Commercialization of Electronic Information

JEAN-HENRY MORIN, University of Geneva, Switzerland
DIMITRI KONSTANTAS, University of Geneva, Switzerland

Information dissemination is slowly moving from printed media to electronic media. However this step cannot be completed if the electronic commercialization of information does not provide the same guarantees against copyright infringement as with the printed media. In this paper we present the major requirement for the commercialization of electronic information and describe Hep, an agent-based framework we developed for the commercialization of arbitrary electronic documents over open networks. The Hep electronic document commercialization model follows the secure content encapsulation model and regards documents as programs (agents) that need to be executed in order to reveal their contents. This way the document provider can include arbitrary checks and controls against possible copyright infringement attempts.[1]

One of the most valuable commodities in today's world is *information*. Business, as well as private persons, purchase every day information provided in different forms, ranging from newspapers to highly specialized business reports, from information video clips to music logos, and from free advertisement to expensive commercial updates. The medium however that dominates the dissemination of information is paper. The majority of information is disseminated in the form of printed documents, from leaflets to magazines and business reports. On the other hand, the vast majority of printed information is prepared using electronic means (i.e., computers). However only a small percentage of the (electronically prepared) commercial information is commercialized in electronic form. For many reasons information providers are very reluctant to commercialize their (valuable) information in electronic form.

Their prudence is well motivated by a number of reasons which become clear if we compare the traditional commercialization of printed documents with the commercialization of their electronic counterparts. With printed documents there are practical limitations to copying, modifying or redistributing a document. For example, it is very expensive, if not impossible, to modify the contents of a printed document without leaving traces, whereas copying a book is in general more expensive than buying a new copy. In addition, even if the printed document is passed from one person to another, there is still a single copy of it. Electronic documents on the other hand can be easily modified, copied, and distributed to many individuals without the original owner losing his own copy and without the need to inform or even get authorization from the publisher of the document. Although techniques exist allowing one to control the integrity of a document and authenticate its publisher in an unambiguous way, the major problem lies in the ability to create and distribute unauthorized copies of the document at virtually no cost. This is eventually the major reason for which commercial intellectual work like books seldom appear in electronic form.

In order to commercialize information in electronic form, a certain number of (basic) requirements, taken from the printed document world to which people are used, should be fulfilled. From the information provider's point of view, the commercialization of information in electronic form should protect the intellectual rights of the author and/or publisher and guarantee his ability to make profit. On the other hand, the information consumer requires guarantees for

the authenticity of the information and preservation of his anonymity in accessing it. These requirements combined with marketing needs, like payment policies and trust chains as well as alliances with banking and electronic payment institutions, will greatly influence the design and implementation of any system targeting the commercialization of information in electronic form.

In the last few years a number of systems appeared providing different means for the commercialization of electronic content. The most notable of these systems include IBM's cryptographic envelopes, *Cryptolope* (Kaplan, 1996; Kohl, 1997), and InterTrust's digital box, *DigiBox* (Sibert, 1995), as well as *SoftLock* of SoftLock Inc. and *Folio4* products of OpenMarket Inc. (Open Market, 1999). The main characteristic of this technology is to bind the usage policy to the content in a secure way. This approach of "boxing up bytes" is commonly known under many terms such as *cryptographic content wrappers*, *boxology* or *secure content encapsulation*. These systems have a strong emphasis on content commercialization, copyright protection and usage metering. However, they have major limitations in that they bind their users to proprietary systems or commercial partners and networks.

In the frame of the Swiss SPP project MEDIA (Konstantas, 1996) we have designed and implemented *Hep* (Morrin, 1995), a *Hypermedia Electronic Publishing* agent-based framework for the commercialization of arbitrary information in electronic form and a pilot application, *HyperNews*, allowing the commercialization of electronic newspapers (Morin, 1997; Morin, 1998). In this paper we present the implementation of Hep and describe the pilot application *HyperNews*. In the next section we present the concept of Superdistribution and give an overview of the most representative systems in the area of electronic content commercialization. Then we describe the requirements that a system must support for the commercialization of electronic information. Later we present Hep, a framework for the commercialization of electronic documents and some quantitative results of the system. Finally our conclusions and directions are presented.

Superdistribution

The term *superdistribution* was coined by Ryoichi Mori in 1987 and described in a paper published in 1990 (Mori, 1990). The initial idea was conceived independently by Mori in 1983, as the *Software Service System* (*SSS*) (Mori, 1987), and Brad Cox in 1984, as the *CopyFree Software* (Cox, 1996). Both systems aimed in solving the crucial problem of software distribution enforcing fair compensation to software producers and protection of the software against modification with the least possible burden from the user's point of view.

Mori observed that while trying to detect whether software was copied (i.e., software piracy) was particularly difficult, it was easier or almost trivial for a program to detect and monitor its use. From there on, he proposes a model where programs are encrypted prior to their release, thus enabling and allowing wide and uncontrolled copying and distribution without any problem of piracy since payment becomes bound to usage rather than to acquisition of the software. Mori describes a set of four desirable properties that must be satisfied for software superdistribution:

- Software products are freely distributed without restriction. The user of a software product pays for using that product, not for possessing it.
- Software products can be executed by any user having the proper equipment, provided that the user adheres to the conditions of use set by the vendor and pays the fees charged by the vendor.
- The proper operation of the superdistribution system, including the enforcement of the conditions set by the vendor, is ensured by tamper-resistant electronic devices such as digitally protected modules.
- The vendor of a software product can set the terms and conditions of its use and the schedule of fees, if any, for its use.

The resulting proposed superdistribution architecture relies on three principal functions: first, administrative arrangements for collecting accounting information on software usage, and fees for software usage; second, an accounting process that records and accumulates usage charges, payments and the allocation of usage charges among different software vendors; and last, a defense mechanism utilizing digitally protected modules that protects the system against interference with its proper operation.

In Mori's design, computers are equipped with a device called *Superdistribution Box* (*S-box*). Computers equipped with such devices become S-computers. These boxes are to be understood as tamper-resistant devices embodying microprocessors, RAM, ROM and a real-time clock intended for storage, processing and management of sensitive elements such as deciphering keys and other aspects of the superdistribution system. The resulting encrypted software together with its usage terms and conditions is called an S-program. Its permanent encrypted state has the nice property of enabling it to be transmitted over untrusted and insecure communication channels. Furthermore, since programs are encrypted, they can be copied and distributed by anybody without causing any prejudice.

In doing so, Mori turns a major drawback into a major asset. Namely, the inherent nature of software that allows it to be copied and distributed in a marginal, cost-effective way, turns out to be a real asset. In this scope, users become themselves "legal" re-distributors of software they like and use most. Based on this work, two prototype S-box systems were built: the first one based on a NEC9801 personal computer in 1987; the second built as a co-processor for a Macintosh in 1990.

Secure Content Encapsulation

In the last few years, coupled with the general advent of the Internet and emerging electronic commerce, commercial systems based on work done by Mori and Cox on superdistribution have appeared. These include IBM's cryptographic envelopes, Cryptolope (Kaplan, 1996; Kohl, 1997), InterTrust's digital box, DigiBox (Sibert, 1995), SoftLock of SoftLock Inc. (Softlock, 1999), Breaker Technologies' SoftSEAL (Mauth, 1998), and Folio4 products of OpenMarket Inc. (OpenMarket 1999). These systems have a strong emphasis on content commercialization, copyright protection and usage metering. However, their major limitation is that they bind their users to proprietary systems or commercial partners and networks. The main characteristic of such technology is to bind the usage policy to the content in a secure way. This approach of "boxing up bytes" is commonly known under many terms such as cryptographic content wrappers, boxology, secure content encapsulation etc.

Cryptolope is a Java-based software relying on three components. First, the *Cryptolope Builder* can be thought of as a packaging tool allowing building the cryptographic envelope holding both the content and the business rules for its use. This tool is basically used by content providers. The second component is to be used by information consumers: the *Cryptolope Player* is the interpreter for accessing the Cryptolope content. It uses a trusted HTML viewer and interacts with the *Cryptolope Clearing Center*, which is the third component of the architecture. It is basically a trusted third party providing key management, payment system and event logging/usage metering. The major problem faced with their approach was that it was a closed proprietary system. Users were forced to use IBM's InfoMarket infrastructure for the clearing center acting as a trusted third party thus binding them to IBM. This is probably the reason why Cryptolope has not encountered the anticipated success. In fact, a key factor of success for this type of technology relies in how open it is to integrate other commercial partners be they clearing centers for copyright and/or usage, financial institutions or content providers.

The ***DigiBox*** technology (by analogy to the idea of a digital box) is probably the leader in the field currently. This technology, developed by STAR Lab (Strategic Technologies and Architectural Research Laboratory) is also a secure content wrapper and is the foundation of the commercial products Commerce 1.0 and Enterprise 1.0 of InterTrust Technologies Corp.

The DigiBox architecture is a secure content wrapper. In their approach, content is called *properties* and the policies defining their usage are called *controls*. A DigiBox can hold one or more properties as arbitrary data. The controls can be delivered in the same DigiBox or independently in a separate DigiBox. Controls are linked to properties by cryptographic means.

In a DigiBox, high-level elements such as headers and general information are encrypted with a transport key. Properties are encrypted with other keys which can be delivered separately if needed. The transport key is composed of two parts. One key is included in the digibox and is combined (XOR) with another one stored locally in protected storage where the DigiBox is opened. The part included in the DigiBox is encrypted with a public key algorithm.

SoftLock of SoftLock Services Inc. is a password-based locking mechanism for software and documents. SoftLock's technology ensures that the password, which unlocks a particular product in one context, differs from the password, which will unlock the same product in another context. This is done by a proprietary scheme that generates a SoftLock ID (SLID) based upon the context in which the document is used. The SLID can be linked to anything: the user's name, a specific computer, or even, when technology becomes available the user's voice print. How these parameters are passed to SoftLock depends upon the authoring or programming environment.

SoftSEAL of Breaker Technologies Ltd. is a plug-in based system for on-line license acquisition. With the SoftSEAL system, the vendor seals his product into a secure wrapper and associates it with a product code, which eventually defines the licensing type. The same product must be sealed and associated with different product codes in order to provide it with a different licensing schema. Feature codes associate with a set of capabilities to a product code providing different access levels to the underlying product. When the customer downloads the Web page containing the sealed component, his browser should be able to recognize, handle the cryptographic wrapper and "display" the content. This is done by using browser plug-ins.

Folio4 products of Open Market Inc. provide a whole set of tools for Internet-based payment, content management and publishing. The SecurePublish product provides an enterprise solution for rights control and usage metering within an organization's intranet. The operation of SecurePublish is based on the Rights Administration, a system for securing and managing protected content for a local environment via licenses. Each license controls access and limit rights for one or more rights managed infobases. Access rights include the tasks a user can perform or what a user may see once access has been given. The collected access information is sent to the publisher at regularly scheduled intervals and is used for license renewal negotiation.

Requirements for the Commercialization of Electronic Information

Based on the superdistribution concept and targeting in the design and implementation of a system that will allow the commercialization of information in electronic form, we first defined the requirements that need to be satisfied. These

requirements reflect the fundamental interests of the electronic information publisher and consumer, namely the fact that the publisher is interested in providing a profitable service fulfilling the needs of the consumer, while the consumer is interested in obtaining a reliable service for the right price. Here we give an overview of the basic requirements; an extended description can be found in Morin (1997).

- **Anonymity.** As with traditional commerce an information consumer does not need to reveal his identity to the publisher of a magazine or newspaper in order to buy it, so with electronic information the consumer should be able to buy information without having to reveal his identity in any direct or indirect way.
- **Information Granularity**. In the traditional publishing industry, the smallest information unit that can be put on the market is the issue, which bundles a substantial number of information pieces and its price is fixed accordingly. In an electronic environment however, the granularity of the marketable information unit can be brought down to the level, for example, of a single article. In fact what the electronic information consumer will be interested in is buying independent pieces of information and not complete editions.
- **Superdistribution.** It is quite common for a person to read something and to wish to show it to somebody else. With printed material this is easily done by simply cutting the item or giving the complete edition to another person. In an electronic system, however, this process although feasible has important copyright violation side effects. The reason is that instead of passing to the other person the specific item, and in consequence losing possession of it, we actually make a copy of the item. This copying can violate copyright law. Thus the electronic system should allow the reader to freely pass information items to other persons without violating copyright law.
- **Subsequent Access.** Once the consumer has paid for an information item, he should be able to read it again at a later time without having to pay for it a second time. Although this might look like an obvious possibility it is an important requirement since payment of author rights is done at the moment of reading the document. Subsequent reading of the information by the same reader should not result in a second payment of the author rights (unless explicitly expressed by the policy attached to the document).
- **Free Choice of Providers.** In an electronic commerce environment, the consumer should be able to choose freely from where he buys services and goods. This means that any system installed should not be bound to a specific provider, allowing the consumer to freely choose the provider from whom he will buy information.
- **Off-Line Activity.** It is a common practice for a person to buy a magazine or newspaper and read it at different locations, like when traveling or even at the beach. An electronic information system should allow the reader to read, and consequently pay for, information items that are stored locally, even in the absence of a network.
- **Notification of Update Availability**. Being informed "on-time" is a major issue for the information consumer. In an electronic information dissemination system means are needed to offer the information consumer the possibility of being notified immediately upon availability of information updates on desired issues.
- **Information Selection.** Different information providers have different specializations and present information in different ways. In an electronic information world we will have a large number of electronic information providers available. The information consumer should thus be able to define which kind of information he wishes to obtain from each electronic information provider.
- **User Interface.** Since it is neither feasible nor desirable to create a new hypertext browser, the electronic information system should be able to run within any widely available browser, like Java-enabled Web browsers.
- **Marketing policies.** Of major importance in the success of a service is the choice of the right marketing policy. The information providers should be able to implement flexible and adaptable payment policies. For example the price of an article can change depending on its publication date (last week's news has in general no value). This should be feasible without any need for the reader to interact with the publisher: the article itself should be able to figure out its current price.
- **Information Access and Information Evolution.** The information consumer should be able to easily access older information and trace the evolution of events. This means that published information should be immutable and identifiable.
- **The Information Consumer as an Information Provider**. To illustrate this requirement we consider the following two examples where a user receives an information item: (i) the user decides to forward it to a friend together with some comments; (ii) the user decides to forward it to a client together with some comments for which a fee must be paid. In both cases a new information item is created, which contains the original information and the comments with a possible corresponding price. The idea here is that an information consumer can become an information provider of his own added value and a reseller of other information provider's material without infringing any copyright or intellectual property law. Thus information consumers should be able to publish new information items, embedding information of other providers together with their own added value information. The final reader will have to pay both providers in order to read the information and the attached comments.

The Hep Framework

The Hep framework is part of the MEDIA project, which aims at developing an environment for the commercialization of electronic documents. The MEDIA environment aims at offering the information providers of newspapers, magazines, books and alike the means to commercialize electronically the information they hold under similar conditions to those of printed versions. The target of the Hep framework is to provide a platform for the development of applications for the commercialization of electronic content/documents, ranging from electronic newspapers to electronic books and from music to video, enforcing copyright control and revenue collection at the point where the information is actually accessed. Note that in the rest of the paper we will use the general term "electronic document" to refer to any type of electronically encoded information, like text, music, video, images, etc.

Traditionally electronic documents are seen as a collection of data that include the document contents (text, images etc.) and possible simple or complex formatting instructions. The reader of the electronic document has all the software needed to display the data. Hep however takes a different direction and views electronic documents as programs that need to be interpreted in order to reveal their contents. The reader does not have a simple data displaying software but an interpreter for the language in which the electronic document is encoded. This way the electronic document is a program that the reader must execute in order to read it. The document producer can thus include instructions not only defining the structure of the information and how it should be displayed, but also for interacting with the reader, e.g., asking authorization passwords, verifying the integrity of its contents, decrypting sensitive parts of the document, allowing the reader to interrogate it and obtain basic information about its contents, and even sending messages through the network to the document publisher. These types of programs are referred in the literature as *agents*.

The most important issue in the distribution of electronic documents is how to enforce copyright and ensure the payment of ownership rights. Traditional approaches enforce the copyright control and payment at the point of the distribution of the electronic document delivering the raw document data. The customer receives either a password, which enables him to login to the provider's server and retrieve the desired documents, or/and a decrypting key allowing him to decrypt the document. More recent approaches (Kaplan, 1996; Kohl, 1997; Sibert, 1995) encapsulate the document data in a secure wrapper and can guarantee that the content will be delivered only after payment has been received. However they do not consider what happens after the delivery of the content. The user has the cleartext of the document and can produce and distribute infinite copies at virtually no cost.

Our approach with Hep differs from these approaches at the point where the copyright control and payment of ownership rights are enforced. The Hep approach transfers the point of payment from the provider to the reader site, delivering an agent instead of raw data, so that the copyright control and payment is enforced when reading the document and not when downloading it or unwrapping it. An agent contains an encrypted version of the document as well as instructions on how and when to decrypt it. When the agent is interpreted by the Hep system-interpreter, it initiates the payment of the corresponding fees and when the payment transaction has been completed, it decrypts the document and *presents* it to the reader. Note that the Hep system does not deliver the content to the user nor does it store the decrypted document locally in the reader's system but rather keeps it in memory and constantly under the control of the agent. This way the unauthorized extraction of the plaintext document by the reader is a non-trivial task. On the other hand the agent-encapsulated article can be freely copied and distributed without compromising the author's rights.

Enabling Technologies

The Hep design and implementation started in early 1996. The technology choices upon which Hep is based are strongly interdependent, rely heavily on the Java language, and are based on the state-of-the-art technology available at the beginning of the project. In this section we briefly present the technology choices we made and describe how the current sate of the art will affect them in the forthcoming migration to today's (early 1999) technology.

- **The programming language**. Very early in the project definition phase, the Java language was beginning to establish itself as a choice language for rapid prototyping and platform independence. The whole implementation was carried out using the Java Development Kit Version 1.0.2 (JDK 1.0.2) together with early releases of the RMI (Remote Method Invocation) and Object Serialization packages. With the currently available versions of the JDK (i.e., JDK 1.2), most of the technology described below comes for "free", in the sense they are either part of the Java distribution or have been taken into account for easy integration within the Java environment, e.g. Remote Method Invocation and Object Serialization, Security, Java Beans, Java Foundation Classes (JFC), Swing Graphical User Interface Components, etc. Currently we are working on the migration of *HyperNews* to JDK1.2 in order to take full advantage of these integrated features and drop on the way most of the external packages we discuss below.
- **The Agent Execution Platform.** At the time the project started very few agent environments were available. *Mole* (Strasser, 1996), developed at the University of Stuttgart IPVR, was one of the few existing environments that was fully implemented in Java. This was a strong advantage with respect to the time-critical aspect of the project, portability and limited manpower. During the imple-

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.