stored in the database 34 is relatively current. Other predetermined time periods, such as a default value of 24 hours, may be configured by a systems operator.

[0027] The first user with the first processing unit 12 initiates a call using, for example, a Send command and/or a command to speeddial an N^{TH} stored number, which may be labeled [SND] and [SPD] [N], respectively, by the input device 18 and/or the output device 20, such as shown in FIGS. 5-6. In response to either the Send or speeddial commands, the first processing unit 12 retrieves from memory 16 a stored E-mail address of the callee corresponding to the N^{TH} stored number. Alternatively, the first user may directly enter the E-mail address of the callee.

[0028] The first processing unit 12 then sends a query, including the E-mail address of the callee, to the connection server 26. The connection server 26 then searches the database 34 to determine whether the callee is logged-in by finding any stored information corresponding to the callee's E-mail address indicating that the callee is active and on-line. If the callee is active and on-line, the connection server 26 then performs the primary point-to-point Internet protocol; i.e. the IP address of the callee is retrieved from the database 34 and sent to the first processing unit 12. The first processing unit 12 may then directly establish the point-to-point Internet communications with the callee using the IP address of the callee.

[0029] If the callee is not on-line when the connection server 26 determines the callee's status, the connection server 26 sends an OFFLINE signal or message to the first processing unit 12. The first processing unit 12 may also display a message such as "Called Party Off-Line" to the first user.

[0030] When a user logs off or goes off-line from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user's information, or by flagging the user as being off-line. The connection server 26 may be instructed to update the user's information in the database 34 by an off-line message, such as a data packet, sent automatically from the processing unit of the user prior to being disconnected from the connection server 26. Accordingly, an off-line user is effectively disabled from making and/or receiving point-to-point Internet communications.

[0031] As shown in FIGS. 2-4, the disclosed secondary point-to-point Internet protocol may be used as an alternative to the primary point-to-point Internet protocol described above, for example, if the connection server 26 is non-responsive, inoperative, and/or unable to perform the primary point-to-point Internet protocol, as a non-responsive condition. Alternatively, the disclosed secondary point-to-point Internet protocol may be used independent of the primary point-to-point Internet protocol. In the disclosed secondary point-to-point Internet protocol, the first processing unit 12 sends a <ConnectRequest> message via E-mail over the Internet 24 to the mail server 28. The E-mail including the <ConnectRequest> message may have, for example, the subject

[*wp#XXXXXXXX#nnn.nnn.nnn.#emailAddr]

where nnn.nnn.nnn.nnn. is the current (i.e. temporary or permanent) IP address of the first user, and XXXXXXXX is a session number, which may be unique and associated with the request of the first user to initiate point-to-point communication with the second user.

[0032] As described above, the first processing unit 12 may send the <ConnectRequest> message in response to an unsuccessful attempt to perform the primary point-to-point Internet protocol. Alternatively, the first processing unit 12 may send the <ConnectRequest> message in response to the first user initiating a SEND command or the like.

[0033] After the <ConnectRequest> message via E-mail is sent, the first processing unit 12 opens a socket and waits to detect a response from the second processing unit 22. A timeout timer, such as timer 32, may be set by the first processing unit 12, in a manner known in the art, to wait for a predetermined duration to receive a <ConnectOK> signal. The processor 14 of the first processing unit 12 may cause the output device 20 to output a Ring signal to the user, such as an audible ringing sound, about every 3 seconds . For example, the processor 14 may output a *.wav file, which may be labeled RING.WAV, which is processed by the output device 20 to output an audible ringing sound.

[0034] The mail server 28 then polls the second processing unit 22, for example, every 3-5 seconds, to deliver the E-mail. Generally, the second processing unit 22 checks the incoming lines, for example, at regular intervals to wait for and to detect incoming E-mail from the mail server 28 through the Internet 24.

[0035] Typically, for sending E-mail to users having associated processing units operatively connected to a host computer or server operating an Internet gateway, E-mail for a specific user may be sent over the Internet 24 and directed to the permanent IP address or the SLIP/PPP account designation of the host computer, which then assigns a temporary IP address to the processing unit of the specified user for properly routing the E-mail. The E-mail signal may include a name or other designation such as a user name which identifies the specific user regardless of the processing unit assigned to the user; that is, the host computer may track and store the specific device where a specific user is assigned or logged on, independent of the IP address system, and so the host computer may switch the E-mail signal to the device of the specific user. At that time, a temporary IP address may be generated or assigned to the specific user and device.

[0036] Upon detecting and/or receiving the incoming E-mail signal from the first processing unit 12, the second processing unit 22 may assign or may be assigned a temporary IP address. Therefore, the delivery of the E-mail through the Internet 24 provides the second processing unit 22 with a session number as well as IP addresses of both the first processing unit 12 and the second processing unit 22.

[0037] Point-to-point communication may then be established by the processing unit 22 processing the E-mail signal to extract the <ConnectRequest> message, including the IP address of the first processing unit 12 and the session number. The second processing unit 22 may then open a socket and generate a <ConnectOK> response signal, which includes the temporary IP address of the second processing unit 22 as well as the session number of the first processing unit.

[0038] The second processing unit 22 sends the <ConnectOK> signal directly over the Internet 24 to the IP address of the first processing unit 12 without processing by the mail server 28, and a timeout timer of the second processing unit 22 may be set to wait and detect a <Call> signal expected from the first processing unit 12.

[0039] Realtime point-to-point communication of audio signals over the Internet 24, as well as video and voicemail, may thus be established and supported without requiring permanent IP addresses to be assigned to either of the users or processing units 12, 22. For the duration of the realtime point-to-point link, the relative permanence of the current IP addresses of the processing units 12, 22 is sufficient, whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).

[0040] In the exemplary embodiment, a first user operating the first processing unit 12 is not required to be notified by the first processing unit 12 that an E-mail is being generated and sent to establish the point-to-point link with the second user at the second processing unit 22. Similarly, the second user is not required to be notified by the second processing unit 22 that an E-mail has been received and/or a temporary IP address is associated with the second processing unit 22. The processing units 12, 22 may perform the disclosed point-to-point Internet protocol automatically upon initiation of the point-to-point communication command by the first user without displaying the E-mail interactions to either user. Accordingly, the disclosed point-to-point Internet protocol may be transparent to the users. Alternatively, either of the first and second users may receive, for example, a brief message of "CONNECTION IN PROGRESS" or the like on a display of the respective output device of the processing units 12, 22.

[0041] After the initiation of either the primary or the secondary point-to-point Internet protocols described above in conjunction with FIGS. 1-2, the point-to-point communication link over the Internet 24 may be established as shown in FIGS. 3-4 in a manner known in the art. For example, referring to FIG. 3, upon receiving the <ConnectorOK> signal from the second processing unit 22, the first processing unit 12 extracts the IP address of the second processing unit 22 and the session number, and the session number sent from the second processing unit 22 is then checked with the session number originally sent from the first processing unit 12 in the <ConnectRequest> message as E-mail. If the session numbers sent and received by the processing unit 12 match, then the first processing unit 12 sends a <Call> signal directly over the Internet 24 to the second processing unit 22; i.e. using the IP address of the second processing unit 22 provided to the first processing unit 12 in the <ConnectOK> signal.

[0042] Upon receiving the <Call> signal, the second processing unit 22 may then begin a ring sequence, for example, by indicating or annunciating to the second user that an incoming call is being received. For example, the word "CALL" may be displayed on the output device of the second processing unit 22. The second user may then activate the second processing unit 22 to receive the incoming call.

[0043] Referring to FIG. 4, after the second processing unit 22 receives the incoming call, realtime audio and/or video conversations may be conducted in a manner known in the art between the first and second users through the Internet 24, for example, by compressed digital audio signals. Each of the processing units 12, 22 also display to each respective user the words "IN USE" to indicate that the point-to-point communication link is established and audio or video signals are being transmitted.

[0044] In addition, either user may terminate the point-to-point communication link by, for example, activating a termination command, such as by activating an [END] button or icon on a respective processing unit, causing the respective processing unit to send an <End> signal which causes both processing units to terminate the respective sockets, as well as to perform other cleanup commands and functions known in the art.

[0045] FIGS. 5-8 illustrate examples of display screens 36 which may be output by a respective output device of each processing unit 12, 22 of FIGS. 1-4 for providing the disclosed point-to-point Internet protocol and system 10. Such display screens may be displayed on a display of a personal computer (PC) or a PDA in a manner known in the art.

[0046] As shown in FIG. 5, a first display screen 36 includes a status area 38 for indicating, for example, a called user by name and/or by IP address or telephone number; a current function such as C2; a current time; a current operating status such as "IN USE", and other control icons such as a down arrow icon 40 for scrolling down a list of parties on a current conference line. The operating status may include such annunciators as "IN USE," "IDLE," "BUSY," "NO ANSWER," "OFFLINE," "CALL," "DIALING," "MESSAGES," and "SPEEDDIAL."

[0047] Other areas of the display screen 36 may include activation areas or icons for actuating commands or entering data. For example, the display screen 36 may include a set of icons 42 arranged in columns and rows including digits 0-9 and commands such as END, SND, HLD, etc. For example, the END and SND commands may be initiated as described above, and the HLD icon 44 may be actuated to place a current line on hold. Such icons may also be configured to substantially simulate a telephone handset or a cellular telephone interface to facilitate ease of use, as well as to simulate function keys of a keyboard. For example, icons labeled L1-L4 may be mapped to function keys F1-F4 on standard PC keyboards, and icons C1-C3 may be mapped to perform as combinations of function keys, such as CTRL-F1, CTRL-F2, and CTRL-F3, respectively. In addition, the icons labeled L1-L4 and C1-C3 may include circular regions which may simulate light emitting diodes (LEDs) which indicate that the function or element represented by the respective icon is active or being performed.

[0048] Icons L1-L4 may represent each of 4 lines available to the caller, and icons C1-C3 may represent conference calls using at least one line to connect, for example, two or more parties in a conference call. The icons L1-L4 and C1-C3 may indicate the activity of each respective line or conference line. For example, as illustrated in FIG. 5, icons L1-L2 may have lightly shaded or colored circles, such as a green circle, indicating that each of lines 1 and 2 are in use, while icons L3-L4 may have darkly shaded or color circles, such as a red or black circle, indicating that each of lines 3 and 4 are not in use. Similarly, the lightly shaded circle of the icon labeled C2 indicates that the function corresponding to C2 is active, as additionally indicated in the status are 38, while darkly shaded circles of icons labeled C1 and C3 indicate that such corresponding functions are not active.

[0049] The icons 42 are used in conjunction with the status area 38. For example, using a mouse for input, a line that is in use, as indicated by the lightly colored circle of the icon, may be activated to indicate a party's name by clicking a right mouse button for 5 seconds until another mouse click is actuated or the [ESC] key or icon is actuated. Thus, the user may switch between multiple calls in progress on respective lines.

[0050] Using the icons as well as an input device such as a mouse, a user may enter the name or alias or IP address, if known, of a party to be called by either manually entering the name, by using the speeddial feature, or by double clicking on an entry in a directory stored in the memory, such as the memory 16 of the first processing unit 12, where the directory entries may be scrolled using the status area 38 and the down arrow icon 40.

[0051] Once a called party is listed in the status area 38 as being active on a line, the user may transfer the called party to another line or a conference line by clicking and dragging the status area 38, which is represented by a reduced icon 46. Dragging the reduced icon 46 to any one of line icons L1-L4 transfers the called party in use to the selected line, and dragging the reduced icon 46 to any one of conference line icons C1-C3 adds the called party to the selected conference call.

[0052] Other features may be supported, such as icons 48-52, where icon 48 corresponds to, for example, an ALT-X command to exit the communication facility of a processing unit, and icon 50 corresponds to, for example, an ALT-M command to minimize or maximize the display screen 36 by the output device of the processing unit. Icon 52 corresponds to an OPEN command, which may, for example, correspond to pressing the O key on a keyboard, to expand or contract the display screen 36 to represent the opening and closing of a cellular telephone. An "opened" configuration is shown in FIG. 5, and a "closed" configuration is shown in FIG. 6. In the "opened" configuration, additional features such as output volume (VOL) controls, input microphone (MIC) controls, waveform (WAV) sound controls, etc.

[0053] The use of display screens such as those shown in FIGS. 5-6 provided flexibility in implementing various features available to the user. It is to be understood that additional features such as those known in the art may be supported by the processing units 12, 22.

[0054] Alternatively, it is to be understood that one skilled in the art may implement the processing units 12, 22 to have the features of the display screens in FIGS. 5-6 in hardware; i.e. a wired telephone or wireless cellular telephone may include various keys, LEDs, liquid crystal displays (LCDs), and touchscreen actuators corresponding to the icons and features shown in FIGS. 5-6. In addition, a PC may have the keys of a keyboard and mouse mapped to the icons and features shown in FIGS. 5-6.

[0055] Referring to FIG. 7, the disclosed point-to-point Internet protocol and system 10 is illustrated. First processing unit 12 initiates the point-to-point Internet protocol in step 56 by sending a query from the first processing unit 12 to the connection server 26. If connection server 26 is operative to perform the point-to-point Internet protocol, in step 58, first processing unit 12 receives an on-line status signal from the connection server 26, such signal may include the IP address of the callee or a "Callee Off-Line" message. Next, first processing unit 12 performs the primary point-to-point Internet protocol in step 60, which may include receiving, at the first processing unit 12, the IP address of the callee if the callee is active and on-line. Alternatively, processing unit 60 may initiate and perform the secondary point-to-point Internet protocol in step 62, if the called party is not active and/or on-line.

[0056] Referring to FIG. 8, in conjunction with FIGS. 1 and 3-4, the disclosed point-to-point Internet protocol and system 10 is illustrated. Connection server 26 starts the point-to-point Internet protocol, in step 64, and timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66. Connection server 26 receives a query from a first processing unit 12 in step 68 to determine whether a second user or second processing unit 22 is logged-in to the Internet 24, with the second user being specified, for example, by an E-mail address. Connection server 26 retrieves the IP address of the specified user from the database 34 in step 70, if the specified user is logged-in to the Internet, and sends the retrieved IP address to the first processing unit 12 in step 72 to enable first processing unit 12 to establish point-to-point communications with the specified second user.

[0057] The disclosed secondary point-to-point Internet protocol operates as shown in FIG. 9. First processing unit 12 generates an E-mail signal, including a session number and a first IP address corresponding to a first processing unit in step 76. First processing unit 12 transmits the E-mail signal as a <ConnectRequest> signal to the Internet 24 in step 78. The E-mail signal is delivered through the Internet 24 using a mail server 28 to the second processing unit 22 in step 80. Second processing unit 22 extracts the session number and the first IP address from the E-mail signal in step 82 and transmits or sends the session number and a second IP address corresponding to the second processing unit 22, back to the first processing unit 12 through the Internet 24, in step 84. First processing unit 12 verifies the session number received from the second processing unit 22 in step 86, and establishes a point-to-point Internet communication link between the first processing unit 12 and second processing unit 22 using the first and second IP addresses in step 88.

[0058] While the disclosed point-to-point Internet protocols and system have been particularly shown and described with reference to the preferred embodiments, it is understood by those skilled in the art that various modifications in form and detail may be made therein without departing from the scope and spirit of the invention. Accordingly, modifications such as those suggested above, but not limited thereto, are to be considered within the scope of the invention.

Claims

1. In a computer system (12) having a display (36) and audio transducer, the computer system coupled to other computer systems and a server (26) over a computer network (24), an apparatus for establishing a point-to-point communication link comprising:

   a. means for transmitting, from the computer system (12) to the server (26) a query as to whether a second computer system (22) is connected to the computer network (24);
   b. means for receiving a network protocol address of the of the second computer from the server when the second computer system is connected to the computer network; and
   c. means, responsive to the network protocol address of the second computer system, for establishing a point-to-point communication link between the first computer system and the second computer system over the computer network.

2. The apparatus of claim 1 further comprising:

   d. means for receiving audio data and transmitting the audio data to the second computer over the established point-to-point communication link.

3. The apparatus claim 1, wherein the network protocol addresses comprise Internet Protocol Addresses (IP addresses).

4. The apparatus claim 1 wherein the query transmitted to the server includes an E-mail address of the second computer system (22).

5. The apparatus claim 1, further comprising:

   f. means for transmitting an E-mail signal containing a network protocol address from the computer system (12) to a second computer system over the computer network (24) when the server (26) indicates that the second computer system is not connected to the computer network;
   g. means for receiving a second network protocol address from the second computer system over the computer network; and
   h. means, responsive to the second network protocol address, for establishing a point-to-point communication link between the first computer system (12) and the second computer system (22) over the computer network.

6. A method of operating a processing unit (12) for establishing a point-to-point communication between the processing unit (12) and a second one of a plurality of other processing units (22) over a computer network (24) including a server (26), each of said processing units having a display and an audio transducer, the method comprising the steps of:

   a. transmitting from the processing unit (12) to the server a query as to whether the second processing (22) unit is connected to the computer network (24);
   b. receiving a network protocol address of the second processing unit from the server when the second processing unit is connected to the computer network; and
   c. establishing in response to the network pro-

tocol address of the second processing unit a point-to-point communication link between the first processing unit and the second processing unit over the computer network.

7. The method of claim 6 further comprising the step:

d. receiving audio data and transmitting the audio data to the second processing unit over the established point-to-point communication link.

8. The method of claim 6, wherein the network protocol addresses comprise Internet Protocol Addresses (IP addresses).

9. The method of claim 6 wherein the query transmitted to the server includes the E-mail address of the second computer system (22).

10. The method of claim 4, further comprising the steps of:

e. transmitting an E-mail signal containing a network protocol address from the first processing unit to the second processing unit over the computer network when the server indicates that the second computer system is not connected to the computer network;
f. receiving a second network protocol address from the second processing unit over the computer network; and
g. establishing in response to the second network protocol address a point-to-point communication link between the first processing unit and the second processing unit over the computer network.

11. A computer program product for use in a processing unit (12) having a memory (16), a display (36) and an audio transducer, to establish a point-to-point communication between the processing unit (12) and a second one of a plurality of other processing units (22) over a computer network (24) including a server (26), the computer program product having a computer usable medium containing computer readable program code, comprising:

a. program code for transmitting from the processing unit to the server a query as to whether the second processing unit is connected to the computer network;
b. program code for receiving a network protocol address of the second processing unit from the server when the second processing unit is connected to the computer network; and
c. program code for establishing in response to the network protocol address of the second processing unit a point-to-point communication link between the first processing unit and the second processing unit over the computer network.

12. The program product of claim 11 further comprising:

d. program code for receiving audio data and transmitting the audio data to the second processing unit over the established point-to-point communication link.

13. The program product of claim 11, wherein the network protocol addresses comprise Internet Protocol Addresses (IP addresses).

14. The program product of claim 11, wherein the query transmitted to the server includes the E-mail address of the second processing unit (22).

15. The program product of claim 11, further comprising:

e. program code for transmitting an E-mail signal containing a network protocol address from the first processing unit to the second processing unit over the computer network when the server indicates that the second computer system is not connected to the computer network;
f. program code for receiving a second network protocol address from the second processing unit over the computer network; and
g. program code for establishing in response to the second network protocol address a point-to-point communication link between the first processing unit and the second processing unit over the computer network.

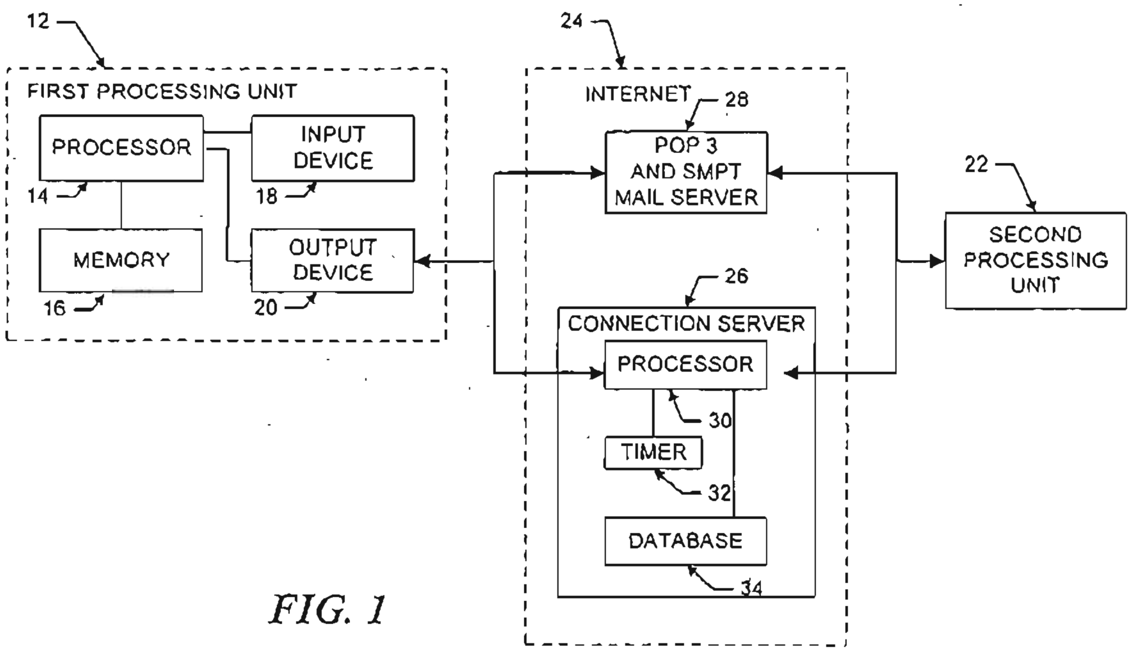

12
FIRST PROCESSING UNIT
PROCESSOR
14
INPUT DEVICE
18
MEMORY
16
OUTPUT DEVICE
20
24
INTERNET
28
POP 3 AND SMPT MAIL SERVER
26
CONNECTION SERVER
PROCESSOR
30
TIMER
32
DATABASE
34
22
SECOND PROCESSING UNIT

FIG. 1

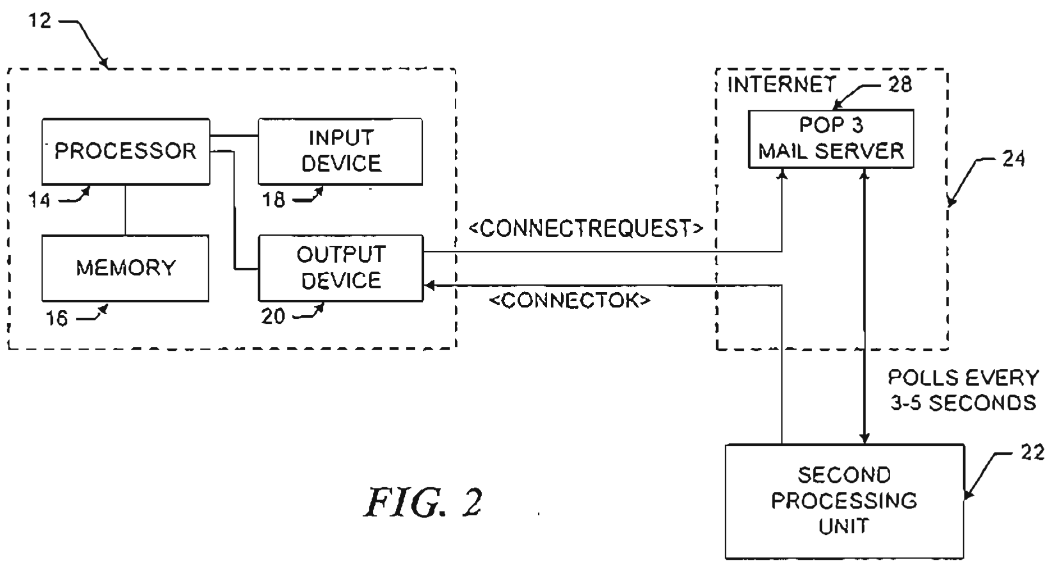

12
PROCESSOR
14
INPUT DEVICE
18
MEMORY
16
OUTPUT DEVICE
20
INTERNET
28
POP 3 MAIL SERVER
24
<CONNECTREQUEST>
<CONNECTOK>
POLLS EVERY 3-5 SECONDS
SECOND PROCESSING UNIT
22

FIG. 2

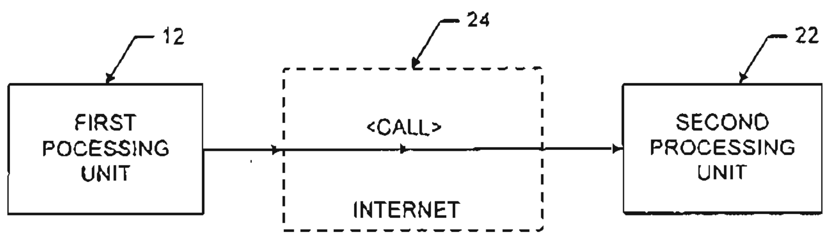

12
24
22
FIRST
POCESSING
UNIT
<CALL>
INTERNET
SECOND
PROCESSING
UNIT

FIG. 3

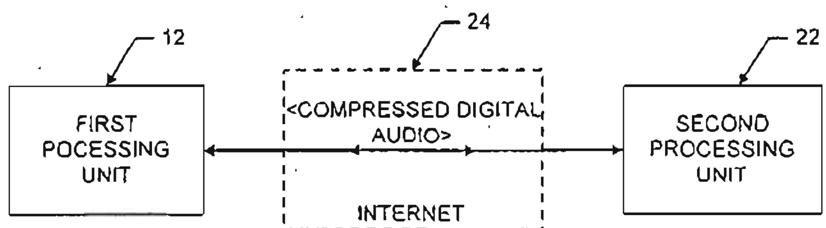

12
24
22
FIRST
POCESSING
UNIT
<COMPRESSED DIGITAL
AUDIO>
INTERNET
SECOND
PROCESSING
UNIT

FIG. 4

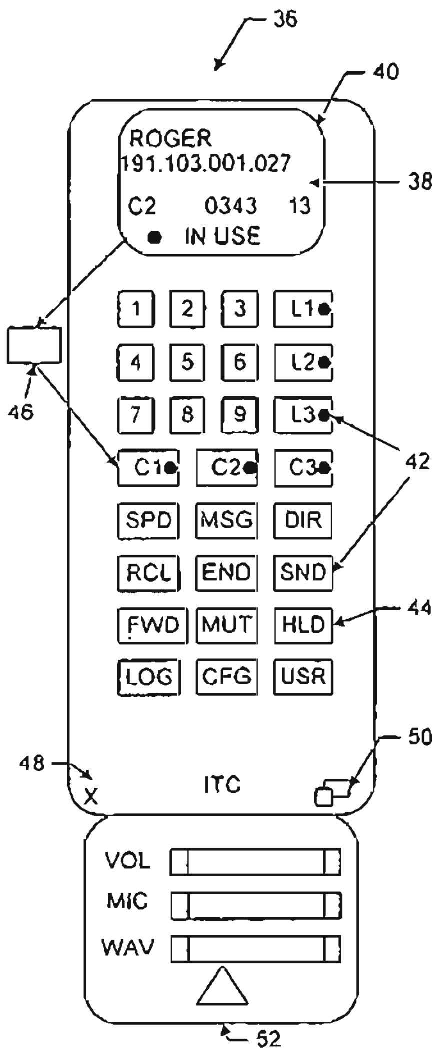

36
40
ROGER
191.103.001.027
C2 0343 13
IN USE
38
46
1
2
3
L1
4
5
6
L2
7
8
9
L3
C1
C2
C3
42
SPD
MSG
DIR
RCL
END
SND
FWD
MUT
HLD
44
LOG
CFG
USR
48
X
ITC
50
VOL
MIC
WAV
52

FIG. 5

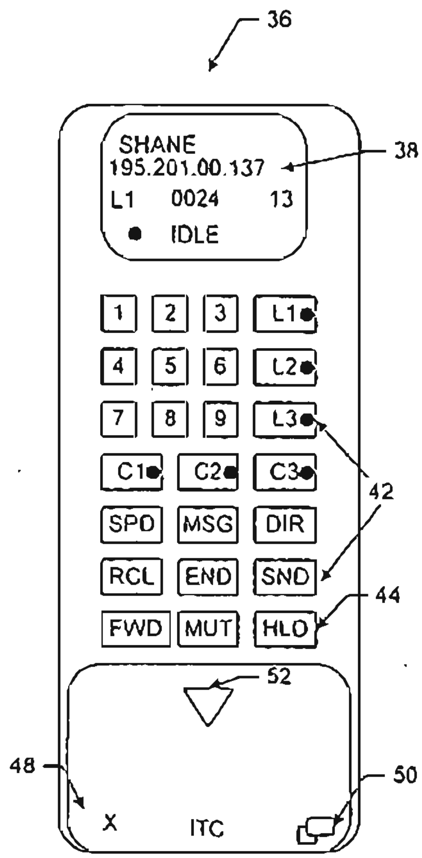

36
SHANE
195.201.00.137
L1 0024 13
IDLE
38
1
2
3
L1
4
5
6
L2
7
8
9
L3
C1
C2
C3
42
SPD
MSG
DIR
RCL
END
SND
FWD
MUT
HLD
44
52
48
X
ITC
50

FIG. 6

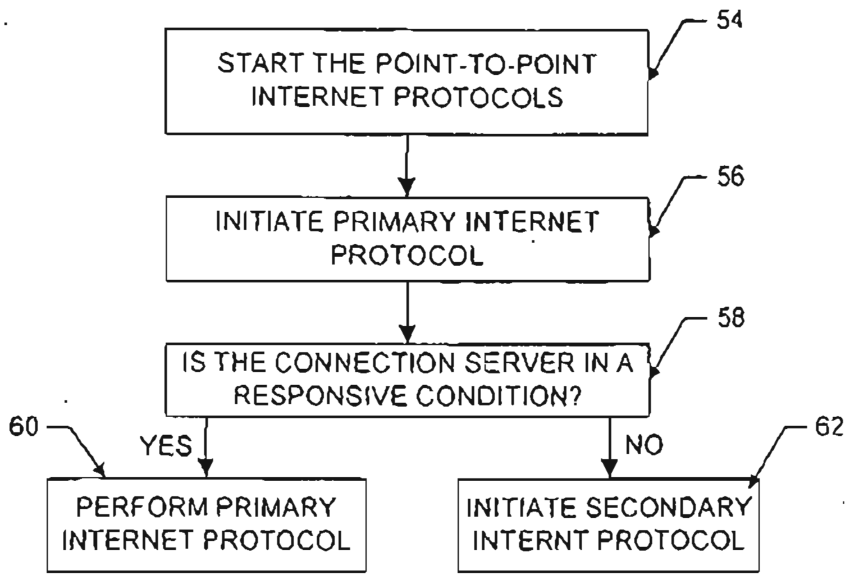

54
START THE POINT-TO-POINT INTERNET PROTOCOLS
56
INITIATE PRIMARY INTERNET PROTOCOL
58
IS THE CONNECTION SERVER IN A RESPONSIVE CONDITION?
60
YES
NO
62
PERFORM PRIMARY INTERNET PROTOCOL
INITIATE SECONDARY INTERNT PROTOCOL

FIG. 7

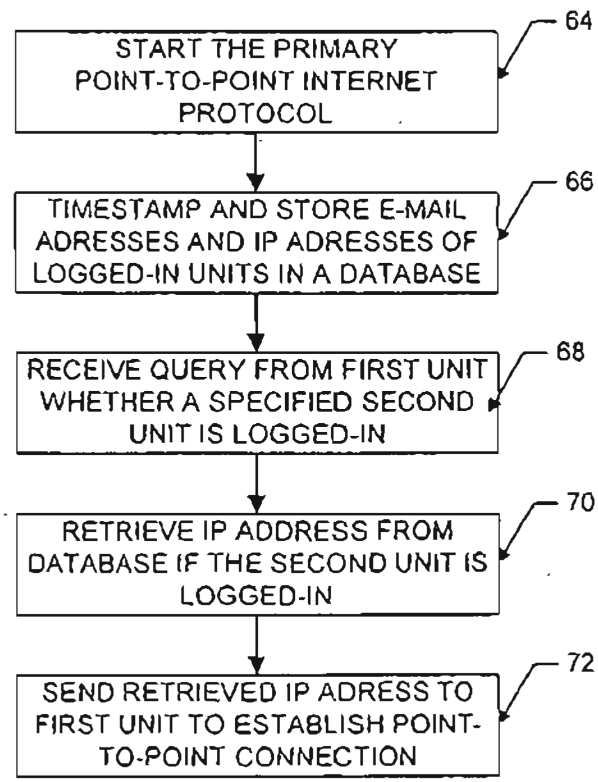

64
START THE PRIMARY POINT-TO-POINT INTERNET PROTOCOL
66
TIMESTAMP AND STORE E-MAIL ADRESSES AND IP ADRESSES OF LOGGED-IN UNITS IN A DATABASE
68
RECEIVE QUERY FROM FIRST UNIT WHETHER A SPECIFIED SECOND UNIT IS LOGGED-IN
70
RETRIEVE IP ADDRESS FROM DATABASE IF THE SECOND UNIT IS LOGGED-IN
72
SEND RETRIEVED IP ADRESS TO FIRST UNIT TO ESTABLISH POINT-TO-POINT CONNECTION

FIG. 8

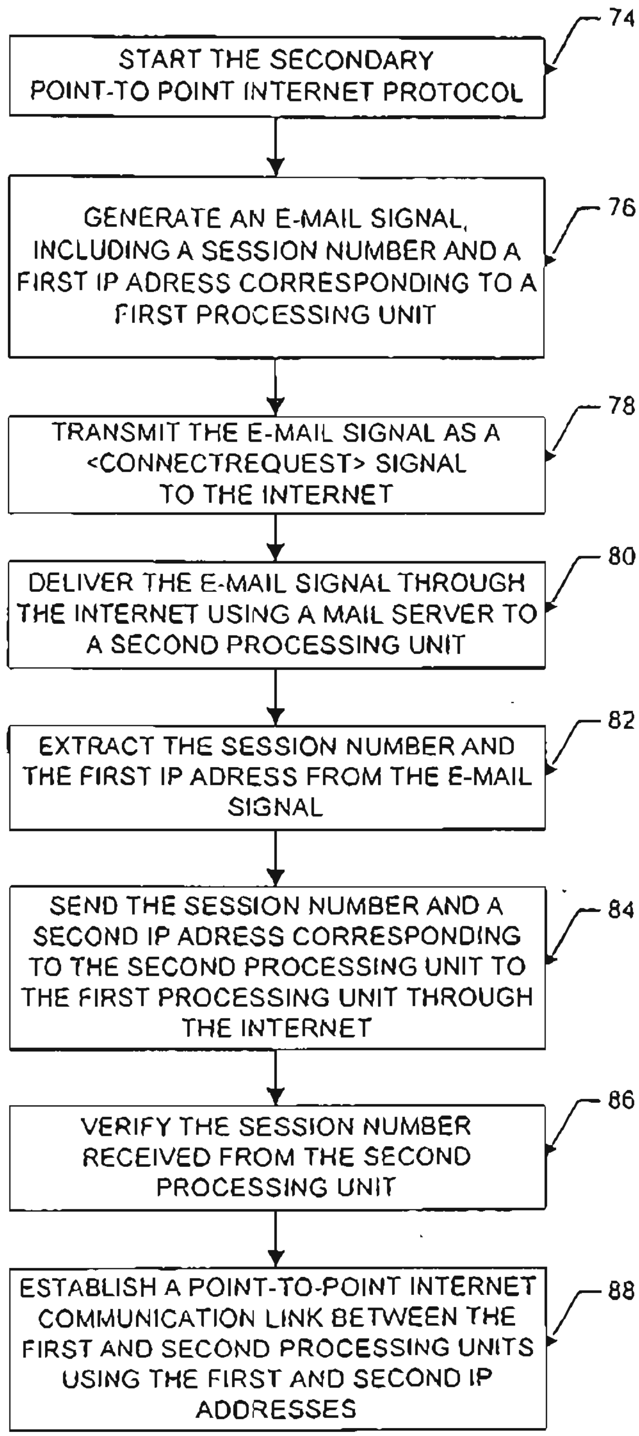

74
START THE SECONDARY POINT-TO POINT INTERNET PROTOCOL
76
GENERATE AN E-MAIL SIGNAL, INCLUDING A SESSION NUMBER AND A FIRST IP ADRESS CORRESPONDING TO A FIRST PROCESSING UNIT
78
TRANSMIT THE E-MAIL SIGNAL AS A <CONNECTREQUEST> SIGNAL TO THE INTERNET
80
DELIVER THE E-MAIL SIGNAL THROUGH THE INTERNET USING A MAIL SERVER TO A SECOND PROCESSING UNIT
82
EXTRACT THE SESSION NUMBER AND THE FIRST IP ADRESS FROM THE E-MAIL SIGNAL
84
SEND THE SESSION NUMBER AND A SECOND IP ADRESS CORRESPONDING TO THE SECOND PROCESSING UNIT TO THE FIRST PROCESSING UNIT THROUGH THE INTERNET
86
VERIFY THE SESSION NUMBER RECEIVED FROM THE SECOND PROCESSING UNIT
88
ESTABLISH A POINT-TO-POINT INTERNET COMMUNICATION LINK BETWEEN THE FIRST AND SECOND PROCESSING UNITS USING THE FIRST AND SECOND IP ADDRESSES

FIG. 9

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 379 050 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
10.03.2004 Bulletin 2004/11

(43) Date of publication A2:
07.01.2004 Bulletin 2004/02

(21) Application number: 03022288.9

(22) Date of filing: 25.09.1996

(51) Int Cl.7: H04L 29/06, H04L 12/58

(84) Designated Contracting States:
CH DE ES FI FR GB IE IT LI NL SE

(30) Priority: 25.09.1995 US 533115

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
96933928.2 / 0 852 868

(71) Applicant: Netspeak Corporation
Boca Raton, FL 33487 (US)

(72) Inventor: Hutton, Glenn W.
Miami, FL 33196 (US)

(74) Representative: Kindermann, Manfred
Patentanwalt,
Sperberweg 29
71032 Böblingen (DE)

(54) Server mediated point-to-point communication over a computer network

(57) In a computer system (12) having an audio transducer and a display device and being operatively coupled to other computers (22) and a server (26) over a computer network (24), means are described for establishing a point-to-point communication link between computer systems. The means provide for transmitting from a first process to the server a query as to whether a second process is connected to the computer network and for receiving a network protocol address of the second process from the server when the second process is connected to the computer network. In response to the received network protocol of the second process a point-to-point communication link is established between the first process and the second process over the computer network.

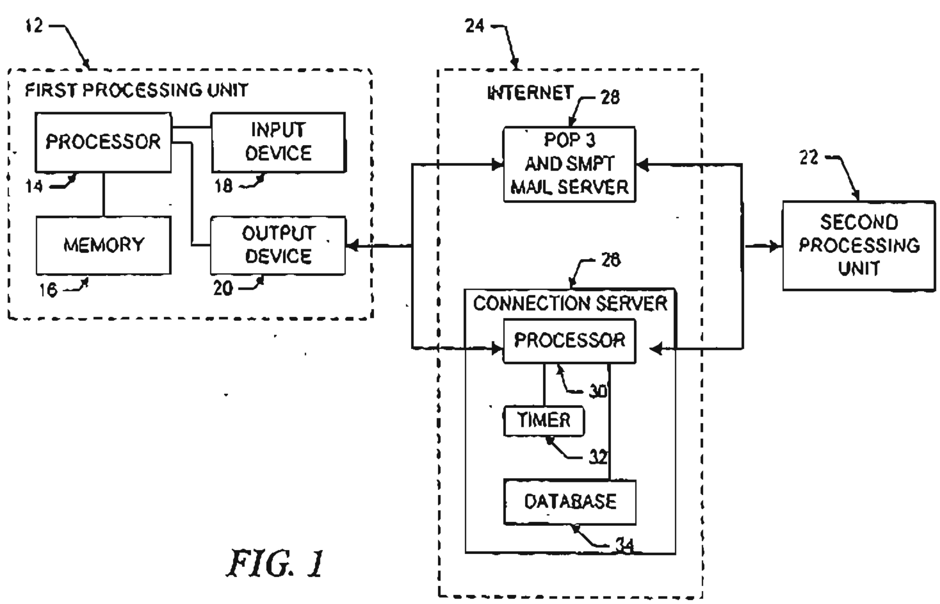


FIG. 1

EP 1 379 050 A3

Printed by Jouve, 75001 PARIS (FR)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 03 02 2288

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 5 440 547 A (ESAKI HIROSHI ET AL) 8 August 1995 (1995-08-08) * column 1, line 9 - column 4, line 43 * * column 16, line 9 - column 17, line 6 * ---	1-15	H04L29/06 H04L12/58
Y	GB 2 283 645 A (DIGITAL EQUIPMENT INT) 10 May 1995 (1995-05-10) * page 2, line 26 - page 3, line 2 * * page 11, line 17 - line 23 * ---	1-15	
Y	WO 94 22087 A (3COM CORP) 29 September 1994 (1994-09-29) * page 1, line 2 - page 6, line 8 * * page 12, line 17 - page 13, line 30 * * page 14, line 6 - line 10 * * page 15, line 12 - page 16, line 7 * * page 17, line 17 - line 22 * ---	1-15	
A	US 5 315 705 A (IWAMI NAOKO ET AL) 24 May 1994 (1994-05-24) * abstract * * column 1, line 44 - line 62 * ---	1,6,11	TECHNICAL FIELDS SEARCHED (Int.Cl.7)
A	US 5 287 103 A (KASPRZYK MARLON Z ET AL) 15 February 1994 (1994-02-15) * column 4, line 18 - column 6, line 60 * ---	1,6,11	H04L
A	EP 0 648 038 A (IBM) 12 April 1995 (1995-04-12) * column 7, line 33 - line 52 * ---	1,6,11	
A	EP 0 518 596 A (DIGITAL EQUIPMENT CORP) 16 December 1992 (1992-12-16) * column 1, line 31 - line 36 * * column 3, line 12 - line 30 * --- -/--	1,6,11	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	20 January 2004	Vaskimo, K

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 03 02 2288

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	EP 0 597 691 A (IBM) 18 May 1994 (1994-05-18) * page 6, line 10 - line 27 * ---		
A	EP 0 632 672 A (IBM) 4 January 1995 (1995-01-04) * column 4, line 17 - line 24 * * column 6, line 24 - line 37 * * figure 1 * -----		
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	20 January 2004	Vaskimo, K

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 03 02 2288

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

20-01-2004

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5440547	A	08-08-1995	JP	6318951 A	15-11-1994
			JP	6261094 A	16-09-1994
GB 2283645	A	10-05-1995	NONE		
WO 9422087	A	29-09-1994	US	5526489 A	11-06-1996
			AU	672097 B2	19-09-1996
			AU	6017894 A	11-10-1994
			CA	2157193 A1	29-09-1994
			EP	0689696 A1	03-01-1996
			JP	8507911 T	20-08-1996
			WO	9422087 A1	29-09-1994
US 5315705	A	24-05-1994	JP	4269049 A	25-09-1992
US 5287103	A	15-02-1994	NONE		
EP 0648038	A	12-04-1995	GB	2281793 A	15-03-1995
			EP	0648038 A2	12-04-1995
			JP	2505116 B2	05-06-1996
			JP	7093238 A	07-04-1995
			US	6282569 B1	28-08-2001
EP 0518596	A	16-12-1992	US	5420862 A	30-05-1995
			DE	69216915 D1	06-03-1997
			DE	69216915 T2	11-09-1997
			EP	0518596 A1	16-12-1992
			JP	5207022 A	13-08-1993
EP 0597691	A	18-05-1994	CA	2103766 A1	13-05-1994
			EP	0597691 A1	18-05-1994
			JP	2566730 B2	25-12-1996
			JP	7212363 A	11-08-1995
EP 0632672	A	04-01-1995	US	5694548 A	02-12-1997
			EP	0632672 A2	04-01-1995
			JP	2868178 B2	10-03-1999
			JP	7058809 A	03-03-1995

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

(12) UK Patent Application (19) GB (11) 2 283 645 (13) A

(43) Date of A Publication 10.05.1995

(21) Application No 9322910.2

(22) Date of Filing 06.11.1993

(71) Applicant(s)
Digital Equipment International Limited

(Incorporated in Switzerland)

1 Grand Places, 1700 Fribourg, Switzerland

(72) Inventor(s)
Stewart F Bryant
Ian Michael Charles Shand

(74) Agent and/or Address for Service
Eric Potter Clarkson
St Mary's Court, St Mary's Gate, NOTTINGHAM, NG1 1LE, United Kingdom

(51) INT CL[6]
H04L 12/46 12/66

(52) UK CL (Edition N)
H4P PPA

(56) Documents Cited
GB 2267418 A

(58) Field of Search
UK CL (Edition M) H4P PPA PPG
INT CL[5] H04L 12/46 12/66
Online databases:WPI,INSPEC

(54) Digital communication systems

(57) A digital communication system comprising a network of routers R1-R3 linked together by links LK12-LK23 and having LANs LAN1-LAN7 coupled to them, and using IP (Internet Protocol), under which each LAN has a subnet address, and each host on a LAN has the subnet address as the high-order part of its own address. In IP, each router contains a set of interface/LAN tables each listing the low-order address portions of the addresses of the hosts attached to the LAN plus the MAC (medium access control) identifiers of those hosts, and a set of link tables listing the subnet addresses of the LANs reachable through those links. In the present system, both the interface tables and the link tables contain the full host addresses of all hosts reachable through those interfaces and links, and the routers also contain means for polling the interfaces for unknown hosts. Each router also contain an ARP (address resolution protocol) unit (30, Fig. 2) for detecting ARP requests from a source for a destination having the same subnet address as the source but not on the same Interface, and returning a proxy ARP response giving the router's identification. A host can thereby be moved to a LAN whose address does not match that of the host.

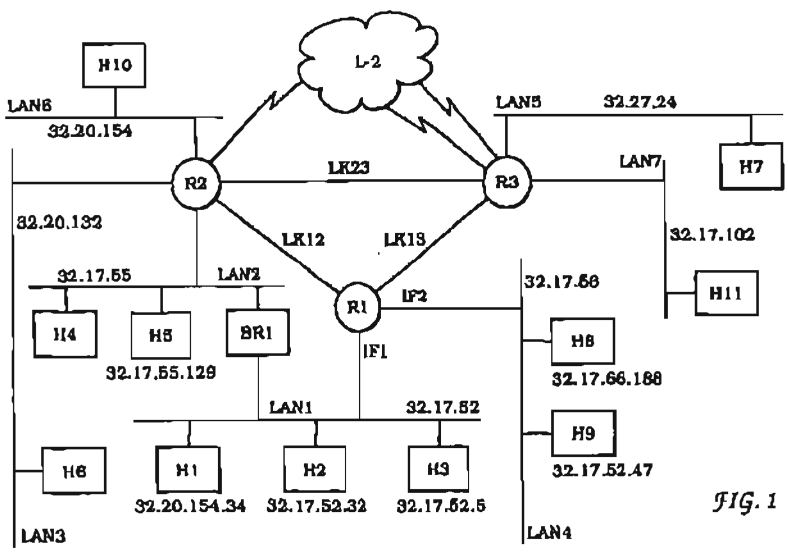
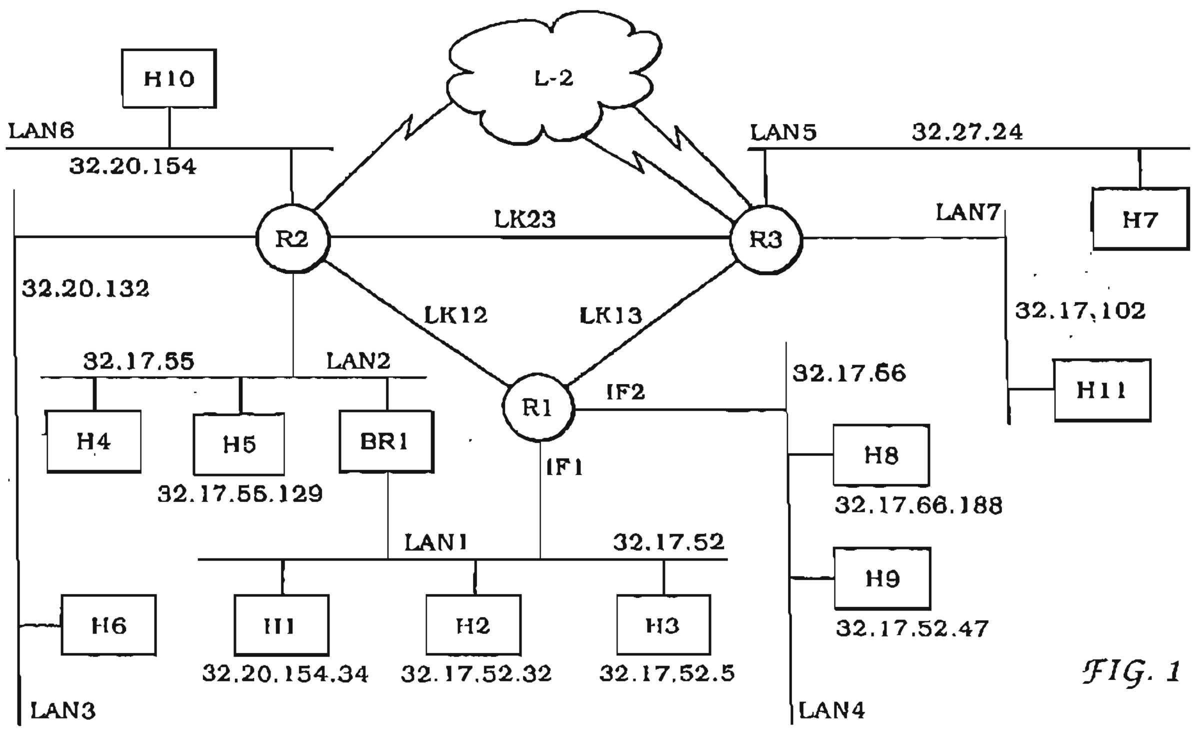


FIG. 1

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

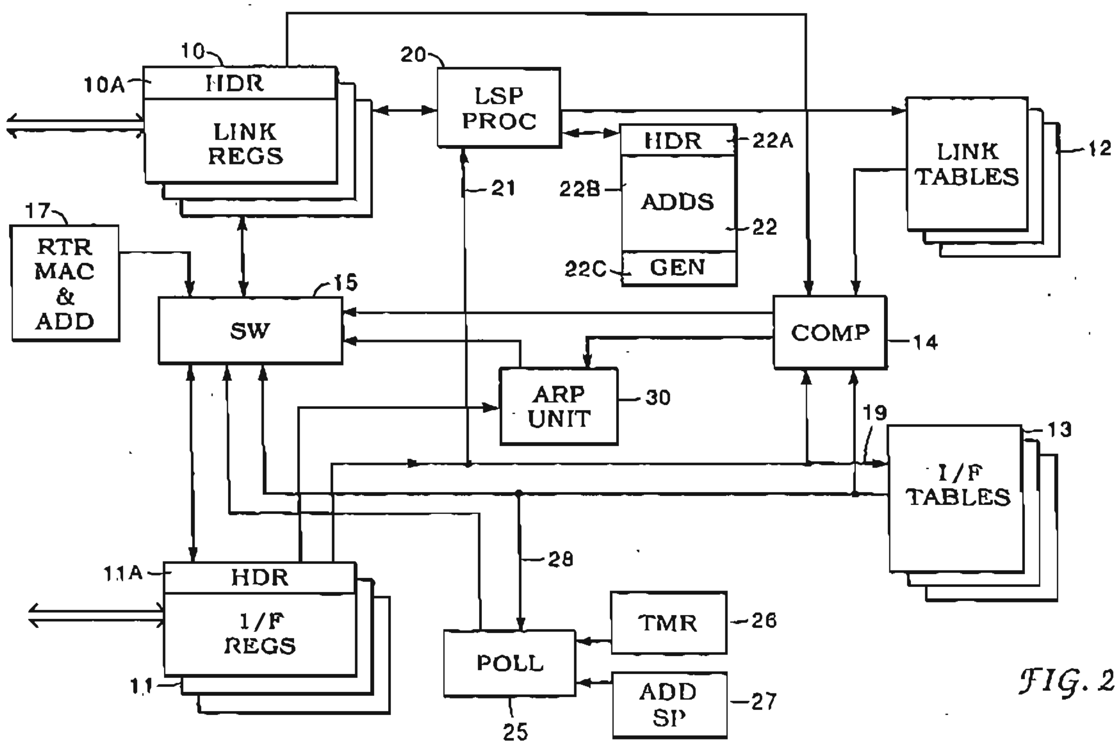
GB 2 283 645 A



H10
L-2
LAN6
32.20.154
LAN5
32.27.24
LK23
R2
R3
LAN7
H7
32.20.132
LK12
LK13
32.17.102
32.17.55
LAN2
32.17.66
R1
IF2
H11
H4
H5
BR1
H8
IF1
32.17.55.129
32.17.66.188
LAN1
32.17.52
H9
H6
H1
H2
H3
32.17.52.47
32.20.154.34
32.17.52.32
32.17.52.5
LAN3
LAN4

FIG. 1

1/2



10
10A
HDR
LINK
REGS
20
LSP
PROC
HDR
22A
ADDS
22B
22
GEN
22C
21
LINK
TABLES
12
17
RTR
MAC
&
ADD
SW
15
COMP
14
ARP
UNIT
30
19
I/F
TABLES
13
11A
HDR
I/F
REGS
11
28
POLL
25
TMR
26
ADD
SP
27

FIG. 2

2/2

Digital Communication Systems

The present invention relates to digital communication systems, and more particularly to the addressing of units therein.

Digital communication systems: general

There is a considerable variety of digital communication systems. We are primarily concerned here with systems which interconnect a considerable number of essentially independent units (typically devices such as personal computers and work-stations), and are typically geographically extensive. Depending on the particular type of system, the units which it interconnects are termed end units, end-systems, or hosts.

As a very general matter, there are two extreme forms of system: a pure switching system and a pure broadcast-type system. In a pure switching system, the connections between the hosts are all individual, passing through a network of switching nodes. In a pure broadcast-type system, all end-units are connected to all other end-units by means of a common message medium.

It is clear that both these extreme types of system have major disadvantages. A pure switching system requires a highly complicated network of switching nodes, while there are obvious capacity limits on a pure broadcast-type system. A hybrid style of system has therefore become well established, in which there are local broadcast-type subsystems which are connected to each other by means of a switching system. (In a sense, this constitutes a hierarchy, but the term "hierarchy", and the associated term "levels", are normally used to describe the organization of the more complicated and elaborate forms of switching network.)

A simple and common form of local broadcast-type subsystem is the LAN (local area network). A LAN consists essentially of a common message medium to which a number of hosts are connected. When a host wants to send a message, it monitors the LAN to determine whether any other host is currently using the LAN. If not, then the host sends its message. Every host permanently monitors the LAN, watching to see whether any of the messages on the LAN are directed to itself. (There are various mechanisms for dealing with collisions, where two hosts try to transmit substantially simultaneously.)

There are various specific forms and various modifications of LANs, and there are other similar broadcast (common medium) systems. We shall use the term LAN loosely to cover all such systems, regardless of the details of the manufacturer or protocol.

As noted above, a number of LANs may be coupled together or interconnected by means of a switching network. The switching network in general consists of a number of nodes or switching devices, which we shall term "routers". (Alternative terms are "intermediate systems" and "gateways".) The connection from a router to a LAN is termed an interface; the connection from a router to another router is termed a link.

Obviously, there must be a suitable addressing system. Each host must have an address, and the communication system must somehow deliver messages from any host to any other host.

<u>IP systems</u>

We will consider primarily the types of system known as IP (Internet Protocol) systems from now on, because that is the main type of system for which the present invention is applicable. However, the principles of the present invention are not limited to IP systems, but are also applicable to other systems having similar characteristics, such as Appletalk.

Identifiers and addresses

Hosts generally have unique identifiers which are physically defined by their manufacturers, eg. by hard-wiring or burning in, often termed MAC (medium access control) identifiers. A MAC identifier is normally globally unique; it will typically include a portion distinguishing the manufacturer from all other manufacturers and a serial number distinguishing it from all other machines made by that manufacturer.

It is however preferred to assign each machine a logical address, which can be chosen to facilitate the finding of connection paths in the system. (If desired, a single physical machine can be given more than one logical address, in which case it will behave as more than one logical host.) The MAC identifier is more usually termed a MAC address, but we will use the term "identifier" for MAC addresses to avoid confusion with logical addresses.

In the IP system, the logical address (IP address) is a 32-bit number, which is conventionally divided into 4 bytes or octets which are then written in decimal form (eg. 1.5.21.178). These logical addresses are normally assigned manually.

A major feature of the IP system is that all hosts attached to a LAN have a common high-order part (which is typically the top 3 bytes) of their addresses; this high-order part thus forms the address of the LAN. Thus the host with address 1.5.21.178 will be attached to a LAN with address 1.5.21, and all the other devices attached to that LAN will have addresses with the same high-order part, eg. 1.5.21.17, 1.5.21.8, etc. Each LAN forms a subnet; the address of the LAN is normally termed a subnet address.

If only part of an address is significant, then the significant part is indicated by a mask associated with the address (and of the same length as the address). Thus for the above LAN, the mask will be 255.255.255.0, because only the top 3 bytes of the address are significant. (In theory, the mask can be used to define non-contiguous bits, but in practice this rarely happens.)

The IP system as so far described therefore consists of a network of routers with LANs attached to the routers. A host on a LAN can send messages to other hosts on the same LAN directly over the LAN. To send a message to a host on a different LAN, the host must send the message to the router attached to the LAN. The network of routers then has the responsibility for passing the message to the router attached to the destination LAN. That router then puts the message on that LAN, and the destination host receives it.

System elaborations

There are certain elaborations of this basic system which are worth noting.

First, there is a special address used for broadcasting. In effect, every device has two addresses; its own normal address and the special broadcast address. A message with the broadcast address is received and accepted by every device. Messages with the broadcast address are normally confined to a single LAN; the routers do not attempt to pass such messages through the router network. (There are in fact various special addresses, to allow multicasting (to a group, but not all, of the hosts), and a second broadcast address, but this is not relevant for present purposes.)

Second, a LAN can be connected to more than one router. This may be the most convenient way to connect two parts of the system, with the LAN forming the only connection between the two routers. More often, however, the two routers are both in the same router network, so that they provide two alternative paths to the LAN (from some other LAN). This redundancy allows the system to maintain communication with the LAN even if one of the routers connected to the LAN fails; also, it may allow the message flow rate to or from the LAN to be increased above the limit attainable with one router.

Third, two LANs can be coupled together by means of a bridge in known manner. A bridge is, in effect, a relay device which repeats any message on either of the LANs onto the other LAN. Thus LANs can be connected together into an extended LAN network in manner well known in the art. (We shall use the simple term "LAN" to include extended LANs.)

Fourth, a LAN (which may be a single or extended LAN) can have more than one logical address; the router to which it is attached will treat the single physical port or interface to which that LAN is attached as two separate logical interfaces. Any message put onto such a LAN at any point is physically transmitted to all hosts on it. (In fact, a bridge may have some form of filtering built into it, but this is not relevant for present purposes.) Logically, however, the LAN consists of two or more distinct subnets with different subnet addresses.

IP message flow protocol on LANs

In the IP system, message routing through the router network is determined by the IP addresses, but message routing over LANs is determined by the MAC identifiers. Mechanisms are therefore required to convert IP addresses to MAC identifiers when messages pass over LANs.

There are three main cases to consider for a source host sending a message to a destination host: the destination host may be on the same subnet as the source host, it may be on a different LAN, or it may be on the same LAN but a different subnet.

In the first case, the source sends out an ARP (address resolution protocol) request message with the logical (IP) address of the destination. That ARP request is received by the destination, which sends back an ARP response message to the source. (The ARP request is a broadcast message which is received by all the hosts on the subnet, but only the destination host responds; all the other hosts recognize that the destination address in the ARP request does not match their own address, and they therefore discard the message.) The destination host includes its MAC identifier in its ARP response. The source then sends the actual data message to the destination using the destination's MAC identifier.

This involves a large message overhead, since the passing of each data message is preceded by an ARP request and ARP response. The various units of the system therefore store tables of IP (logical) addresses and MAC identifiers, so that most data messages can be sent out with the MAC identifiers without having to be preceded by ARP requests and responses.

In the second case, the message has to be passed through the router network. In general, each host knows of the existence of at least one router on its LAN. (This may be achieved by routers advertising their presence to their hosts by means of broadcast messages.) The source therefore sends the message to a router (using the router's MAC identifier). The router network forwards the message to a router attached to the LAN including the destination host. That router then sends an ARP request to the destination host, which returns an ARP response. The router then sends the data message to the destination host, using the host's MAC identifier. (If the router does not get an ARP response, the packet is discarded or sent back to the source with an error status.)

The third case is where the source and destination hosts are on the same LAN but different subnets. The first message is passed in the same way as for the second case; the router accepts the message and then transmits it again on the same LAN. The router also returns a redirect message to the source, informing the source of the MAC address of the destination. The source stores this information, and can then send any further messages direct to the destination over the LAN common to the source and destination.

More specifically, each host maintains a connection table which lists the IP addresses and corresponding MAC identifiers of other hosts with which it has recently been in communication (without passing through a router - ie. in the first and third cases above). If a host wants to send a message, the destination is initially identified by its IP address. The host checks its table for the IP address, and if it is in the table, it extracts the associated MAC identifier from the table and sends the data message directly to that MAC identifier. If the IP address is not in the table, then the host has to send an ARP request to obtain a MAC identifier for the data message to be sent to. It enters the MAC identifier and associated IP address in the table for future use.

Similarly, each router has a set of interface tables, one for each interface. Each table lists the logical subnet addresses for that interface and, for each subnet address, lists the hosts with that subnet address, by logical address and MAC identifier. Obviously, the router will only know of the hosts which have sent out ARP requests. Each table in the router also has the identifier of its physical interface associated with it.

The various tables normally incorporates a time-out mechanism, so that entries which have not been used for some considerable time are deleted. This minimizes the chance of a unit trying to send a message to a unit which has disappeared from the system.

Router network organization

As noted above, if a message has to pass through the router network, that network has the responsibility for passing the message to the router attached to the destination LAN. This means that the routers have to pass routing (addressing) information between themselves so that when a router on a LAN receives a message (from a host on that LAN) for another LAN, it will know how to forward the message through the router network (and similarly, the router to which the message is forwarded will in turn know which router to forward the message to, and so on throughout the router network). This routing information is passed between the routers by means of routing control messages.

We are assuming here that a subnet address consists of the top 3 bytes of a 32-bit IP address (as determined by the associated mask), and that all hosts on that subnet have the subnet address as the high-order part of their own addresses. (The routing control messages will also generally contain other information, eg. about the cost and capacity of the paths between routers.) The routers therefore only have to deal with subnet addresses.

We are primarily concerned with the type of network in which every router is in communication, directly or indirectly, with all other routers on essentially the same basis. This is a single-level (level 1) system, and the number of routers will generally be fairly modest for such a system. Various types of routing control mechanisms are known for achieving this; for convenience, we shall assume that the routing control messages are link state packets (LSPs), and that the router network passes the LSPs around so that each router maintains a set of link tables, one for each link to other routers, with each subnet address being held in the table for a link which points to a router which is in some sense nearer to the actual location of that subnet.

Further details of the mechanisms which the router network uses to pass messages (either data messages passing between hosts, or router network control messages) through itself are not relevant for present purposes.

Router network elaborations

This basic mechanism for establishing and maintaining the topology of the router network is subject to possible elaborations.

As noted above, some of the links in the router network may pass through LANs. This does not affect the operation of the system, though of course any messages in the router network which pass through such links have to be encapsulated by the LAN messaging mechanism for their passage over those links.

The routers can operate algorithms for combining (condensing) subnet addresses. Thus if a router has several LANs attached to it with the same top 2 bytes in their addresses (eg. subnet addresses 1.5.21, 1.5.34, 1.5.26, etc), and these LANs are the only ones in the system with these top 2 bytes, that router can identify all those LANs by the single address of just the top 2 bytes (1.5). (As noted above, a mask will define the address as consisting of only 2 bytes.)

This mechanism allows the router network to be hierarchical. In each local (level 1) region of the network, the routers will have full information about all the LANs attached to that region, but will have only summary (condensed) information about the subnet addresses of other local regions. The mechanisms used for passing messages between different level 1 regions of the router network form a second level, level 2, of the network.

The condition for combining subnet addresses in a router in a level 1 area can in fact be relaxed slightly. If there is a odd subnet with say 1.5.102 as its address attached to some other router, the router to which all other subnets with 1.5 as the top 2 bytes in their addresses are connected can advertise itself as the router for address 1.5, provided that it forwards any messages it receives for subnet 1.5.102 on to the other router having the 1.5.102 subnet attached to it.

<u>The problem</u>

In the IP system, the address of a host includes the address of its LAN (as a subnet address). A host is therefore "tied" to its LAN. It can be moved to a physically different place on its LAN; all physical locations on a LAN are logically identical. However, it cannot be moved to another LAN. If it is so moved, it will be inaccessible; although it will be physically attached to the new LAN, no other host will be able to reach it.

In some situations, this restriction on moving hosts is not significant; in others, it provides a useful security feature.

However, in a large company or other organization, there may be a number of different LANs which are connected in an IP system, and for a variety of reasons, such as changes of organization, it may be desirable or even necessary to physically move a host in such a way that it has to be removed from its LAN and attached to another LAN.

This causes a problem. To move a host from one LAN to another, the host's address has to be changed to match the address of the new LAN. Since addresses are manually assigned, it is possible to make this change. Making the change may not in itself be particularly difficult. However, that will in effect turn the host into a new host. None of the other hosts which have been in communication with it will know its new address, and communication will have to be re-established from scratch with all these other hosts. This can be highly inconvenient.

One potential solution to this difficulty is to couple the different LANs together to form an extended LAN, as discussed above. However, this increases the complexity of system management, and involves difficulties arising from the complexity and proprietary nature of multipath bridging. The message density on the extended LAN is also increased, eg. by the increased multicast traffic, and this may limit the extent to which this solution is feasible.

Another potential solution utilizes a directory service. IP systems often have a directory service, which is essentially a table correlating host "names" with their IP addresses. This allows a source host to identify a destination host by means of the destination host's name; but before a source host can actually communicate with a destination host, the source has to obtain the destination's IP address from the directory service by sending the destination's name to the directory service, which returns the associated IP address.

If a host is moved, it can be given a new IP address consistent with its new location, and the directory service can be updated to associate the new IP address with the host's name (which is unchanged). If a source host wants to communicate with the host which has moved, the source host will find that messages directed to the old destination IP address will fail to reach their destination. It can then use the directory service to obtain the destination's IP address, as if it were trying to establish communication with the destination for the first time, and will thereby acquire the destination's new IP address.

This solution requires manual updating of the directory service, which is likely to involve considerable time delays during which the migrated host is inaccessible. It also requires the organization to have a management structure capable of dealing with the changes involved in an acceptably simple and effective manner.

A third potential solution is to provide re-addressing. This involves giving the migrated host a new IP address, consistent with its new LAN, and recording its old and new addresses in the router for its original LAN. A message sent to the host using its old address will reach its old router; that router will replace the old address by the new address and forward the message to the new router. However, this has various disadvantages. For example, message paths through the router network are considerably extended; also, the number of host addresses used in the system is increased each time a host migrates, and the need for the migrating host to be given an address consistent with its new LAN may be inconvenient. Also, the return path for messages between the two hosts is different to the outward path, which can cause difficulties.

The broad object of the present invention is therefore to provide an improved technique whereby a host in an IP or similar system can be moved from one subnet to another without having to have its address changed.

There are some important constraints implied in this formulation of the problem. Any solution must be compatible with existing IP systems; any modifications to only some of the routers and/or hosts in an existing IP system to provide the required technique must not interfere with the operation of the remaining routers and/or hosts. Further, IP-type systems are of course very well established, and include huge numbers of existing hosts. It is therefore desirable, if possible, for the solution to involve modifications to only routers, so that existing hosts can be moved without having to be modified.

As just noted, IP-type systems are very well established, and many such systems are extremely large, both in the numbers of hosts and geographically. The ideal solution in an abstract sense would permit a host to be moved from any location on the system to any other location. However, a solution which allowed only a limited degree of mobility of hosts around the system would be of great practical value, even though it would theoretically be only a partial solution.

The solution

The present invention provides a solution which comprises a combination of several features, all involving modifications of the details of the manner in which the routers operate.

According to the present invention there is provided a digital communication system comprising a network of routers linked together by links and having interfaces with local area networks (LANs) coupled to them, and operating under a protocol under which each LAN has a subnet address, and each host on a LAN has the subnet address as the high-order part of its own address, each router containing a set of interface/LAN tables listing the low-order address portions of the addresses of the hosts attached to the LAN plus the MAC (medium access control) identifiers of those hosts, and a set of link tables listing the subnet addresses of the LANs reachable through those links, wherein: both the interface tables and the link tables in the routers contain the full addresses of all hosts reachable through those interfaces and links; the routers contain means for detecting ARP (address resolution protocol) requests from a source host for a destination host having the same subnet address as the source host but not on the same interface, and returning a proxy ARP response giving the router's identification; and the routers contain polling means for polling the interfaces for unknown hosts.

Since the present system does not require any changes to the hosts, the address space of the system is unchanged. However, the present system in effect decouples the host addresses from the subnet addresses and hence from the geographical LAN locations. This allows considerably greater freedom in assigning addresses to hosts within the system address space.

In the standard system as described above, we have taken the top 3 bytes of the 32-bit address space as being used for subnet addresses, and the bottom byte as being used for different host addresses on the subnet. In fact, the division between the subnet address and the host addresses on the subnet can be defined more flexibly, by the use of suitable masks. However, the number of possible host addresses on a subnet must obviously be a power of 2, and the actual number of hosts on the subnet is likely to fall well short of the maximum.

Thus in the standard system, there are likely to be many spare addresses, which cannot be used (or can only be used by hosts added to the subnet with which those unused addresses are associated). In the present system, these spare addresses can be used much more freely, since they can be assigned to hosts regardless of which subnets (and hence LANs) those hosts are to be attached to.

<u>Router network organization</u>

A major feature of router operation is that the present modified routers use full host addresses for level 1 routing.

In the standard IP system, the routers use abbreviated addresses - the subnet addresses of the LANs - for level 1 routing; in effect, the routers operate an address compression algorithm which compresses the addresses of all the hosts on a subnet into a single subnet address. In the present system, the router operation is modified so that this address compression is no longer performed.

The result is that in a router network using the present modified routers, each router will hold effectively the same routing information as before, albeit in an expanded form. The operation of the router network is therefore effectively unchanged in principle (as far as the routing of messages through the router network is concerned). However, the amount of LSP traffic is increased, the amount of processing required for routing is increased, and the routers have to have a greater storage capacity.

If the level 1 network forms part of a larger system coupled to other level 1 networks through a level 2 organization, the level 2 organization is unaffected. Compressed or summary addresses are used unchanged for level 2 routing. The migration of hosts is restricted to within their own level 1 systems; it is not possible for a host to migrate from one level 1 system to another. As mentioned above, this restriction is rarely significant.

The present routers are largely compatible with standard routers, so that a network can consist of a mixture of standard and compatible routers. For present purposes, it is convenient to regard the resulting network as a network of modified routers to which standard routers have been added.

In a standard router network, the routing information consists of subnet addresses, which are distributed by the LSPs and stored by the routers. A subnet address has the form of a full address plus a mask, with the mask defining which part of the full address forms the subnet address; the rest of the address is ignored. The ignored part of the address is in fact, of course, a host address (on the subnet defined by the mask).

In a mixed system, the modified routers will send LSPs with full addresses in the same format, ie. address plus mask pairs, and any standard router receiving such an LSP will automatically store this address in the usual way. As far as such a standard router is concerned, there is no difference between subnet and full (host) addresses; the distinction arises solely from the contents of the masks associated with the various addresses. In a mixed system, therefore, the presence of standard routers will not affect the performance of the subsystem of modified routers (provided, of course, that the standard routers have sufficient storage capacity). The migration of hosts in such a mixed system is of course limited to the subsystem of modified routers.

As noted above, in the present system the amount of address information which has to be propagated through the router system is considerably increased. It may therefore be desirable to introduce a new LSP option type or format, to reduce the size and/or number of LSPs.

In the standard system, routing information is exchanged between the routers in the form of information units termed "options", of which there can be various formats or types. To reduce the number of LSP messages, a number of options are typically assembled into a single LSP. The standard option type can be taken as consisting of a header, an address section, and a general information section. The header contains an identifier which defines the LSP option type and length; the address section will consist of the address plus mask pair; and the general information section will contain associated routing information such as cost and distance.

The new LSP option type has the same general format, but the address section contains a set of host addresses without masks. Thus a considerable number of addresses can be sent as a single option of the new type, instead of needing a separate option (of the old type) in the LSP for each address. The length of this new LSP option will be considerably less than the total length of the separate LSP options of the old type, because there will be only one header and general information section, and each address will consist of a pure address with no accompanying mask.

The number of addresses in the new LSP option type may be included explicitly in the header, or may be calculated from the total length of the option by subtracting the header and general information lengths and dividing by the address length. Different addresses cannot, of course, have different associated routing information, because the addresses all share the common general information in the final section of the option. The routers will normally assemble the addresses of hosts on a common LAN when constructing an LSP option of the new type; those addresses will then all have the same characteristics and can share the same general information.

The new LSP option type can be used in a mixed system, as standard routers forward all LSP options (including those of the new type); the full host address information will thus be maintained throughout the subsystem of modified routers. However, the standard routers will not update themselves with the contents of LSP options of the new type. The modified routers must therefore also send out LSP options of the old type, so that the routing information in the standard routers is maintained. Also, if the standard routers split the subsystem of modified routers into disconnected parts, hosts cannot migrate between those parts because the standard routers connecting those parts will maintain only subnet addresses, not full host addresses.

LAN addressing by routers

For the router network to be able to route messages correctly, each router must know the whereabouts of all the hosts. This knowledge is distributed amongst the different routers using LSPs. However, before the a router can distribute information about the location of a host, it must become aware of the existence of the host.

In the standard system, each router is aware of the subnets attached to it (this knowledge may, for example, be entered manually). A router need not be explicitly aware of the existence of the hosts attached to the LAN. If a message for a host on the LAN is received, the router can send out an ARP request, and the ARP response confirms the existence of the host on the LAN. (If there is no ARP response, then it is assumed that the host does not exist.)

As discussed above, in the standard system the router in fact maintains an interface table for the various hosts on the LAN, so that it can forward future messages to them without having to obtain their MAC identifiers by ARP requests. The interface table is built up partly from ARP requests sent out by the router, and partly by ARP requests sent out from the hosts.

A standard router preferably maintains this table actively, by polling the hosts at suitable intervals. The poll message is simply an ARP request to the host. An ARP response confirms the existence of the host; if there is no ARP response, the host no longer exists.

In the present system, the modified router maintains its interface tables in broadly the same way as do standard routers. The present router, however, necessarily constructs its interface tables entirely automatically, whereas in a standard router the subnet addresses may be entered manually.

The present routers listen promiscuously for ARP requests from hosts (for a reason discussed later), and may listen similarly to other messages. This listening helps the routers to maintain their interface tables. The routers will therefore automatically learn of the existence of hosts which are involved in message transmission. However, it is possible that when a host is moved from one router to another, some other host may want to send a message to it before it has itself tried to send any messages. (In the standard IP system, hosts do not advertise themselves; the identity between the subnet address and the top part of the host address means that the location of a host is inherent in the system.)

Another way in which the routers can automatically discover the existence of hosts is for the hosts to announce their existence when they are first turned on, with the routers listening for such messages. However, this requires the hosts to issue suitable identification messages when first turned on; this may require modification of some hosts, and some types of host may not be modifiable.

Some mechanism must therefore be provided for the router network to discover the existence of silent hosts. Since changes cannot be imposed on the hosts themselves, the standard routers must therefore be modified, in the present system, so that they can discover the existence of such silent hosts. There are two ways in which this can be done, which may be termed active and passive. (It may be noted that in standard routers, the subnet addresses may be passed round automatically but will normally be set manually. In the present system, the subnet addresses are not of such importance, and the modified routers must determine all host addresses automatically.)

With the active technique, the routers actively search for hosts. Each router has to be modified to perform polling. For this, the routers in the level

1 area are informed (eg. manually) of the address space of the hosts in that area. Each router then polls each of its interfaces in turn; that is, for each interface in turn it sends out a series of ARP requests, working through the host address space address by address. It will therefore elicit responses from all hosts attached to it.

This polling is of course distinct from the polling, mentioned above, which standard routers perform. The standard router polling is not through the host address space, but through the actual addresses of hosts which are already recorded in the routers' tables, to confirm their existence.

Provided that the host address space is manageably small, this is the preferred mechanism. This polling automatically takes care of the normal maintenance of the interface tables. Since the router network can only route messages to hosts which it knows about, it is important to confirm the disappearance of a host, eg. by a suitable number of retries. A discovery time of say 500 s (comparable to the ARP time-out), and a polling rate of 7 polls per second will accommodate an address space of 4000 hosts, which is much larger than most practical LANs.

The polling message density can be reduced if a router does not poll for addresses which it knows to be attached to other of its interfaces, or to other routers. However, a router needs to poll for hosts which are attached to it to confirm their existence, just as with a standard router; also, polling for hosts which are listed in its tables as being attached to other routers accelerates their discovery if they are moved.

Instead of polling by ARP requests, a router could poll by sending a suitable broadcast message, asking the hosts to report their existence. However, this has two disadvantages. One is that it requires the hosts to return suitable identification messages in response to the broadcast enquiry; this may require modification of some hosts, and some types of host may not be modifiable. The other is that the response messages from the hosts will temporarily produce a very high message density, which may for example overwhelm the router.

With the preferred mechanism of ARP polling, the polling intensity can be reduced by partitioning the host address space so that certain segments of it will only contain hosts which will announce their presence when first turned on. It will then not be necessary to poll through those address space segments.

A possible refinement of ARP polling is that if a router discovers that a host has disappeared, that host address can be distributed to all routers, with all routers then sending out ARP requests at higher than normal polling frequency for that host for some convenient period of time. (The router which has lost the host should be included in this, because the host may be migrating to another of its interfaces.) This will result in rapid detection of the migrating host if it is reconnected into the network.

With the passive technique for routers to discover the existence of silent hosts, they only search for a host when there is a message for that host. If the router network receives a message for a host which it (the router network) does not recognize, then the message is passed around the routers, and each router polls each of its interfaces with an ARP request.

This requires a more complex router network organization, to ensure that the message is distributed to all routers, but it may reduce the amount of polling, as the occurrence of messages to silent hosts will usually be relatively uncommon. The message may be distributed rapidly to all routers, with all the routers then polling their LANs; this may impose a significant transient load on the system. Alternatively, each router in turn may poll its interfaces for the destination host, and forward the message on to the next router only if it fails to find the destination host on any of its interfaces; this may result in a large delay.

Host to host communication

In the standard system, there are 4 mechanisms for a source host to send a message to a destination host. First, if the destination has the same subnet address as the source, if the source does not know the destination's MAC identifier it will send an ARP request to the destination; otherwise (second), it sends the message direct to the destination using the MAC identifier. Third, if the destination is on a different subnet, the source sends the message to a router. Fourth, if the destination is on a different subnet but the same extended LAN as the source, the source can send direct to the destination's MAC identifier as a result of a redirect message from a router.

The present system must maintain all these modes of message transmission as far as the hosts are concerned; in particular, it must cope with all possible combinations of source and destination subnet addresses and LAN locations. The source and destination may be on the same or different LANs, and may have the same or different subnet addresses.

If the source and destination are on the same LAN and have the same subnet address, then if the source knows the destination's MAC identifier it will send the message direct to the destination using the MAC identifier. Otherwise, the source will send an ARP request to the destination and, because the two are on the same LAN, it will get an ARP response and then send the message using the MAC identifier returned in the ARP response. This is the same as in the standard system.

If the source and destination are on different LANs and have different subnet addresses, the source will send the message to a router, which will forward it to the router to which the destination is attached. This is broadly similar to the standard system (though the router uses the more detailed routing information of the present system).

If the source and destination are on the same LAN but have different subnet addresses, the source will send the message to a router; this will return a redirect message to the source, which will then send the message direct to the destination using the destination's MAC address. This is broadly similar to the standard system (though again the router uses the more detailed routing information of the present system).

If the source and destination are on different LANs but have the same subnet address, then the source will send an ARP request to the destination, expecting to receive an ARP response with the destination's MAC identifier. The router on the source LAN must listen for such ARP requests and return ARP responses (this is the promiscuous listening for ARP requests mentioned above). On hearing an ARP request on an interface, the router must check its link tables and its interface tables for its other interfaces for the destination. If the destination is in those tables, it is in fact on a different LAN from the source. However, the source is expecting an ARP response. The router must therefore return a proxy ARP response - ie, it must return an ARP response on behalf of the destination. This proxy ARP response will of course contain the router's MAC identifier. The source will then send the message to the router, which must then forward it through the router network.

Specific Embodiment

A communication system embodying the invention will now be described, by way of example, with reference to the drawings, in which:

Fig. 1 is a general block diagram of the system; and

Fig. 2 is a highly simplified block diagram of a modified router (ie. the present router).

Fig. 1 shows a communication system with various typical features. The system consists of a level 1 network of 3 routers R1-R3 coupled by links LK12, LK23, and LK13 (the digits indicating the routers which each link couples together). This level 1 network forms part of a level 2 system (the rest of which is shown merely as a cloud L-2), and is coupled to the rest of the level 2 network by links shown as zig-zag lines.

Router R1 has 2 physical LAN interfaces, with LAN1 (with subnet address 32.17.52) and LAN4 (with subnet address 32.17.66) coupled to them; router R2 has 3 LAN interfaces, with LAN2 (with subnet address 32.17.55), LAN3 (with subnet address 32.20.132), and LAN6 (with subnet address 32.20.154) connected to them; and router R3 has 2 LAN interfaces, with LAN5 (with subnet address 32.27.24) and LAN7 (with subnet address 32.17.102) connected to them. LAN1 and LAN2 are connected together in known manner through a bridge BR1, forming a single extended LAN with two subnet addresses.

Hosts H1-H11 are coupled to the various LANs as shown. Each host has an address consisting of 4 bytes. In the standard system, each host's address will be the address of its LAN plus a final byte added to the end of the subnet address, as shown for hosts H2 (address 32.17.52.32), H3 (address 32.17.52.5), H5 (address 32.17.55.129), and H8 (address 32.17.66.188).

Each host maintains a connection table for its connections. Host H2, for example, will maintain the following table:

<u>Host H2, connection table</u>

Router connection:
R1-MAC

Host list:

32.17.52.5 (H3)	H3-MAC
32.17.55.129 (H5)	H5-MAC
...	

This table has two parts, a router connection and a host list. The router connection part is a single entry, the MAC of router R1, which the host uses for sending messages to hosts which are not on its own extended LAN. The second part lists the hosts which H2 has recently sent messages to, together with the MAC identifiers which it uses to send messages to those hosts. Communication with H3 is direct, over LAN1, so messages to that host are sent to that host's MAC identifier. Communication with H8 is via the router network, so messages to that host are sent to router R1, using that router's MAC identifier. Communication with H5 is also direct; H2 has learnt H5's MAC identifier as the result of a redirect message from router R1 or R2 at some time in the past.

The host maintains this table as a cache with time-out, so that entries which have not been used for more than a certain time are deleted. New entries are added as communication with new hosts is desired, by using the ARP requests as discussed above.

Each router maintains interface and link tables. Router R1, for example, would maintain the following interface tables if it were a standard router.

Router R1 (standard form), interface tables

IF1 (interface)
    32.17.52 (subnet address)
        5   H3-MAC
        32   H2-MAC
        ...
    32.17.55 (subnet address)
        129 H5-MAC
        ...

IF2 (interface)
    32.17.66 (subnet address)
        188 H8-MAC
        ...

Each interface table is divided into a separate section for each logical subnet address of the (possibly extended) LAN attached to that interface. Each section records the subnet address and then lists the hosts on the LAN with that subnet address. Each host entry consists of the host's address and its MAC identifier. The host's address is recorded as only the final byte, since the first 3 bytes of the address are the address of its subnet. The first interface table has two sections because the two LANs LAN1 and LAN2, with different subnet addresses, are both connected to that physical interface (via the bridge BR1 in the case of LAN2).

The routers also maintain link tables for their links to other routers. In the standard system, each router passes the addresses of the LANs to which it is coupled to the other routers in the level 1 network, and those other routers hold that information in their link tables. Thus if router R1 were standard, it would maintain two link tables as follows.

Router R1 (standard form), link tables

LK12:

32.20.154	(LAN6)
32.20.132	(LAN3)
...	

LK13:

32.27.24	(LAN5)
32.17.102	(LAN7)
...	

The entries in each table are the addresses of the LANs which can be reached through the associated link. If link LK13 did not exist, then the link table LK12 would contain the addresses 32.27.24 and 32.17.102 (as well as 32.17.154 and 32.20.132), because the message route for those addresses would then be via router R2.

Each subnet address in the link tables can be regarded as a compressed version of the set of host addresses on that LAN. This can be represented more fully by writing the subnet addresses as 32.17.154.xx, etc, where the final byte is masked off by a mask.

For the coupling to the rest of the level 2 system, the routers provide further compressed addresses over the zig-zag links to region L-2. In this case, the level 2 addresses will be simply the single value 32.0001xxxx.xx.xx (where the second byte is written in binary). Similarly, the routers R2 and R3 will maintain level 2 connection tables (with further compressed entries) for addresses in the L-2 region. Also, router R1 will maintain these region L-2 addresses (preferably in the same compressed form) in its link tables LK12 and LK13, so that messages from hosts on its LANs to the L-2 region can be correctly routed. (This is why the tables LK12 and LK13 are shown as having further entries beyond the 2 shown explicitly for each.)

In the present system, the routers are modified from the standard form to maintain the level 1 connection information within the interface and link tables in uncompressed form. Thus the interface table for interface IF1 for router R1 will contain:

Router R1 (modified form), interface table IF1

IF1:

32.17.52.5	H3-MAC
32.17.52.32	H2-MAC
32.17.55.129	H5-MAC
32.20.154.34	H1-MAC
...	

Switching, for convenience, to router R3 to discuss the link tables, this would contain the following link table for link LK13 in the standard form:

Router R3 (standard form), link table LK13

LK13:

32.17.52	(LAN1)
32.17.55	(LAN2)
32.17.66	(LAN4)
...	

For the modified form of router R3, this link table will contain all the entries for router R1's interface tables, in the same form as in those interface tables, instead of just the compressed or subnet addresses. Thus router R3 will contain the following link table for link LK13 in the modified form:

Router R3 (modified form), link table LK13

LK13:

32.17.52.5	H3-MAC
32.17.52.32	H2-MAC
32.17.55.129	H5-MAC
32.17.66.188	H8-MAC
...	

The connection information relating to connections to the level 2 area L-2 is unchanged from the standard form.

With the host addresses discussed up to now, the operation of the system with the present (modified) routers is substantially unchanged from the operation with standard routers. Suppose, however, that host H1 was originally on LAN6 (address 32.20.154), and was given the address 32.20.154.34 while it was on that LAN. Suppose also that it is desirable to transfer that host to LAN1 as shown.

In the standard system, it would not be possible for any messages to reach H1, because its address does not match LAN1's address. For H1 to be logically connected to the system, either its address would have to be changed to match that of LAN1 (so that it would effectively be a new host), or LAN1 and LAN6 would have to be coupled together by a bridge (so that H1 could still be reached by router R2), or some form of address conversion would have to be provided.

In the present system, however, messages can reach H1. This is because all the routers' tables (ie. both link and interface tables) contain the individual addresses of all hosts (of the level 1 area) in full; they do not now contain the subnet addresses as such.

Thus router R1 will contain the address 32.20.154.34 in its interface table for interface IF1, so that it can forward a message for H1 reaching it from another router (or from another of its interfaces). Similarly, the link tables of R2 and R3 will contain the address 32.20.154.34 in full, so that any message for H1 reaching either of those routers can be forwarded to router R1 (assuming that for some reason, messages are not passed to it from router R2 through the extended LAN network of the 2 LANs LAN2 and LAN1).

In the standard system, router R2 would contain the address 32.20.154 of LAN6 in one of its interface tables, and would capture all messages to any host with that as the first 3 bytes of its address. In the present system, however, router R2 contains only the addresses of the individual hosts attached to it, not the subnet address as such. It will therefore not capture any messages to host H1, ie. to address 32.20.154.34, and will therefore not interfere with the correct routing of messages to that host.

Fig. 1 also shows a second host, host H9, which has migrated, in this case from LAN1 to LAN4. Router R1's interface table for interface IF2 contains the address (32.17.52.47) of this host, and routers R2 and R3 contain this address in their link tables LK12 and LK13, so that messages to this host from LANs attached to R2 and R3 will reach it as desired.

There is however a complication if a host such as H2 on H9's original or "home" subnet wants to send a message to it. H2 finds that its own subnet address is the same as H9's subnet address, and therefore sends an ARP request to H9 on LAN1 (hosts' behaviour is unchanged from in the standard system).

As discussed above, the present routers listen to all ARP requests from hosts on their interfaces, to detect ARP requests for migrated hosts. When a router detects an ARP request, it checks the address of the destination host against the contents of its tables (both the interface tables and the link tables). If the destination host is on the same interface as the source host, the router ignores the ARP request (the destination host will respond to the ARP request with an ARP response and message transmission will proceed normally). But if the destination is not on the same interface as the source, the router responds with a proxy ARP response which includes its own MAC identifier. The source will then send the message to the router, and the router then forwards the message to the actual location of the destination.

Thus router R1 will detect the ARP request from host H2 for host H9, find that host H9 is not on interface IF1, and return a proxy ARP response to H2. H2 will then send the message to the router, which will pass it to interface IF2 so that it reaches H9.

If host H2 has previously been in communication with host H9 over their original common LAN, H2 will of course still have H9's MAC address in its connection table, and will continue to use that MAC address when trying to send messages to H9; and when H9 migrates, H2 will find that H9 has apparently disappeared. H2 will thereupon flush its connection table (or at least the entry for H9), and attempt to re-open communication with H9 by sending an ARP request. Router R1 will return a proxy ARP response to this, as just discussed, and communication with H9 will therefore be re-established.

Fig. 2 shows the general logical organization of the preferred form of modified (present) router, which we may take as router R1.

There is a plurality of link registers 10, one per link, for receiving messages (including LSPs) coming in over links from other routers and for holding messages to be transmitted over those links. There is a plurality of interface registers 11, one per interface, for receiving messages coming in over the router's interfaces and for holding messages to be transmitted over those interfaces. There is a plurality of link table stores 12, one per link, for storing the link tables discussed above. There is a plurality of interface table stores 13, one per interface, for storing the interface tables discussed above. The link and interface registers 10 and 11 are coupled to switching circuitry 15.

Each of the link registers 10 and interface registers 11 has a header section 10A, 11A respectively for containing header information including, for example, the source and destination addresses and (in the case of messages in the interface registers) MAC identifiers. When a message is received in one of these registers, its destination address is compared by a comparator 14 with the addresses in the link and interface tables and moved from its initial register to the appropriate register for output, ie. from a link register to another link register, from a link register to an interface register, or from an interface register to an interface register. In addition, if the message is moved into an interface register, the MAC identifier of the destination is copied over line 16 from the interface table into the header section of the interface register.

(In practice, the messages may be stored in a common memory, with pointers being used to identify different memory areas as the different registers, and the movement of a message from one register to another being achieved by changing the pointers. Also, the headers may be processed separately from the bodies of the messages.)

The interface registers 11 are also coupled to an ARP unit 30. All ARP requests on the LANs attached to the interfaces are received by the router, ie. are written into the interface registers 11. When an ARP request is so received, comparator 14 compares the host destination address in its header with the host addresses in the link tables 12 and the interface tables 13.

If the destination is not in the interface table for the interface on which the ARP request was received, ie. is in some other interface table, or in a link table, then the comparator 14 sends a signal to the ARP unit 30, which then converts the ARP request in the interface register to a proxy ARP response. This ARP response includes the router's address and MAC identifier, which are stored in a router address and MAC identifier store 17 and are copied into the header into of the interface register for return to the host as the ARP response.

If the destination is in the interface table for the interface on which the ARP request was received, then the router makes no response. However, for all ARP requests which it receives, the router checks whether the source is listed in the interface table for the interface on which the ARP request was received. If it is not, then it updates its tables by adding the source's address to the appropriate interface table and deleting it from any other tables which it is in. In addition, the host's address (ie. the full address) is passed (over line 21) to an LSP processor 20.

A polling unit 25 is also coupled, through the switching circuitry 15, to the header sections 11A of the interface registers 11. The polling unit 25 performs two functions, under the control of a timer 26 to which it is coupled.

First, the polling unit is coupled to the interface tables 13, and selects each entry in the interface tables in turn for verification. For this, the address of each end-station in turn is copied into the appropriate one of the interface registers 11 and sent out as an ARP request. The MAC identifier in the response is passed back to the interface table and compared therein with the stored MAC identifier, to verify the entry. If verification fails (after a suitable number of retries), the table entry is deleted and the address of the deleted host is passed to the LSP processor 20.

Second, the polling unit is also coupled to an address space store 27, which is set to contain the address space of the (level 1) system. Under control of the timer 26, the polling unit 25 works sequentially through all addresses of the address space. Addresses which are already in the interface tables are filtered out. The remaining addresses are passed, in sequence, to each of the interface registers for sending out an ARP request, to see whether a host with that address exists. If it does, then the address and MAC identifier in the ARP response are passed to the appropriate interface table and to the LSP processor 20.

Turning now to the LSP processor 20, this receives the addresses of hosts newly discovered by the router and of hosts which disappear from the router's interfaces. It constructs LSP options containing these addresses, assembles them into LSPs, and passes them to the set of link registers 10 for transmission to other routers. This processor 20 also processes LSP options received by the link registers 10 from other routers, updating the entries in the corresponding link table 12 by adding and/or deleting entries appropriately. The LSP processor 20 is coupled to an LSP option memory 22 in which LSP options of the new type discussed above are constructed; this memory comprises a header section 22A, an address section 22B for the addresses of the LSP, and a general information section 22C. This memory is used to assemble LSP options of the new type which are to be sent out by the router, and to store incoming new type options received from other routers ready for analysis and transfer of their contents into the link tables 12.

In the system shown in Fig. 1, each router is coupled to every other router. In general, however, this will not always be so. LSP options must therefore be forwarded throughout the level 1 area. The LSP processor is responsible for this; it causes an incoming LSP option to be copied to all other link registers 10 for forwarding (as parts of LSPs) to other routers. Various techniques can be used to prevent the unlimited circulation and multiplication of LSP information.

Claims

1. A digital communication system comprising a network of routers linked together by links and having interfaces with local area networks (LANs) coupled to them, and operating under a protocol under which each LAN has a subnet address, and each host on a LAN has the subnet address as the high-order part of its own address, each router containing a set of interface/LAN tables listing the low-order address portions of the addresses of the hosts attached to the LAN plus the MAC (medium access control) identifiers of those hosts, and a set of link tables listing the subnet addresses of the LANs reachable through those links, wherein:

both the interface tables and the link tables in the routers contain the full addresses of all hosts reachable through those interfaces and links;

the routers contain means for detecting ARP (address resolution protocol) requests from a source host for a destination host having the same subnet address as the source host but not on the same interface, and returning a proxy ARP response giving the router's identification; and

the routers contain means for interrogating the interfaces for unknown hosts.

2. A digital communication system according to claim 1 wherein the means for interrogating the interfaces comprises polling means.

3. A digital communication system according to claim 2 wherein the polling means include timing means causing the polling means to perform polling for unknown hosts.

4. A digital communication system according to claim 3 wherein each router contains an address space store settable to contain the address space of the system.

5. A digital communication system according to claim 2 wherein the polling means of a router poll for unknown hosts is in response to the router receiving a message for an unknown destination host, and the message is passed through the network of routers until the destination host is located.

6. A digital communication system according to any previous claim wherein each router contains an LSP option memory for assembling, storing, and analyzing LSP options, the LSP option memory comprising a header section, an address section capable of storing a plurality of addresses, and a general section.

7. A method of operating a digital communication system comprising a network of routers linked together by links and having interfaces with local area networks (LANs) coupled to them, said method including the steps of:

operating the system under a protocol under which each LAN has a subnet address, and each host on a LAN has the subnet address as the high-order part of its own address;

providing each router with a set of interface/LAN tables listing the low-order address portions of the addresses of the hosts attached to the LAN plus the MAC (medium access control) identifiers of those hosts, and a set of link tables listing the subnet addresses of the LANs reachable through those links;

providing both the interface tables and the link tables in the routers with the full addresses of all hosts reachable through those interfaces and links;

each router, upon detection of ARP (address resolution protocol) requests from a source host for a destination host having the same subnet address as the source host but not on the same interface, returning a proxy ARP response giving the router's identification; and

providing each router with means for interrogating the interfaces for unknown hosts.

8. The method of claim 7 including the step of locating unknown hosts by a router by routine systematic polling of predetermined address space.

9. The method of claim 8 including the step of reserving a second predetermined address space for self-announcing hosts which address space is not systematically polled by a router.

10. The method of claim 7 including the step of initiating a poll for an unknown host in response to a router receiving a message for said unknown destination host, and passing the message through the network of routers until the destination host is located.

Patents Act 1977 **Examiner's report to the Comptroller under Section 17** **(The Search report)**	Application number GB 9322910.2
Relevant Technical Fields (i) UK Cl (Ed.M) H4P (PPA, PPG) (ii) Int Cl (Ed.5) H04L 12/46, 12/66	Search Examiner MR J P COULES Date of completion of Search 7 DECEMBER 1994
Databases (see below) (i) UK Patent Office collections of GB, EP, WO and US patent specifications. (ii) ONLINE DATABASES: WPI, INSPEC	Documents considered relevant following a search in respect of Claims :- 1-10

Categories of documents

X: Document indicating lack of novelty or of inventive step.

Y: Document indicating lack of inventive step if combined with one or more other documents of the same category.

A: Document indicating technological background and/or state of the art.

P: Document published on or after the declared priority date but before the filing date of the present application.

E: Patent document published on or after, but with priority date earlier than, the filing date of the present application.

&: Member of the same patent family; corresponding document.

Category	Identity of document and relevant passages		Relevant to claim(s)
A, E	GB 2267418 A	(ICL PERSONAL SYSTEMS OY) 1 December 1993 whole document	1 and 7

Databases: The UK Patent Office database comprises classified collections of GB, EP, WO and US patent specifications as outlined periodically in the Official Journal (Patents). The on-line databases considered for search are also listed periodically in the Official Journal (Patents).

Electronic Acknowledgement Receipt

EFS ID:	6625922
Application Number:	90010416
International Application Number:	
Confirmation Number:	1061
Title of Invention:	Point-to-Point Internet Protocol
First Named Inventor/Applicant Name:	6108704
Customer Number:	42624
Filer:	Michael R. Casey
Filer Authorized By:	
Attorney Docket Number:	2655-0188
Receipt Date:	14-DEC-2009
Filing Date:	17-FEB-2009
Time Stamp:	11:02:59
Application Type:	Reexam (Third Party)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal Letter	20091214_IDS_0188.pdf	119104 a78aa65dfc7941xc8e543fc680ed0ca3421c448fb	no	2

Warnings:

Information:

2	Information Disclosure Statement (IDS) Filed (SB/08)	20091214_1449_0188.pdf	5132723 90cc4cc10a75dc4fb48e582173da107148b4cb3c	no	30
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	NPL Documents	F0000_CircuitSwitching.pdf	72814 4c3f34c22d2f56a6c82c889818a1376a3ae835d6	no	2
Warnings:					
Information:					
4	NPL Documents	F0001_CCITT_Fax_Recommendation.pdf	2878754 bd5979dd97b05d2d70b3fa5c0d6febbda1c5cef5	no	76
Warnings:					
Information:					
5	NPL Documents	F0002_IBM_Full_Duplex_SpeakerPhone.pdf	188679 cb97c993b041210becea018966bdd2259894de0f	no	4
Warnings:					
Information:					
6	NPL Documents	F0003_One_Per_Desk.pdf	1400117 72c9138f2988bc43ac628c2b75d736de8b71c996	no	24
Warnings:					
Information:					
7	NPL Documents	F0004_NTIS_WideBand_Integrated.pdf	1550423 59ffcb33e6fbd5e5941bb0297223ff17072849aa	no	39
Warnings:					
Information:					
8	NPL Documents	F0005_Integrated_VoiceData_PABX.pdf	87079 b36ad7f1eff3a459b4bee53124753c5ac75f0b92	no	2
Warnings:					
Information:					
9	NPL Documents	F0006_level_1-5.pdf	83654 885a286424d1ffb70f2eafb905db6695283a92f9	no	2
Warnings:					
Information:					
10	NPL Documents	F0007_Multi-Service_Networks.pdf	70355 609b6ba08e641be9acd8de0d6c7270ccdc93d2a5	no	2

Warnings:					
Information:					
11	NPL Documents	F0008_The_History_Of_TCP_Int.pdf	126453 9f3b2c7baa0e4aba3f052942d3e641aecab14c8d	no	2
Warnings:					
Information:					
12	NPL Documents	F0009_WideBandPacketAccess.pdf	460534 64d292b8de4355aa835d5f3124ca203a8c21daf2	no	7
Warnings:					
Information:					
13	NPL Documents	F0010_Ahrens_Richard_FAQ.pdf	236274 dd81539f88e976d03f61caf7ae2613dfcae165b7	no	6
Warnings:					
Information:					
14	NPL Documents	F0011_AndyHopper_Pandora.pdf	814785 a38f37e55bf5e26e58e9882331b81c42dcba0625	no	17
Warnings:					
Information:					
15	Foreign Reference	F0012_AU-200059377-A1.pdf	1204979 f4a9fef78f8bf3bbbb696e1e203fc826970a0d02	no	36
Warnings:					
Information:					
16	Foreign Reference	F0013_AU-200059378-A1.pdf	1310540 708f072d3b0326860aa3f0deadd68c03e1a7673f	no	39
Warnings:					
Information:					
17	Foreign Reference	F0014_AU-200059379-A1.pdf	1602806 09a8fc70c04c67bda29052a79de04f94be1848a2	no	46
Warnings:					
Information:					
18	NPL Documents	F0015_Bennett_Designing_TCPIP_Networks.pdf	231122 21ee74366cccb037695552a191d7d967568504f	no	5
Warnings:					
Information:					
19	NPL Documents	F0016_Bernard_Gold_Digital_speech_networks.pdf	1612953	no	23

Warnings:					
Information:					
20	NPL Documents	F0017_Bill_Newman_ISDN_Data_and_Voice.pdf	249255 822beb08e99b78517d6bddfe03697014e2fe19a	no	7
Warnings:					
Information:					
21	NPL Documents	F0018_borland_john_technology_uses.pdf	356477 c051baa16e035ee7f3b4634a11c1178b8cefd9dc	no	3
Warnings:					
Information:					
22	NPL Documents	F0019_rfc1530.pdf	289167 1fb3dbb44d74e7de5d600af4541e59fe3ff91989	no	7
Warnings:					
Information:					
23	NPL Documents	F0020_rfc1529.pdf	218374 7e8a6e910ce8d9a00830bac852eb071157afb7de	no	5
Warnings:					
Information:					
24	NPL Documents	F0021_rfc1528.pdf	334059 04c5a34d821d18619f6942291efc4bfab1d2b936	no	12
Warnings:					
Information:					
25	NPL Documents	F0022_rfc1190txt.pdf	235842 607a6ee50cae87d9ef865aba4edba87b2a840fb7	no	148
Warnings:					
Information:					
26	NPL Documents	F0023_rfc1789.pdf	760350 d7de41b0d92bd6f2a01e483a7e3938100d123e84	no	6
Warnings:					
Information:					
27	NPL Documents	F0024_Broad-band_personal_computer_lans.pdf	671846 17ba6b37f68b7c1be528e64cf29cc849018bd0c1	no	8
Warnings:					
Information:					
28	NPL Documents	F0025_cindy_mueller.pdf	166643	no	2

Warnings:					
Information:					
29	NPL Documents	F0026_weinstein_experiences.pdf	1565803 fe191064adbe072f84e24ff1e3eed9e219f18beb	no	19
Warnings:					
Information:					
30	NPL Documents	F0027_d_adolphs_adapters.pdf	560241 6af1fc4a6ac826b87abddc2b1e26c33f0434419f	no	9
Warnings:					
Information:					
31	NPL Documents	F0028_rfc1171.pdf	1235123 9986ea61542c08ff417dfa4a7d4536d37e2c6b1e	no	48
Warnings:					
Information:					
32	NPL Documents	F0029_swinehart_adding_voice.pdf	674818 f0c14fa062141214c36fb80b03a4ed245bcc2cb6	no	8
Warnings:					
Information:					
33	NPL Documents	F0030_dale_gulick_interface_the_ISDN.pdf	278408 caf00286c171e8392f8d313736de04adefbb5a68	no	4
Warnings:					
Information:					
34	NPL Documents	F0031_swinehart_telephone_management.pdf	511941 ba1465c0fd2b4754e08907068e754bb5038b548e	no	7
Warnings:					
Information:					
35	NPL Documents	F0032_danny_cohen_nvp-II.pdf	2564697 b601969244c4b2feb1d2c1276a8625d885424b93	no	74
Warnings:					
Information:					
36	NPL Documents	F0033_danny_cohen_packet_comm.pdf	1076825 b0c02194acc804a06085f975462138bea815bc12	no	10
Warnings:					
Information:					
37	NPL Documents	F0034_rfc741.pdf	693778	no	34

Warnings:					
Information:					
38	NPL Documents	F0035_don_johnson_local_access_network.pdf	722219 0b1e1b9f00b0d9516ebcbac01084300d430988a9	no	10
Warnings:					
Information:					
39	NPL Documents	F0036_terry_managing_stored_voice.pdf	122110 cdda2eb9b5228544b25cb1f7f0320a0e3014f4ce	no	2
Warnings:					
Information:					
40	NPL Documents	F0037_terry_managing_stored_voice_ACM.pdf	1750830 313a6a81a691511ead7f77d874c86df8672d9006	no	25
Warnings:					
Information:					
41	Foreign Reference	F0038_EP-0518569.pdf	1383931 a2dd99d2892cc50be82a6efbd41afd83b88b35d3	no	22
Warnings:					
Information:					
42	Foreign Reference	F0039_EP-0559047.pdf	275980 93b58e5fb0005125b332196aadedea1aec65537a	no	6
Warnings:					
Information:					
43	Foreign Reference	F0040_EP-0597691.pdf	495599 243506e1fd4d4c7bb4d2470f21a98ffe353e7dcb	no	14
Warnings:					
Information:					
44	Foreign Reference	F0041_EP-0632672.pdf	809166 38aa972377db1b56ae391464c275090f0dc3704d	no	17
Warnings:					
Information:					
45	Foreign Reference	F0042_EP-0648038.pdf	694683 f5cb1513536ad2e32a569b908d2251ef8a4e2e86	no	12
Warnings:					
Information:					
46	Foreign Reference	F0043_EP-1379039.pdf	884338	no	18

Warnings:					
Information:					
47	Foreign Reference	F0044_EP-1379050.pdf	920398 b32becb7ba7b3ff732c8cef715cf64115ff145cd	no	19
Warnings:					
Information:					
48	NPL Documents	F0045_eve_schooler_packet-switched.pdf	704711 76ffe5c2345ee31a5637188463f5ffa7386d4a09	no	13
Warnings:					
Information:					
49	NPL Documents	F0046_gary_kessler_ISDN.pdf	361313 c106ad99e2ee3fe199744adfa22400738bac1c6e	no	7
Warnings:					
Information:					
50	Foreign Reference	F0047_GB-2283645.pdf	1449005 5ebbe51c981abfe4ff062dee24a6a45d15178141	no	42
Warnings:					
Information:					
51	NPL Documents	F0048_giulio_Barberis_Coded_Speech.pdf	873532 6a62155018dffe1c8842188f94d2782776e16a0c	no	11
Warnings:					
Information:					
52	NPL Documents	F0049_chao_packet_video_system.pdf	330220 42b66e819077341314155808b462a0b7c61e8e71	no	7
Warnings:					
Information:					
53	NPL Documents	F0050_rfc632.pdf	212561 813783ec5aabdf5f310b187e9afa6e36831790af	no	6
Warnings:					
Information:					
54	NPL Documents	F0051_henning_schulzrinne_voice_communication_across_the_internet.pdf	1685713 9c163be60757acd4173d38dd3b7475c802602120	no	34
Warnings:					
Information:					
55	NPL Documents	F0052_hiroshi_kobayashi_voice_data.pdf	617388	no	10

Warnings:					
Information:					
56	NPL Documents	F0053_hiroyuki_ichikawa_high_speed.pdf	769683 377a37e818156ffcb70429eb16d184d4bc25db4d	no	11
Warnings:					
Information:					
57	NPL Documents	F0054_ian_merritt_providing.pdf	859747 016fb76c1270ed93ac880e7579fbfea5b9de7096	no	26
Warnings:					
Information:					
58	NPL Documents	F0055_Implementation_of_Next_Generation.pdf	1968206 dd14d7a41b57ff0733650455bddf33b60f45caf83	no	15
Warnings:					
Information:					
59	NPL Documents	F0056_iper_us9615504.pdf	706094 a7766b4aac27fb19a0d7983aee898dc5915aa5e2	no	21
Warnings:					
Information:					
		Total Files Size (in bytes):	50225222		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION OF:	Attorney Docket: 2655-0188
Net2Phone, Inc. (Patent No. 6,108,704)	Group Art Unit: 3992
Control No.: 90/010,416	Examiner: KOSOWSKI, Alexander
Issue Date: August 22, 2000	Date: December 14, 2009
Title: **POINT-TO-POINT INTERNET PROTOCOL**	Confirmation No.: 1061

INFORMATION DISCLOSURE STATEMENT

Hon. Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Pursuant to 37 C.F.R. § 1.56, the attention of the Patent and Trademark Office is hereby directed to the reference(s) listed on the attached PTO-1449. One copy of each non-U.S. Patent reference is attached. It is respectfully requested that the information be expressly considered during the prosecution of this application, and that the reference(s) be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

The submission of any document herewith, which is not a statutory bar, is not intended that any such document constitutes prior art against any of the claims of the present application or is considered to be material to patentability as defined in 37 C.F.R. § 1.56(b). Applicants do not waive any rights to take any action which would be appropriate to antedate or otherwise remove as a competent reference against the claims of the present application.

CHARGE STATEMENT: Deposit Account No. 501860, order no. **2655-0188**.

The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (missing or insufficiencies only) now or hereafter relative to this application and the resulting Official Document under Rule 20, or credit any overpayment, to our Accounting/Order Nos. shown above, for which purpose a duplicate copy of this sheet is attached

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal sheet is filed.

CUSTOMER NUMBER

42624

Davidson Berquist Jackson & Gowdey LLP
4300 Wilson Blvd., 7th Floor,
Arlington Virginia 22203
Main: (703) 894-6400 • FAX: (703) 894-6430

Respectfully submitted,

By: /Michael R. Casey /

Michael R. Casey, Ph.D. (Reg. No.: 40,294)

PATENT ABSTRACTS OF JAPAN

(11)Publication number : **63-131637**
(43)Date of publication of application : **03.06.1988**

(51)Int.Cl. H04L 11/00

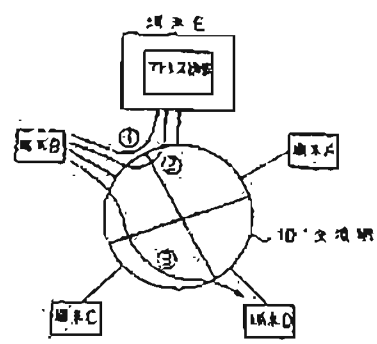
H04L 11/00

(21)Application number : **61-277727**
(22)Date of filing : **20.11.1986**

(71)Applicant : **FUJITSU LTD**
(72)Inventor : **YAHAGI TAKEHIKO**

(54) **ADDRESS MANAGEMENT SYSTEM FOR COMMUNICATION NETWORK**

(57)Abstract:
PURPOSE: To contrive the reduction of the quantity of address information provided to each terminal equipment by allowing a management terminal equipment to manage address information of a communication network altogether and allowing general terminal equipments to inquire about the information to the management terminal equipment, thereby simplifying the addition/revision of a terminal equipment address.



CONSTITUTION: For example, a terminal equipment E among lots of terminal equipments A, B... connected to an exchange network is used as the address management terminal equipment, to which address information of the all terminal equipments is given. In case of the communication by each terminal equipment, the address of a terminal equipment being an opposite party of communication is inquired about the address management terminal equipment, and the address informed from the said terminal equipment is used to make communication with the terminal equipment being the communication party. That is, when the terminal equipment B makes communication with the terminal equipment D, the terminal equipment B inquires about the address of the terminal equipment D to the equipment E, which informs the address of the terminal equipment D to the terminal equipment B, and the terminal equipment B uses the informed address to call the terminal equipment D to apply communication. Thus, terminal equipments A~D other than the equipment E do not require to have address information in this way, then the memory capacity is saved.

LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑲日本国特許庁（JP）　　⑪特許出願公開

⑫公開特許公報（A）　昭63－131637

⑤Int.Cl.⁴	識別記号	庁内整理番号	④公開　昭和63年(1988)6月3日
H 04 L 11/00	310	Z－7928－5K 7830－5K	

審査請求　未請求　発明の数 1　（全4頁）

⑤発明の名称　通信ネットワークのアドレス管理方式

②特　願　昭61－277727

②出　願　昭61(1986)11月20日

⑦発 明 者　矢 作　毅 彦　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内

⑪出 願 人　富士通株式会社　神奈川県川崎市中原区上小田中1015番地

⑭代 理 人　弁理士　青柳　稔

明　細　書

1.発明の名称

通信ネットワークのアドレス管理方式

2.特許請求の範囲

交換機（10）で接続された多数の端末（A，B，C，——）を有する通信ネットワークの各端末のアドレス管理方式において、

該多数の端末のうちの1つ（S）をアドレス管理端末としてこれに全端末のアドレス情報を持たせ、

各端末はアドレス管理端末に通信相手の端末のアドレスを問合せ、アドレス管理端末から知らされたアドレスを用いて通信相手の端末と通信することを特徴とする通信ネットワークのアドレス管理方式。

3.発明の詳細な説明

（概　要）

通信ネットワークのアドレス情報を管理端末に一括管理させ、一般端末は管理端末に問合せるようにした。

（産業上の利用分野）

本発明は、通信ネットワークの各端末のアドレス管理方式に関する。

（従来の技術）

多数の端末があるコンピュータシステムでは、各端末が相手端末のアドレスを持ち、そのアドレスを用いて所望の相手端末と通信している。第2図で説明すると、10は交換機で、これに多数の端末A，B，C，——が接続される。各端末は相手端末のアドレスを所有、管理、即ち端末Aは端末B～Eのアドレスを、端末Bは端末A，C～Eのアドレスを（以下同様）所有、管理し、相手端末と通信するときは自端末が所有、管理しているアドレスから相手端末のアドレスを知り、それを用いて通信する。

（発明が解決しようとする問題点）

この方式は、通信相手のアドレスを直ちに知り得る利点はあるものの、端末が追加されるときは

既設の全端末が所有、管理するアドレス情報に追加端末のアドレスを追加しなければならず、厄介である。多数の端末が通信網で接続されるシステムとしてはＬＡＮ（Local Area Network）があるが、ＬＡＮは端末の増設、撤去が極めて容易という特徴があり、このＬＡＮで端末追加、廃止の度に全端末のアドレス情報を更新するのではＬＡＮの便利性が損なわれてしまう。また各端末がアドレス情報を持つので、システム全体としては各端末のアドレスが何重にも重複して保管されることになり、不経済である。

本発明はかゝる点を改善し、アドレス管理の簡単化、ネットワーク構築／変更の容易化と、端末で所有する通信相手のアドレス情報量の削減を図ろうとするものである。

（問題点を解決するための手段）

第１図に示すように、本発明では交換機に接続された多数の端末Ａ，Ｂ，……の１つ、本例では端末Ｅをアドレス管理端末とし、これに全端末のアドレス情報を持たせる。そして各端末が通信する際は、通信相手の端末のアドレスをアドレス管理端末に問合せ、該端末から知らされたアドレスを用いて通信相手の端末と通信する。

例えば端末Ｂが端末Ｄと通信するときは、端末Ｂがアドレス管理端末Ｅに端末Ｄのアドレスを問合せ①、これを受けてアドレス管理端末Ｅは端末Ｄのアドレスを端末Ｂへ知らせ②、端末Ｂはこの知らされたアドレスを用いて端末Ｄを呼び出し、通信する③。

（作用）

このようにすれば、アドレス管理端末以外の端末はアドレス情報を持つ必要がないからメモリ容量の削減が可能であり、また端末の加入、廃止に伴なうアドレス情報の更新はアドレス管理端末のみがすればよいから簡単である。

（実施例）

第３図に本発明の実施例を示す。各端末Ａ，Ｂ，……はインタフェース変換部、送受信制御部などの他にアドレス管理部を持ち、こゝに自端末のＬＡＮ内アドレスとアドレス管理端末ＥのＬＡＮ内アドレスを持つ。またアドレス管理端末Ｅはインタフェース変換部、送受信制御部などの他にアドレス管理部を持ち、こゝに各端末のアドレス情報テーブルを持つ。このテーブルは各端末の論理名称Ａ，Ｂ，Ｃ，……とその端末のＬＡＮ内アドレスを対応させたものである。このテーブルは例えばシステムジェネレーション時にアドレス管理端末Ｂが各端末Ａ，Ｂ，……に自己のＬＡＮ内アドレスを送出させて作成する。

第４図に示すようにある端末例えばＡがある端末例えばＢと通信するとき①は、論理名称＝ＢからそのＬＡＮ内アドレスへの変換を要求する②。自端末が持っているアドレス管理端末のＬＡＮ内アドレスを用いて該アドレス管理端末Ｂに、論理名称＝Ｂに対応するＬＡＮ内アドレスの獲得を要求する③。アドレス管理端末Ｅはアドレス情報テーブルを検索して該端末ＢのＬＡＮ内アドレス２を得、これを端末Ａへ通知する④。端末Ａのアドレス管理部はインタフェース変換部へＢ＝２を通知する⑤。

次に第５図に示すように端末Ａは、獲得したＬＡＮ内アドレスにより通信を要求する⑥。該端末の送受信制御部は端末Ｂに通信を要求する⑦。該端末Ｂは端末Ａへ応答を返し、該端末Ａの送受信制御部はインタフェース変換部へ通信可能を通知し、該インタフェース変換部は論理名称＝Ｂとの通信可能を通知する。また端末Ｂの送受信制御部は自端末へ通信要求が来たことを通知する。これにより、端末Ａは端末Ｂのアドレスを知らないで端末Ｂとの通信が可能になる。

（発明の効果）

以上説明したように本発明によれば、各端末のアドレスはアドレス管理端末で一括管理するので、端末アドレスの追加、変更が簡単であり、個々の端末が持つアドレス情報量が低減する利点が得られる。

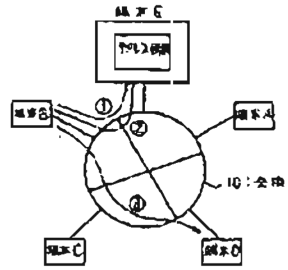
4.図面の簡単な説明

第1図は本発明の要部を示す説明図、

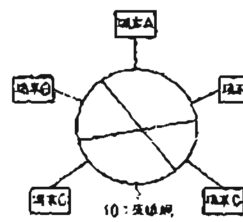
第2図は従来例の説明図、

第3図〜第5図は本発明の実施例の説明図である。

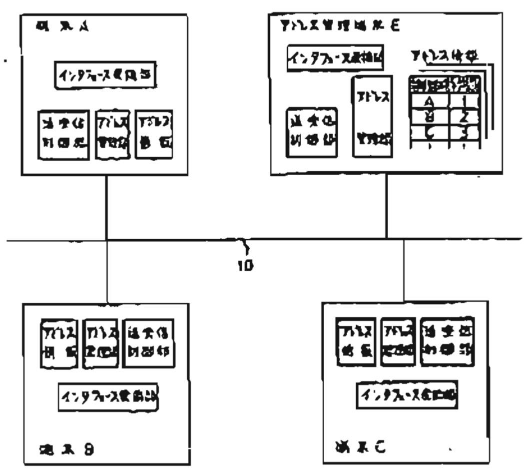
出願人　富士通株式会社
代理人弁理士　青柳　稔



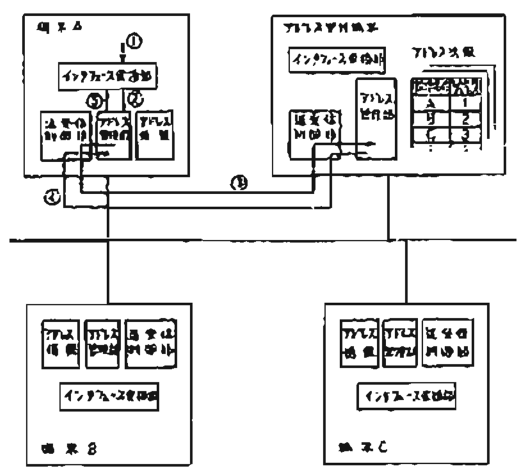
本発明の要部を示す説明図
第1図



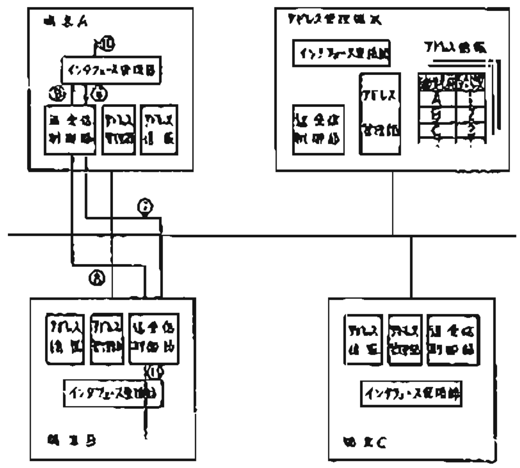
従来例の説明図
第2図



本発明の実施例の説明図
第3図



本発明の実施例の説明図
第4図



本発明の実施例の説明図
第 5 図

STATEMENT OF RELEVANCY

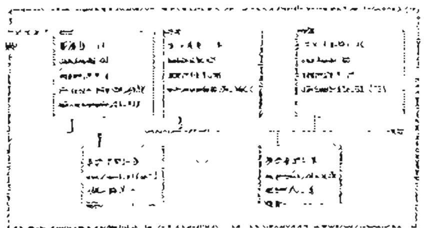
This document was cited in an Office Action corresponding to JP 2008-163825 which is a divisional application corresponding to PCT/US96/15504 (which claims priority to U.S. Patent Application Serial No. 08/533,115, now U.S. Patent No. 6,108,704).

CONSTRUCTING SYSTEM AND CHANGING SYSTEM FOR COMPUTER NETWORK

特許公報番号 JP6062020 (A)
公報発行日 1994-03-04
発明者: TERADA MASATOSHI; FUKUZAWA JUNJI; TAKASHIMA TOSHIBUMI
出願人 HITACHI LTD
分類:
一国際: G06F15/16; G08F15/177; H04L12/28; G06F15/16; H04L12/28; (IPC1-7): H04L12/28; G08F15/16
一欧州:
出願番号 JP19920231282 19920806
優先権主張番号: JP19920231282 19920806

要約 JP 8062020 (A)

PURPOSE:To provide an address and title managing function accompanied by the transfer of nodes by setting node titles in address request messages and setting the node titles indicating that they belong to other networks in the address request messages. CONSTITUTION:A network system is constituted of a domain A501 managed by a management server 502 and the respective nodes 502-505 are mutually connected by a link 506. Then, when the address request message is present from the node, the server 502 assigns an address imparted beforehand when the node title identical to the specified node title is registered and selects the address from free addresses to be assigned when it is not registered. Also, the effective period of the assigned address is decided corresponding to the length specification of the effective period of the address. Also, when the address request message in which the node title is set is present, the server 502 assigns the address from the free addresses and prepares and transmits a change information message to the management server of the present network.



(19)日本国特許庁（ＪＰ）　(12) **公開特許公報**（Ａ）　(11)特許出願公開番号

特開平6－62020

(43)公開日　平成6年(1994)3月4日

(51)Int.Cl.5	識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｈ０４Ｌ　12/28				
Ｇ０６Ｆ　15/16	４００　Ｄ	9190－5L		
		8529－5K	Ｈ０４Ｌ　11/00　　３１０　Ｚ	

審査請求　未請求　請求項の数4(全18頁)

(21)出願番号　特願平4－231282

(22)出願日　平成4年(1992)8月6日

(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目6番地

(72)発明者　寺田　真敏
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内

(72)発明者　福澤　淳二
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内

(72)発明者　高島　俊文
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内

(74)代理人　弁理士　笹岡　茂　(外1名)

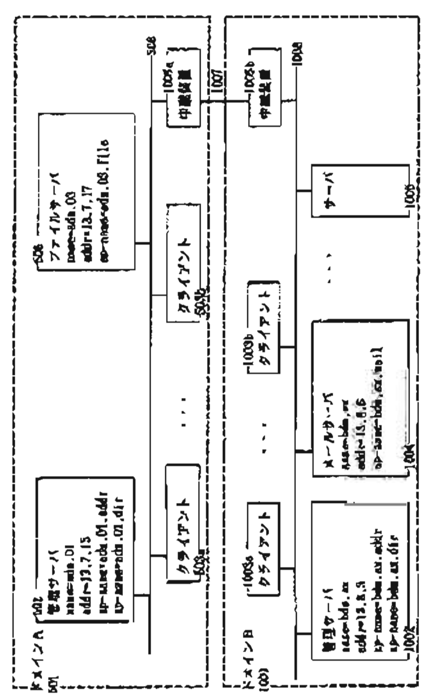
(54)【発明の名称】　コンピュータネットワーク構築方式および変更方式

(57)【要約】

【目的】　ノードの移設を考慮したアドレスおよび名称管理機能を持つコンピュータネットワーク構築方式および変更方式を提供することにある。

【構成】　ノードからのアドレス要求メッセージがあると、管理サーバはメッセージで指定されたノード名称が登録済なら既付与アドレスを割当て、未登録なら空きアドレスの中から選択して割当てる。また、メッセージ中のアドレスの有効期間の長さ指定に応じて割当てアドレスの有効期間を決定する。他ネットワークに属するノード名称を設定したアドレス要求メッセージがあると、管理サーバは、空きアドレスの中から選択してアドレスを割当て、更に現ネットワークの管理サーバへの変更情報メッセージを作成送出する。現ネットワークの管理サーバは、変更情報メッセージを受信し、変更情報に基づき接続変更されたノードに使用されていたアドレスを空きアドレスとして再使用可能にする。

【図10】



【特許請求の範囲】

【請求項1】　アドレス要求メッセージを送信し自ノードのアドレスの割当てを管理サーバに要求するノードと、アドレスを要求するノードに対してアドレスを割り当てる管理サーバを有するノードを備えるコンピュータネットワークシステムにおけるコンピュータネットワーク構築方式であって、

前記要求するノードはアドレス要求メッセージにノード名称を設定する手段を備え、

前記管理サーバは、前記アドレス要求メッセージで指定されたノード名称をキーとして自管理サーバが既に付与し現在有効であるアドレスを有するノード名称に同一のものが有るか否か調べ、有るときには既に付与したアドレスを前記要求するノードに付与し、否のときには空きアドレスの中から選択して前記要求するノードにアドレスを付与する手段を備えることを特徴とするコンピュータネットワーク構築方式。

【請求項2】　請求項1記載のコンピュータネットワーク構築方式において、前記要求するノードはアドレス要求メッセージに割当てアドレスの有効期間の長さ指定を設定する手段を備え、前記管理サーバは、前記アドレス要求メッセージで指定された有効期間の長さ指定に基づき割当てアドレスの有効期間を決定する手段を備えることを特徴とするコンピュータネットワーク構築方式。

【請求項3】　アドレス要求メッセージを送信し自ノードのアドレスの割当てを管理サーバに要求するノードと、アドレスを要求するノードに対してアドレスを割り当てる管理サーバを有するノードを備える複数のネットワークからなるコンピュータネットワークシステムにおけるコンピュータネットワーク変更方式であって、

前記要求するノードは、自ノードが接続された現ネットワークから他ネットワークに接続を変更するとき、アドレス要求メッセージに他ネットワークに属すことを示すノード名称を設定する手段を備え、

前記他ネットワークの管理サーバは、前記アドレス要求メッセージで指定されたノード名称をキーとして自管理サーバが既に付与し現在有効であるアドレスを有するノード名称に同一のものが有るか否か調べ、有るときには既に付与したアドレスを前記要求するノードに付与し、否のときには空きアドレスの中から選択して前記要求するノードにアドレスを付与する手段と前記現ネットワークの管理サーバへの変更情報メッセージを作成送出する手段を備え、

前記現ネットワークの管理サーバは、前記変更情報メッセージを受信し、該変更情報に基づき前記接続変更された要求するノードに使用されていたアドレスを空きアドレスとして再使用可能にする手段を備えることを特徴とするコンピュータネットワーク変更方式。

【請求項4】　請求項3記載のコンピュータネットワーク変更方式において、

前記現ネットワークの管理サーバは、前記変更情報に基づき前記接続変更された要求するノードに使用されていた元のノード名称に対応して変更後のノード名称を別名として登録する手段を備えることを特徴とするコンピュータネットワーク変更方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ＬＡＮ等のネットワークを用いたコンピュータシステムの計算機アドレス、名称管理方式に係り、特にシステム構築時、および構成変更時の構成変更操作の簡易化を図ることを特徴したコンピュータネットワーク構築方式および変更方式に関するものである。

【0002】

【従来の技術】従来の技術は、特開平２－２２３３６号公報に記載されているように、アドレスサーバに対してアドレス問い合わせメッセージを送信し、その応答メッセージから自計算機のアドレスを取得し、記憶するものとなっていた。

【0003】

【発明が解決しようとする課題】上述した従来の計算機アドレス管理方式では、ノード移設に伴いアドレスが変更された場合、移設前に使用していたアドレスを空きアドレスとしてリセットする手順が提供されていない。移設ノードから使用していたアドレスをメッセージの送受信処理を用いてリセットを行う場合には、移設に際し一度ノードを起動させ移設前のアドレスを空きアドレスとしてリセットさせる必要がある。また、アドレスサーバ側のローカルな操作でアドレス情報をリセットする方法もあるが、人手を介する必要があるという欠点があった。本発明の目的は、上述の点を鑑み、ノードの移設を考慮したアドレスおよび名称管理機能を持つコンピュータネットワーク構築方式および変更方式を提供することにある。

【0004】

【課題を解決するための手段】本発明のコンピュータネットワーク構築方式では、自ノードのアドレスを割当て要求するノードはアドレス要求メッセージにノード名称を設定する手段を備え、アドレスを割り当てる管理サーバは、前記アドレス要求メッセージで指定されたノード名称をキーとして自管理サーバが既に付与し現在有効であるアドレスを有するノード名称に同一のものが有るか否か調べ、有るときには既に付与したアドレスを前記要求するノードに付与し、否のときには空きアドレスの中から選択して前記要求するノードにアドレスを付与する手段を備える。また、自ノードのアドレスを割当て要求するノードはアドレス要求メッセージに割当てアドレスの有効期間の長さ指定を設定する手段を備え、管理サーバは、前記アドレス要求メッセージで指定された有効期間の長さ指定に基づき割当てアドレスの有効期間を決定

する手段を備える。本発明のコンピュータネットワーク変更方式では、自ノードのアドレスを割当て要求するノードは、自ノードが接続された現ネットワークから他ネットワークに接続を変更するとき、アドレス要求メッセージに他ネットワークに属すことを示すノード名称を設定する手段を備え、前記他ネットワークの管理サーバは、前記アドレス要求メッセージで指定されたノード名称をキーとして自管理サーバが既に付与し現在有効であるアドレスを有するノード名称に同一のものが有るか否か調べ、有るときには既に付与したアドレスを前記要求するノードに付与し、否のときには空きアドレスの中から選択して前記要求するノードにアドレスを付与する手段と前記現ネットワークの管理サーバへの変更情報メッセージを作成送出する手段を備え、前記現ネットワークの管理サーバは、前記変更情報メッセージを受信し、該変更情報に基づき前記接続変更された要求するノードに使用されていたアドレスを空きアドレスとして再使用可能にする手段を備える。また、前記現ネットワークの管理サーバは、前記変更情報に基づき前記接続変更された要求するノードに使用されていた元のノード名称に対応して変更後のノード名称を別名として登録する手段を備える。

【0005】

【作用】本発明のコンピュータネットワーク構築方式では、ノードからのアドレス要求メッセージがあると、管理サーバは、アドレス要求メッセージで指定されたノード名称と同一のものが登録されていれば、既に付与したアドレスを割当て、同一のものが登録されていなければ、空きアドレスの中から選択して割当てをする。また、アドレス要求メッセージ中のアドレスの有効期間の長さ指定に応じて割当てアドレスの有効期間を決定する。コンピュータネットワーク変更方式では、ノードからの他ネットワークに属すことを示すノード名称を設定したアドレス要求メッセージがあると、管理サーバは、空きアドレスの中から選択してアドレスを割当て、さらに現ネットワークの管理サーバへの変更情報メッセージを作成して送出する。現ネットワークの管理サーバは、変更情報メッセージを受信し、変更情報に基づき接続変更されたノードに使用されていたアドレスを空きアドレスとして再使用可能にし、さらに現ネットワークの管理サーバは、変更情報に基づき接続変更されたノードに使用されていた元のノード名称に対応して変更後のノード名称を別名として登録する。

【0006】

【実施例】以下、（1）章から（2）章において、本実施例に示すネットワークシステムが稼働する環境について説明し、（3）章から（4）章において、ネットワーク構築時の手順について説明する。

【0007】（1）用語説明

本実施例で使われる用語について説明する。

（a）管理サーバ

管理サーバはネットワークシステムのアドレス情報、ディレクトリ情報を保持管理するサーバであり、アドレスを管理するアドレス管理サーバアプリケーション（以下、アドレスサーバ）と、名称を管理するディレクトリ管理サーバアプリケーション（以下、ディレクトリサーバ）の2つのサーバ機能から構成される。なお、ここでは、管理プログラム、制御プログラムをアプリケーションと呼ぶ。

（b）ドメイン

ドメインは管理サーバの管理範囲であり、本実施例のネットワークシステムを構成するノード、利用者端末、アプリケーション、周辺装置等は管理サーバの管理下に入る。

（c）論理アドレス

論理アドレスは、ノードに割り当てられたアドレスであり、広域に渡るネットワークを構築する場合にハードウェアアドレス（例えばMAC（Media Access Control）アドレス）の仕様の違いを吸収したり、階層的なアドレス付けをすることにより宛先ノードへの経路選択を容易にすることができる。論理アドレスとしては、TCP／IP（Transmission Control Program／Internet Protocol）のIPアドレス、OSI（Open Systems Interconnection）のNSAP（Network Service Access Point）アドレス等が知られている。本実施例では、"．"で区切られた3桁の数字列を使用する。

【0008】（d）アプリケーションアドレス

アプリケーションアドレスは、一つのノード上に複数のアプリケーション（管理プログラム、制御プログラム）が存在する場合、これを識別するために使用する。本実施例では、1桁の数字列を使用する。

（e）ノード名称

ノード名称は、ノードに割り当てられた名前であり、ヒューマンリーダブルな形でノードの識別を行うために使用する。

（f）アプリケーション名称

アプリケーション名称は、アプリケーションに割り当てられた名前であり、ヒューマンリーダブルな形でアプリケーションの識別を行うために使用する。

【0009】（g）有効期間

有効期間はアドレスサーバが割当てた論理アドレスの寿命（単位：分）であり、論理アドレス取得要求者はこの期間の間、割り当てられた論理アドレスを利用することができる。また、ノード起動時には以前に割り当てられた論理アドレスを削除しなければならない。このように、割り当てた論理アドレスに寿命を持たせたり、起動時削除することにより、ノードの移設等に伴う論理アド

レスの使用取り消し操作を省いている。

（e）有効期間レベル

有効期間レベルは、アドレスサーバが割り当てる論理アドレスの有効期間に対する要求側の提案値であり、high（長期的に割当てアドレスを利用したい場合）、medium（中期的に割当てアドレスを使用したい場合）、low（短期的に割当てアドレスを利用したい場合）の3値の中から選択する。例えば、ファイルサーバとして据置型の高機能ワークステーションを使用する場合にはノードを移動させることは少ないので”high”を選択すると良い。この結果、アドレスサーバは寿命の長い有効期間を要求ノードに割り当てるため、論理アドレスの寿命経過に伴う再割当て要求処理が少なくなるとともに、論理アドレスが変更される可能性も少なくなる。また、頻繁にネットワークを移動して利用するような携帯用ワークステーションの場合には”low”を選択すると良い。この結果、アドレスサーバは寿命の短い有効期間を要求ノードに割り当てるため、移動に伴って使用されなくなる論理アドレスを有効に再利用することができる。

【0010】（2）実施例の全体構成

本実施例で対象とするネットワークシステムは以下のような特徴を備えている。

（a）本ネットワークシステムは外部記憶装置を備え、かつ比較的大きなメモリ及び計算能力を有したノード（以下、サーバ）、例えば据置型の高機能ワークステーションと、より制限されたメモリ及び計算能力しか有さないノードや外部記憶装置を持たないノード（以下、クライアント）、例えば携帯用ワークステーションから構築され、サーバがクライアントに不足する機能を補うことにより、クライアントの利用者であるアプリケーションプログラムや利用者に、より高機能で性能の良いサービスを提供する。

（b）本ネットワークシステムはドメインという考え方を用いてアドレス管理、名称管理を行う。ドメインとは管理サーバの管理範囲のことであり、このようなドメインの概念を導入することにより管理範囲の分割を行い、論理アドレスの自動割り当て、論理アドレスや名称の変更に伴うデータベースの自動更新を実現する。

（c）本ネットワークシステムはノード相互の情報転送形式は、宛先が設定されたメッセージという形で転送し、各ノードはメッセージに設定された宛先と、自ノード内のルート（宛先指定）テーブルに基づき受信メッセージの中継を行い、メッセージを正しい宛先にまで届ける機能を持っている。

【0011】次に本ネットワークシステムの構成について説明する。図5は、本ネットワークシステム構成の一例を示す図である。本ネットワークシステムは、管理サーバ502によって管理される一つのドメインA（501）から構成されており、ドメインA内の各ノードはリンク506により相互に接続されている。管理サーバ502はドメインA内の論理アドレスと名称を管理するノードであり、ノード名称”name＝adm．01”、論理アドレス”addr＝13．7．15”が割り当てられており、アプリケーション名称”ap－name＝adm．01．addr”という名称を持つアドレスサーバと、アプリケーション名称”ap－name＝adm．01．dir”という名称を持つディレクトリサーバが稼働する。また、メールサーバ504はメールサービスを提供するノードであり、このノードにはノード名称”name＝adm．02”が割り当てられており、アプリケーション名称として”ap－name＝adm．02．mail”を持つメールサーバアプリケーションが稼働する。ファイルサーバ505はファイル登録参照サービスを提供するノードであり、このノードにはノード名称”name＝adm．03”が割り当てられており、”アプリケーション名称としてap－name＝adm．03．file”を持つファイルサーバアプリケーションが稼働する。クライアント503a，bはドメインA内に存在するサーバの利用者であり、それぞれノード名称”name＝adm．client1”，”name＝adm．client2”が割り当てられている。尚、メールサーバ、ファイルサーバ、およびクライアントの各ノードの論理アドレスについては、ネットワークシステム構築時に管理サーバ502により動的に割当てが行われる。

【0012】次に図14を用いて、ネットワークシステムを構築する際に使用する名称体系について説明する。図14は、図5、及び後述する図10に示すネットワークシステムを構築する際に使用する名称体系を示す。”／”（1401）はドメインA（501）とドメインB（1001）を取り纏める上位の名称である。”／”の直下の名称はドメインに一意に割り当てられる名称であり、”adm”（1402）はドメインAを示し、”bdm”（1403）はドメインBを示す。ドメイン名称の直下の名称はノードに一意に割り当てられる名称（1404，1405）であり、ドメイン名称と連結することによりノード名称が作られる。従って、ドメインAの管理サーバ502のノード名称は”adm．01”、メールサーバ504のノード名称は”adm．02”、ファイルサーバ505のノード名称は”adm．03”となる。同様にドメインBの管理サーバ1002のノード名称は”bdm．ax”、メールサーバ504のノード名称は”bdm．sx”となる。さらにその直下の名称は各ノード上のアプリケーションに一意に割り当てられる名称（1406，1407）であり、ノード名称と連結することによりアプリケーション名称が作られる。従って、ドメインAのアドレスサーバアプリケーションのアプリケーション名称は”adm．01．addr”、ディレクトリサーバアプリケーションのアプリケーショ

ン名称は”ａｄｍ．０１．ｄｉｒ”、メールサーバアプリケーションのアプリケーション名称は”ａｄｍ．０２．ｍａｉｌ”、ファイルサーバアプリケーションのアプリケーション名称は”ａｄｍ．０３．ｆｉｌｅ”となる。同様にドメインＢのアドレスサーバアプリケーションのアプリケーション名称は”ｂｄｍ．ａｘ．ａｄｄｒ”、ディレクトリサーバアプリケーションのアプリケーション名称は”ｂｄｍ．ａｘ．ｄｉｒ”、メールサーバアプリケーションのアプリケーション名称は”ｂｄｍ．ｓｘ．ｍａｉｌ”となる。

【００１３】図４はノードのハードウェア構成であり、図２はハードウェア構成例に従って作られたメールサーバ、ファイルサーバ及び、クライアントノードのソフトウェア構成例を示す。４０３はノード間を接続する回線の入出力等を制御する回線接続制御部、４０５は情報を外部記憶装置４０６に保存するために、外部記憶装置との入出力等を制御する制御部、４０７はキーボードからの入力、ディスプレイへの表示を制御する制御部であり、これらの制御部の中にそれぞれ存在するプロセサとプログラムとにより接続する装置の制御を行う。４０４はプログラムの他、アドレス情報及び受信メッセージを格納するメモリである。４０２はプロセサであり、ノード内ハードウェア間の制御を行う。４０４はメモリ、４０８はディスプレイ、キーボードである。

【００１４】２０２はメールサーバ、ファイルサーバアプリケーション、及びこれらアプリケーションにアクセスするためのプログラムである。２０３は管理サーバに論理アドレスを問合せ、応答を受信するためのプログラムである。２０４は装置の入出力を制御するリンク制御プログラムであり回線制御部４０３、外部記憶装置制御部４０５、端末入出力制御部４０７に置かれる。プログラムスケジューラ２０５は、２０３から２０４までのプログラム実行のスケジューリングと管理を行う。

【００１５】図３はハードウェア構成例に従って作られた管理サーバノードのソフトウェア構成例を示す。アドレスサーバ３０２は、ドメインＡ内の論理アドレス情報を管理する。ディレクトリサーバ３０３は、ドメインＡ内の名称情報を管理し、名称から論理アドレスやＡＰアドレスを導き、これらのアドレス情報をクライアントに提供する。リンク制御部３０４は、装置の入出力を制御するリンク制御プログラムであり、プログラムスケジューラ３０５は、３０２から３０４までのプログラム実行のスケジューリングと管理を行う。

【００１６】図６に、論理アドレスの取得、応答に使用するメッセージ形式を示す。（ａ）は、論理アドレスを取得するためにアドレスサーバに転送される要求メッセージである。ＬＩ６０１ａはメッセージの長さフィールドであり、タイプ６０２ａは本メッセージＡＤＤＲｒｅｑ（Ａｄｄｒｅｓｓ ｒｅｑｕｅｓｔ）が論理アドレスを取得するために転送される情報であることを示し、アドレスサーバはこの情報を調べ要求メッセージを取り込む。ノード名称６０３ａは送信元ノード名称を設定するフィールドであり、その名称は送信元ノードの利用者が付与する。有効期間レベル６０４ａはアドレスサーバが割り当てる論理アドレスの有効期間に対する送信側の提案値である。履歴情報６０５ａは、ノード名称の変更が発生した場合に使用するフィールドである。（ｂ）は、論理アドレスを割り当てるためにアドレスサーバが送信する応答メッセージである。タイプ６０２ｂは本メッセージＡＤＤＲｒｅｓｐ（Ａｄｄｒｅｓｓ ｒｅｓｐｏｎｓｅ）が割り当てた論理アドレスを通知するために転送される情報であることを示し、ノード名称６０３ｂは宛先ノード名称が設定され、論理アドレス６０６にはアドレスサーバが割り当てた論理アドレスが格納される。また、有効期間６０７は割当てた論理アドレスの寿命であり、アドレス取得要求者はこの期間（単位：分）の論理アドレスを利用することができる。（ｃ）は、ノード管理情報の再設定を行なう際にアドレスサーバに転送される要求メッセージであり、内容は（ａ）と同様である。これについては、後述の「（４−２）論理アドレスの取得」の項でさらに説明する。

【００１７】（３）ネットワークの新規構築
図５に示すネットワークシステムにクライアント５０３ａを新規に導入する場合について説明する。

【００１８】（３−１）ノード管理情報の設定
図１はノード管理情報の設定手順を示すフローである。ステップ１０２はノード名称の設定を確認する操作であり、クライアント５０３ａは新規導入のためノード名称が未設定なので、ステップ１０３でノード名称”ａｄｍ．ｃｌｉｅｎｔ１”を設定する。既にノード名称が設定されており、これを変更する場合にはステップ１０４を実行する。ステップ１０６ではノード名称の変更を確認し、変更があれば前回設定されていたノード名称を履歴情報として登録し（１０７）、なければ現在設定されているノード名称を履歴情報として登録する（１０８）。ステップ１０９は有効期間レベルの設定を確認する操作であり、クライアント５０３ａは新規導入のため有効期間レベルが未設定なので、ステップ１１０で有効期間レベルを設定するが、クライアント５０３ａがネットワークを移動して利用するような携帯用ワークステーションであることから、ステップ１１０の有効期間レベルとして”ｌｏｗ”を設定する。以降、クライアント５０３ａのノード起動時は、ここで設定されたノード名称、有効期間レベルに基づいてアドレス取得手順が実施される。既に有効期間レベルが設定されており、これを変更する場合にはステップ１１１を実行する。なお、フローにおいて、“→(１)”は“←(１)”への分岐、“→(２)”は“←(２)”への分岐を示す。

【００１９】次に、クライアント５０３ａにおけるアドレス取得手順を図７に従い説明する。

（3－2）論理アドレスの取得
図7は論理アドレス取得手順を示すフローである。ステップ702において図6の（a）に示す要求メッセージのノード名称フィールド603aにノード名称”adm．client1”を設定、ステップ703において有効期間レベル”low”をフィールド604aに設定、ステップ704において履歴情報”adm．client1”をフィールド605aに設定した後、要求メッセージを送信する（705）。次にステップ706で取得要求の応答を待ち、応答を受信できた場合には応答メッセージの論理アドレスのチェックを行い（707）、応答を受信できなかった場合には再度論理アドレスの取得手順を実行する（711d）。ステップ707で受信論理アドレス値の確認を行い、”null”ならば再度論理アドレスの取得手順（711c）を実行し、それ以外ならば受信論理アドレスを登録後（708）、有効期間の減算手順にはいる（710）。また、有効期間の減算とともに、ステップ709，711bで現在使用している論理アドレスを継続するために、有効期間が切れる前に再度アドレス取得手順を実行する。なお、本実施例では全ての有効期間値を減算の対象としているが、ある特定の有効期間値（例えば有効期間値＝”9999”）を持つ論理アドレスについては、有効期間の減算及び、起動時の論理アドレス削除の対象からはずすことにより、一度割り当てられた論理アドレスを有効期間という時間的な制約無しに利用することも可能である。

【0020】（3－3）論理アドレスの割当て
図8は管理サーバ502内のアドレスサーバにおける論理アドレスの割当て手順を示すフローである。ステップ802において要求メッセージを受信すると、ステップ803において要求メッセージのノード名称フィールド603aに設定されたノード名称を参照する。ステップ803はノード名称のみを変更した場合に、論理アドレスを極力変更させないための処理であり、指定されノード名称の論理アドレスが、自アドレスサーバによって割り当てられたものであり、さらにその論理アドレスの割り当て有効期間が過ぎていない場合には、前回割り当てた論理アドレスと同一の論理アドレスを割り当てることにより、論理アドレス値の変更を極力抑えている。クライアント503aは新規導入なので、アドレスサーバはステップ804の処理を行わず、ステップ805で空きアドレスを探し、論理アドレスとして使用されていない論理アドレス”13．7．4”を割当て（806）、応答メッセージの論理アドレスフィールド606に”13．7．4”を設定し、ステップ809で提案された有効期間レベルに従い有効期間をフィールド607に設定した後、応答メッセージを論理アドレス取得要求ノードに送信する（810）。また、ステップ808では他のアドレスサーバ、ディレクトリサーバに変更情報を通知する。

【0021】図9に管理サーバ502が管理するアドレス情報、ディレクトリ情報を示す。アドレス情報901はアドレスサーバのデータベースであり、｛ノード名称902a，割り当て論理アドレス903a，残り有効期間904a｝の3項目から構成され、902a，903a，904aはクライアント503aに関するエントリ、902b，903b，904bはメールサーバ504に関するエントリ、902c，903c．904cはファイルサーバ505に関するエントリとなっている。ノード名称902aは論理アドレス取得要求ノードの名称、論理アドレス903aはアドレスサーバによって割り当てられた論理アドレス値、残り有効期間904aはアドレスサーバが割り当てた論理アドレスの有効期間の残り時間である。アドレスサーバにおいてもステップ709に示すような有効期間の減算手順が実行され、残り有効期間がなくなった時点でこの論理アドレスは空きアドレスとして再利用する。ディレクトリ情報905はディレクトリサーバのデータベースであり、｛エントリ名称906a，論理アドレス907a，APアドレス908a，別名909a｝の4項目から構成され、906a，907a，908a，909aはメールサーバ504に関するエントリ、906b，907b，908b，909bはファイルサーバ505に関するエントリとなっている。エントリ名称906aはディレクトリ情報として登録されるアプリケーション名称であり、論理アドレス907a、APアドレス908a、別名909aはその属性値である。

【0022】（4）ネットワークの再構成
図5に示すネットワークシステムを図10に示すネットワークシステムに構成を変更する場合について説明する。図10は、図5に示すネットワークシステムを拡張した構成の一例を示す図である。ドメインA（501）は管理サーバ502によって管理される一つの領域であり既存のネットワークシステムに相当する。ドメインB（1001）は管理サーバ1002によって管理される領域であり、新たに追加された領域である。ドメインB内の各ノードはリンク1008により相互に接続され、さらに中継装置1006a，1006bとリンク1007を用いてドメインAとの相互接続を可能としている。管理サーバ1002はドメインB内の論理アドレスと名称を管理するノードであり、利用者によってノード名称”name＝bdm．ax”、論理アドレス”addr＝13．8．3”が割り当てられており、アプリケーション名称として”ap－name＝bdm．ax．addr”を持つアドレスサーバと、アプリケーション名称として”ap－name＝bdm．ax．dir”を持つディレクトリサーバが稼働する。メールサーバ1004は、図5のドメインAに置かれていたメールサーバ504を今回の再構成によりドメインB内に移設したものであり、ドメインBへの移設に伴いノード名称が”n

ame＝bdm．sx”に、アプリケーション名称が”ap－name＝bdm．sx．mail”に名称が変更された。この他、ドメインB内には、各種サーバ1005、クライアント1003a，1003bが存在する。

【0023】以下、メールサーバ504をドメインAからドメインBへ移設する場合の処理手順について説明する。

（4－1）ノード管理情報の再設定

図1に従いメールサーバノード管理情報の再設定手順を説明する。移設に伴いノード名称が変更されたのでステップ104でノード名称”bdm．sx”を設定し、ステップ106でノード名称の変更を確認後、ステップ107で前回設定されていたノード名称”adm．02”を履歴情報として登録する。また、有効期間については前回の設定値”high”を使用するので、ステップ109から112の設定処理は行わない。

（4－2）論理アドレスの取得

図7に従いメールサーバノードの論理アドレス再取得手順について説明する。ステップ702において要求メッセージのノード名称フィールド603cにノード名称”bdm．sx”を設定、ステップ703において有効期間レベル”high”をフィールド604cに設定、ステップ704において履歴情報”adm．02”をフィールド605cに設定した後、要求メッセージを送信する（705）。なお、以降の論理アドレスの取得手順については、項番（3－2）のクライアント503aの場合と同一の手順を踏むので説明を省略する。

（4－3）論理アドレスの割当てと更新

図8に従い管理サーバ1002内のアドレスサーバによるメールサーバノードの論理アドレス割当て手順について説明する。テップ802において要求メッセージを受信すると、ステップ803において要求メッセージのノード名称を参照する。メールサーバ1004はドメインBにおいて新規導入のため、ステップ805で空きアドレスを探し、論理アドレスとして使用されていない論理アドレス”13．8．6”を割当て、図6の（b）と同様の応答メッセージの論理アドレスフィールド606に”13．8．6”を設定し、ステップ809で提案された有効期間レベルに従い有効期間”9999”をフィールド607に設定した後、応答メッセージを論理アドレス取得要求ノードに送信する（810）。また、これに合わせてステップ808で他ドメインのアドレスサーバ、ディレクトリサーバに変更情報メッセージを通知する。

【0024】図11は、前記ステップ808で通知する変更情報を他ドメインのアドレスサーバ、自ドメイン及び他ドメインのディレクトリサーバに通知するための変更情報メッセージを示す。LI1101はメッセージの長さフィールドであり、タイプ1102は本メッセージUPDATE（Address Update）が変更情報をアドレスサーバ、ディレクトリサーバに通知するための情報であることを示す。ドメインAのアドレスサーバに変更情報を通知する場合には、宛先サーバ名称1103にはドメインAのアドレスサーバ名称を設定し、送信元サーバ名称1104には自アドレスサーバ名称を設定する。変更情報1108には、アドレス取得要求ノードのノード名称1105として”bdm．sx”、履歴情報1106として”adm．02”、論理アドレス1107として”13．8．6”を設定する。また、ドメインAのディレクトリサーバに変更情報を通知する場合には、宛先サーバ名称1103にはドメインAのディレクトリサーバ名称を設定する。

【0025】今回のメールサーバ504の移設に伴い、この変更情報メッセージは、管理サーバ1002のアドレスサーバから、管理サーバ502のアドレスサーバ、ディレクトリサーバに通知される。変更情報メッセージを受信したドメインAのアドレスサーバは、履歴情報1106をキーとして、ドメインA内のメールサーバ504に使用されていたアドレスエントリを捜し出したのちリフレッシュを行い、使用されていた論理アドレスを空き論理アドレスとして再使用可能な状態とする。

【0026】図12に、メールサーバ504移設に伴う論理アドレス、更新処理が完了した時点での管理サーバ502（ドメインA）のアドレスサーバ（502－1）と管理サーバ1002（ドメインB）のアドレスサーバ（1002－1）が管理するアドレス情報（1201，1211）を示す。1202b，1203b，1204bは管理サーバ1002（ドメインB）のアドレス情報として格納されているドメインB内のメールサーバ504に関するエントリであり、メールサーバ504がドメインBに移設され、論理アドレスの取得処理が行われた際に登録されたものである。1202a，1203a，1204aは管理サーバ502（ドメインA）のアドレス情報として、メールサーバ504がドメインAに存在していた時に使用されていたエントリであり、変更情報受信時にリフレッシュされ、現時点では空きエントリとなっている。

【0027】次に変更情報更新後の管理サーバのディレクトリ情報について説明する。本ネットワークシステムにおけるディレクトリサーバアプリケーションは、ISO9594に規定されるOSIディレクトリに従う問い合わせ、登録情報の更新の他に、アドレスサーバとの間で登録情報の更新手順を持つ。図13に、メールサーバ504移設に伴う論理アドレス、更新処理が完了した時点での管理サーバ502（ドメインA）のディレクトリサーバ（502－2）と管理サーバ1002（ドメインB）のディレクトリサーバ（1002－2）が管理するディレクトリ情報（1301，1311）を示す。変更情報メッセージを受信したドメインAのディレクトリサ

ーバは、図１１に示す履歴情報１１０６をキーとして、ドメインＡ内のメールサーバ５０４に使用されていたディレクトリエントリ”ａｄｍ．０２．ｍａｉｌ”を捜し出したのち、図１１に示すノード名称１１０５からドメインＢに移設されたメールサーバアプリケーション名称”ｂｄｍ．ｓｘ．ｍａｉｌ”を生成する。次にドメインＡのディレクトリサーバは、図１１に示す変更情報として通知された論理アドレス”１３．８．６”（１１０７）と、ディレクトリ情報として格納されていたアプリケーションアドレス”５”（９０８ｂ）を持つエントリ”ｂｄｍ．ｓｘ．ｍａｉｌ”（１３０６ｂ，１３０７ｂ，１３０８ｂ，１３０９ｂ）の生成をドメインＢのディレクトリサーバに要求するとともに、ドメインＡ内のディレクトリ情報の１３０７ａ，１３０８ａをリフレッシュし、別名１３０９ａには移設されたメールサーバのアプリケーション名称”ｂｄｍ．ｓｘ．ｍａｉｌ”を登録する。以上のような処理を経て、ネットワーク再構成時の論理アドレスの割当て、名称の変更処理が終了する。

【００２８】

【発明の効果】１つのノードに複数のアドレスの割当てを防止でき、また、有効期間付きのアドレス割当てを行なうことができる。また、ノード移設による論理アドレスの変更に伴い、移設前に使用していた論理アドレスは、新たに論理アドレスの割当てが行われた時点で空き論理アドレスとなるため、論理アドレスを効率良く利用することができる。さらに、ノード移設に伴いノード名称が変更された場合にも、旧ノード名称に対応して新名称が登録されているので、旧ノード名称でのアクセスが可能である。

【図面の簡単な説明】

【図１】ノード管理情報の設定手順のフローを示す図である。

【図２】ハードウェア構成例に従って作られたメールサーバ、ファイルサーバ及び、クライアントノードのソフトウェア構成例を示す図である。

【図３】ハードウェア構成例に従って作られた管理サーバノードのソフトウェア構成例を示す図である。

【図４】ノードのハードウェア構成を示す図である。

【図５】本ネットワークシステム構成の一例を示す図である。

【図６】論理アドレスの取得、応答に使用するメッセージ形式を示す図である。

【図７】論理アドレス取得手順のフローを示す図である。

【図８】管理サーバ内のアドレスサーバにおける論理アドレスの割当て手順のフローを示す図である。

【図９】管理サーバが管理するアドレス情報、ディレクトリ情報を示す図である。

【図１０】図５に示すネットワークシステムを拡張した構成の一例を示す図である。

【図１１】変更情報メッセージの形式を示す図である。

【図１２】メールサーバ移設に伴う処理終了時点でのドメインＡの管理サーバのアドレスサーバとドメインＢの管理サーバのアドレスサーバが管理するアドレス情報を示す図である。

【図１３】メールサーバ移設に伴う処理終了時点でのドメインＡの管理サーバのディレクトリサーバとドメインＢの管理サーバのディレクトリサーバが管理するディレクトリ情報を示す図である。

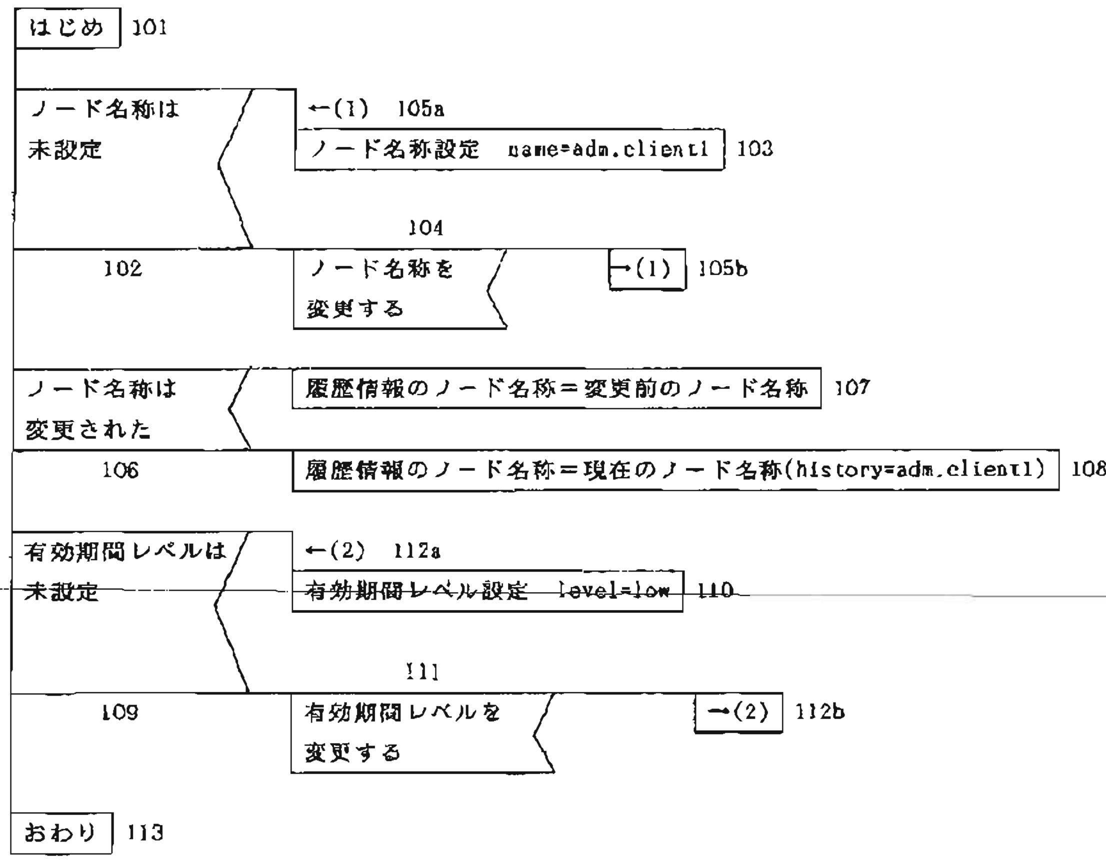
【図１４】ネットワークシステムを構築する際に使用する名称体系を示す図である。

【符号の説明】

４０２　プロセサ
４０３　回線制御部
４０５　外部記憶装置制御部
４０６　外部記憶装置
４０７　端末入出力制御部
４０８　ディスプレイ、キーボード
５０２、１００２　管理サーバ
５０３ａ、５０３ｂ、１００３ａ、１００３ｂ　クライアント
５０４、１００４　メールサーバ
５０５　ファイルサーバ
１００６ａ、１００６ｂ　中継装置
１００５　サーバ

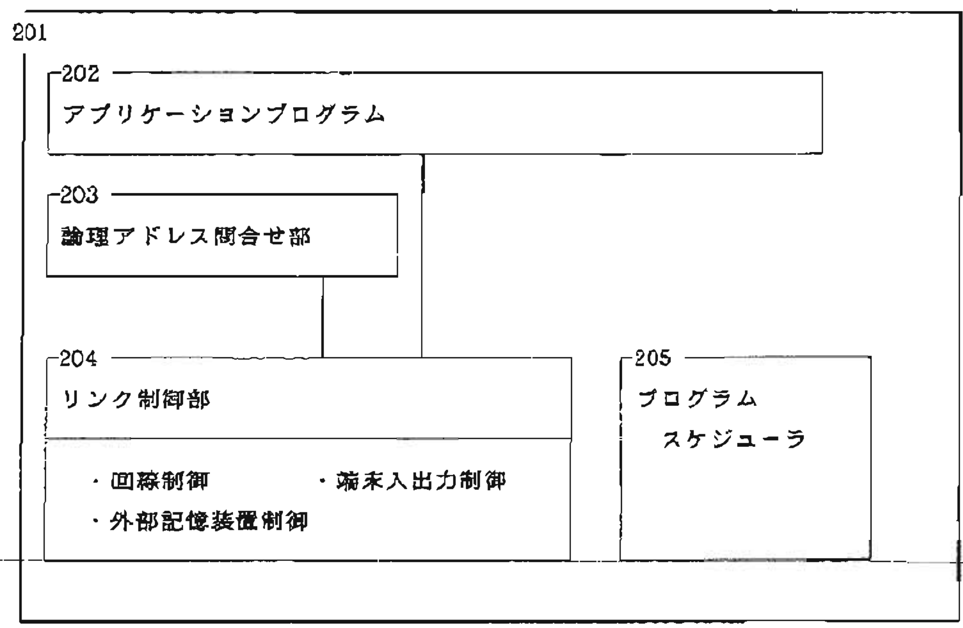
【図1】

【図1】


はじめ
101
ノード名称は
未設定
←(1)
105a
ノード名称設定 name=adm.client1
103
104
102
ノード名称を
変更する
→(1)
105b
ノード名称は
変更された
履歴情報のノード名称＝変更前のノード名称
107
106
履歴情報のノード名称＝現在のノード名称(history=adm.client1)
108
有効期間レベルは
未設定
←(2)
112a
有効期間レベル設定 level=low
110
111
109
有効期間レベルを
変更する
→(2)
112b
おわり
113

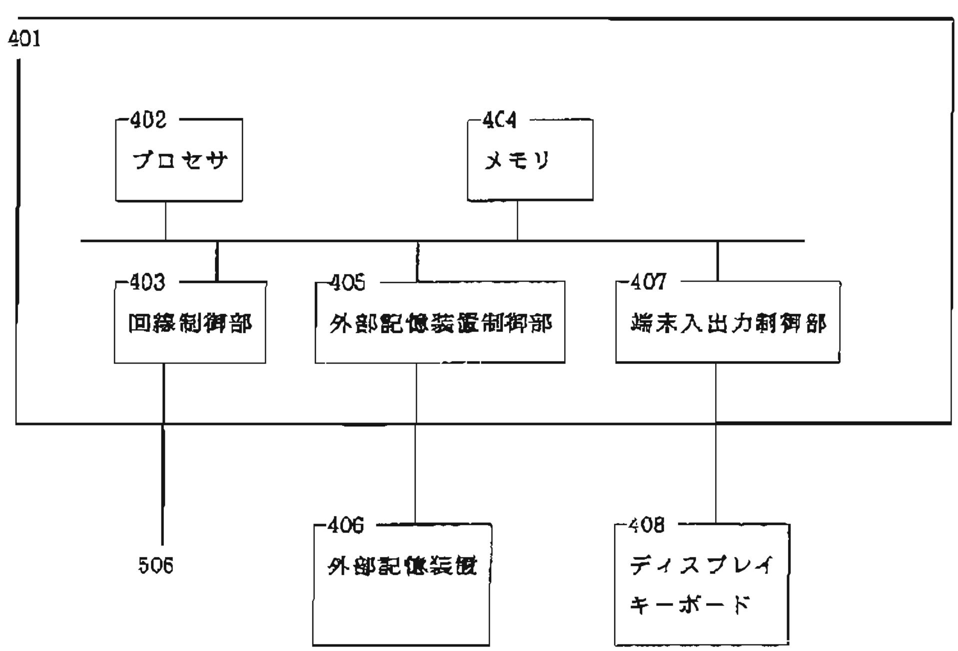
【図2】

【図2】


201
202
アプリケーションプログラム
203
論理アドレス問合せ部
204
リンク制御部
・回線制御
・端末入出力制御
・外部記憶装置制御
205
プログラム
スケジューラ

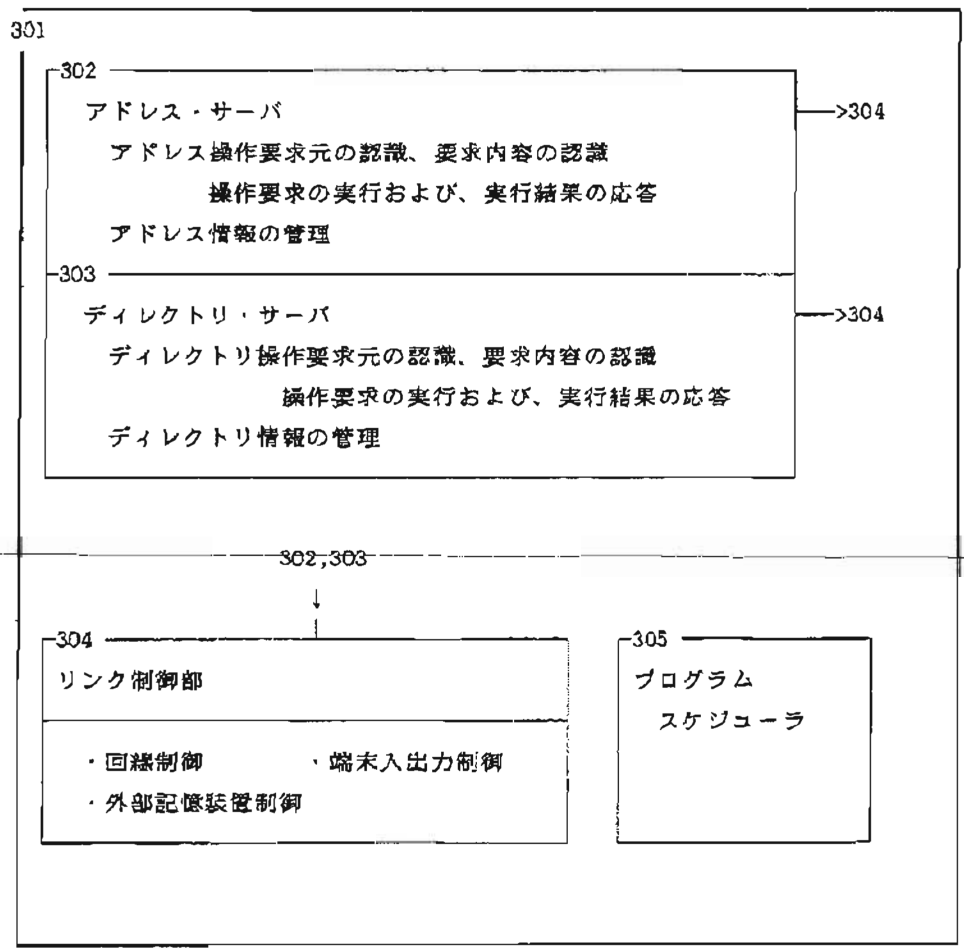
【図4】

【図4】


401
402
プロセサ
404
メモリ
403
回線制御部
405
外部記憶装置制御部
407
端末入出力制御部
506
406
外部記憶装置
408
ディスプレイ
キーボード

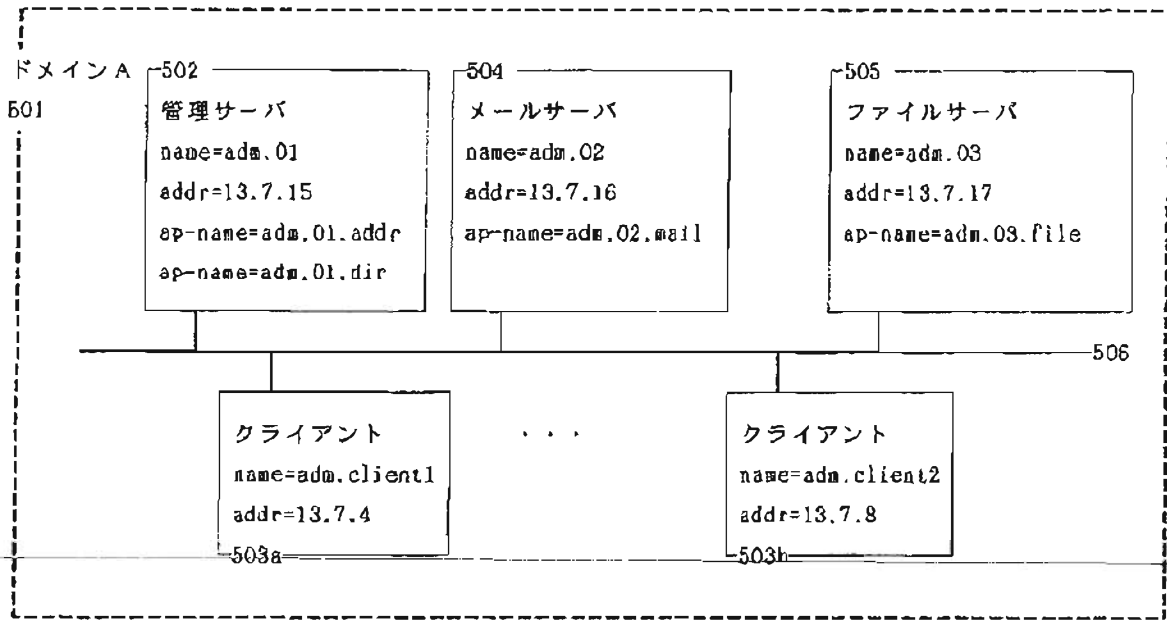
【図3】

［図3］


301
302
アドレス・サーバ
アドレス操作要求元の認識、要求内容の認識
操作要求の実行および、実行結果の応答
アドレス情報の管理
>304
303
ディレクトリ・サーバ
ディレクトリ操作要求元の認識、要求内容の認識
操作要求の実行および、実行結果の応答
ディレクトリ情報の管理
>304
302,303
304
リンク制御部
・回線制御
・端末入出力制御
・外部記憶装置制御
305
プログラム
スケジューラ

【図5】

【図5】



【図6】

【図6】

(a)

LI	タイプ	ノード名称	有効期間レベル	履歴情報
	ADDRreq	adm.client1	low	adm.client1
601a	602a	603a	604a	605a

(b)

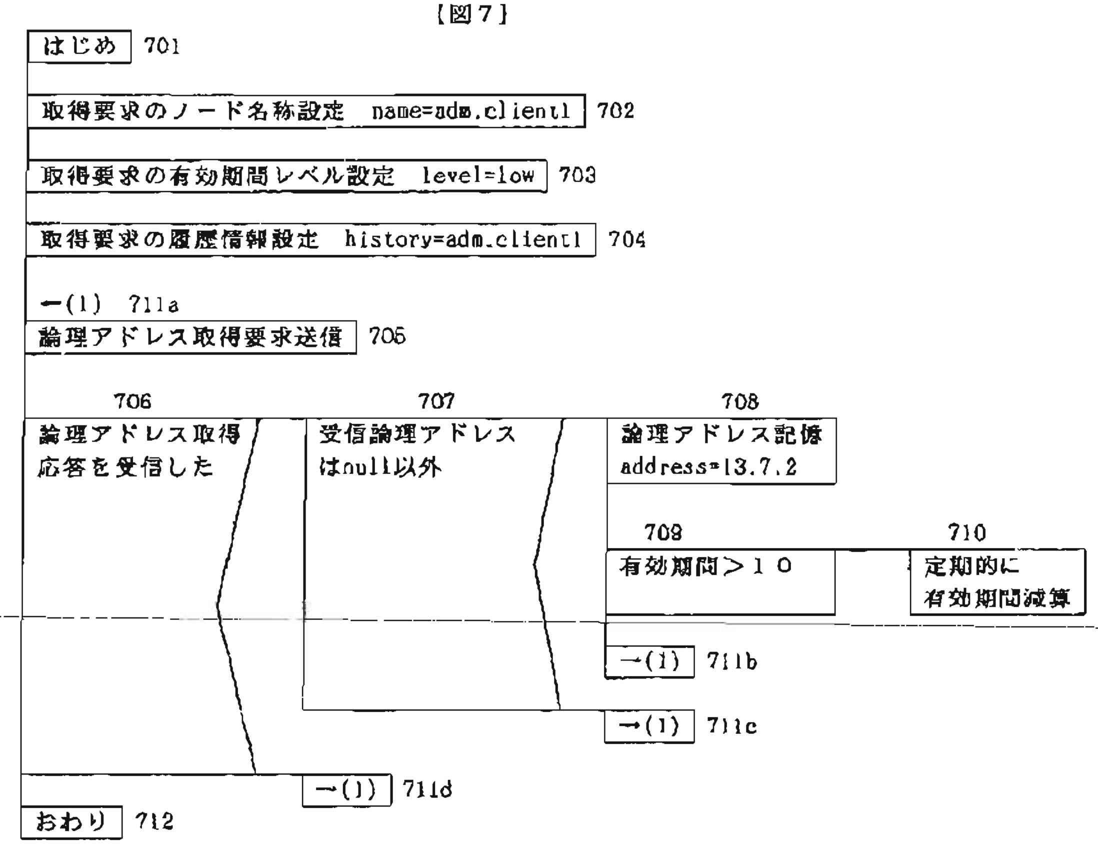
LI	タイプ	ノード名称	論理アドレス	有効期間
	ADDRresp	adm.client1	13.7.4	1000
601b	602b	603b	606	607

(c)

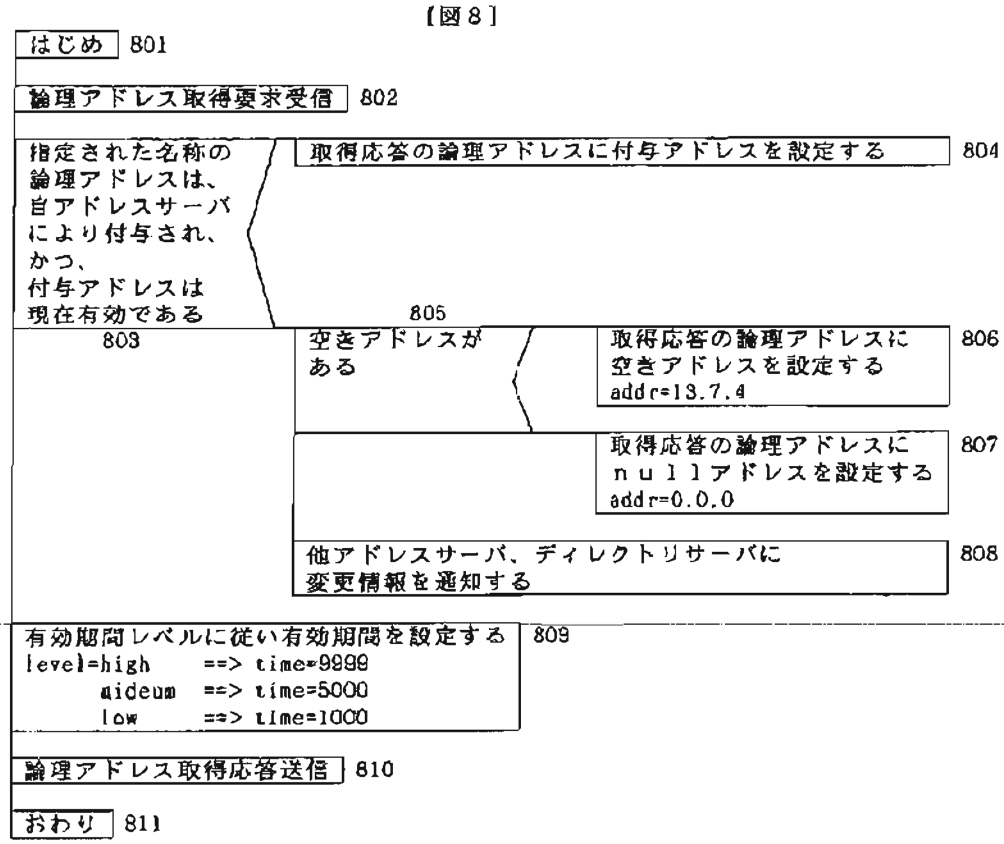
LI	タイプ	ノード名称	有効期間レベル	履歴情報
	ADDRreq	bdm.sx	high	adm.02
601c	602c	603c	604c	605c

【図7】

【図7】

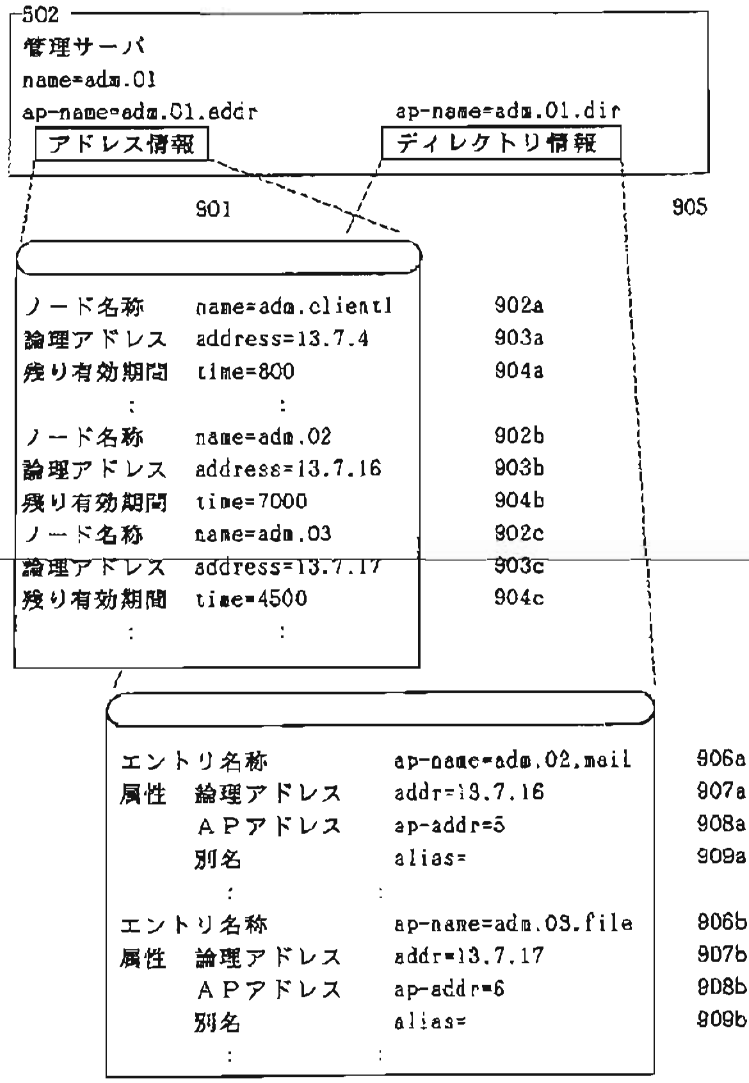

はじめ 701
取得要求のノード名称設定 name=adm.client1 702
取得要求の有効期間レベル設定 level=low 703
取得要求の履歴情報設定 history=adm.client1 704
←(1) 711a
論理アドレス取得要求送信 705
706
論理アドレス取得応答を受信した
707
受信論理アドレスはnull以外
708
論理アドレス記憶 address=13.7.2
709
有効期間＞１０
710
定期的に有効期間減算
→(1) 711b
→(1) 711c
→(1) 711d
おわり 712

【図8】


【図8】
はじめ 801
論理アドレス取得要求受信 802
指定された名称の
論理アドレスは、
自アドレスサーバ
により付与され、
かつ、
付与アドレスは
現在有効である
803
取得応答の論理アドレスに付与アドレスを設定する 804
805
空きアドレスが
ある
取得応答の論理アドレスに
空きアドレスを設定する
addr=13.7.4
806
取得応答の論理アドレスに
ｎｕｌｌアドレスを設定する
addr=0.0.0
807
他アドレスサーバ、ディレクトリサーバに
変更情報を通知する
808
有効期間レベルに従い有効期間を設定する 809
level=high ==> time=9999
mideum ==> time=5000
low ==> time=1000
論理アドレス取得応答送信 810
おわり 811

【図9】

【図9】


502
管理サーバ
name=adm.01
ap-name=adm.01.addr
ap-name=adm.01.dir
アドレス情報
ディレクトリ情報
901
905
ノード名称 name=adm.client1 902a
論理アドレス address=13.7.4 903a
残り有効期間 time=800 904a
ノード名称 name=adm.02 902b
論理アドレス address=13.7.16 903b
残り有効期間 time=7000 904b
ノード名称 name=adm.03 902c
論理アドレス address=13.7.17 903c
残り有効期間 time=4500 904c
エントリ名称 ap-name=adm.02.mail 906a
属性 論理アドレス addr=13.7.16 907a
ＡＰアドレス ap-addr=5 908a
別名 alias= 909a
エントリ名称 ap-name=adm.03.file 906b
属性 論理アドレス addr=13.7.17 907b
ＡＰアドレス ap-addr=6 908b
別名 alias= 909b