The Long March to Interoperable Digital Rights Management

Rob H. Koenen, *Senior Member, IEEE*, Jack Lacy, Michael MacKay, and Steve Mitchell, *Member, IEEE*

***Abstract*— This paper discusses interoperability of digital rights management (DRM) systems. We start by describing a basic reference model for DRM. The cause of interoperability is served by understanding and circumscribing what DRM is "in the whole." Then we outline and contrast three different approaches to achieving interoperability. One approach relies on flexible network services to provide functionality where it is needed, perhaps by bridging different systems. We describe an experimental service orchestration system (NEMO) that enables such an approach.**

***Index Terms*— Digital Rights Management, Trusted Computing, Digital Media Distribution, Standards, Web Services**

I. INTRODUCTION

DIGITAL Rights Management (DRM) is a collection of technologies that enable technically enforced licensing of digital information. DRM makes it possible for commercial publishers to distribute valuable content electronically, without destroying the copyright holder's revenue stream. DRM can also be used in other settings to enable safe distribution of digital content including, for example, document management within and between corporations, protected email, medical patient records handling, and government service access.

At a minimum, a well-designed DRM system provides:

Governance: DRM is different from classical security and protection technologies [1]. Conventional media distribution systems based on conditional access techniques protect media during transmission using a control model based on direct cryptographic key exchange. DRM systems, on the other hand, implement control, or governance, via the use of programming language methods executed in a secure environment.

Secure Association of Usage Rules with Information: DRM systems securely associate rules with content. These rules determine usage of the content throughout its lifecycle. Rules can be attached to content, embedded within content (*e.g.*, via watermarking), or rules can be delivered independently of content.

Persistent Protection: DRM systems are designed to protect and govern information on a persistent basis throughout the content's commercial lifecycle. Protection is frequently provided using cryptographic techniques. Encrypted content is protected even as it travels outside of protected distribution channels.

The use of DRM in commercial end-consumer media distribution is controversial for several reasons. DRM allows content providers to create licenses that are different from, and more rigidly enforceable than, the de facto generally understood licenses that have accompanied traditional media (CD's, VHS tapes, and DVD's). Conversely, the nature of today's DRM technology makes it difficult to automate accurately some existing usage conventions, such as the United States' fair use traditions or European privacy expectations.

DRM license enforcement requires security safeguards on home equipment to protect the interests of content vendors. Although it is common for basic utility vendors to install security systems around home metering systems (*e.g.*, cable television, water, electricity and natural gas), some consumers are wary of DRM systems operating on their family PC, which is used for many personal tasks besides presenting media.

Traditional media distribution (before the mid-1990s) has been tied to physical media, such as music CD's and video tapes. Making and distributing high-quality copies of music and video was difficult for the average consumer. Successful business models have been well established around the processes of manufacturing, distributing, merchandising, and charging consumers for individual copies of a work. Early electronic distribution systems have likewise been built around the notion of digital copies of works ("copy control systems"), but this paradigm is becoming less relevant as it becomes easier for consumers to manage content as disk files on their home network, in their cars, at work, and in school.

It is easy today to find consumers who would think it appropriate to pay full price for a second factory pressed copy of a favorite music CD, but who have few misgivings about downloading free (unauthorized) digital compressed copies of music for which they (or someone in their family) already own a commercial CD. Consequently, consumers are developing their own ideas of what the right business models should be for commercial music licensing. Commercial publishers are scrambling to work through the business and technical hurdles to deploying business models that protect their interests and are acceptable to consumers, device manufacturers, and service providers.

Manuscript received September 12, 2003.

M. MacKay is with InterTrust Technologies Corp., Santa Clara, CA 95054, USA (Corresponding Author; phone: +1 408 855 0100; fax: +1 408 855 0130; email: mmackay at intertrust dot com)

R. Koenen, J. Lacy, and S. Mitchell are also with InterTrust Technologies Corp. (phone and fax as above; email: {rob, lacy, mitchell} at intertrust dot com)

The result is the emergence of DRM-enabled digital music services, such as Roxio's Napster service (originally known as pressplay), Apple's iTunes Music Store, Musicmatch Downloads, and others. Apple's music service has so far been the most popular with consumers, but we have not yet heard the last word in legal on-line music distribution [2],[3],[4]. BuyMusic, Musicmatch, MusicNow, Napster, and numerous others use Microsoft's Windows Media Audio format, which bundles DRM capability with an audio codec and a file format. Apple's iTunes uses an open standard audio codec (MPEG Advanced Audio Coding, or AAC) and a proprietary DRM system. The Microsoft and Apple formats are not compatible. Microsoft's format is supported on the largest variety of portable music players, while Apple's format is currently supported on only one – its own iPod. (Reportedly this is the current top-selling music player [3]). At the time of writing, no portable music player supports both formats.

This paper focuses on the issue of DRM interoperability. There are several reasons why DRM interoperability is desirable. The content industry desperately needs to deploy legitimate content services that compete favorably (based on features, not on price) with unauthorized free services. A simple and seamless user experience must be part of that goal, and DRM interoperability is necessary to achieve it.

Content providers and e-commerce service providers would like to see a healthy business climate from which they can multi-source essential technologies like DRM, especially when these technologies must adapt rapidly to evolving industry needs and consumer expectations. The DRM market is strongly influenced by network effects: a DRM technology becomes more valuable as it becomes more widely adopted. Thus there are strong forces pushing DRM technology providers toward interoperability, even as vendors attempt to differentiate their products based upon features.

While many people have articulated a goal for media distribution where any content is available to anyone, anytime, anywhere on any useful device using viable business models, significant barriers exist to the goal of an interoperable and secure world of media related services:

- Overlapping de facto and formal standards
- Implementation technologies are not interoperable
- Consumer devices cannot locate and connect to needed services
- Web services standards do not bridge services spanning web distribution and personal area network protocols
- Impedance mismatches between different trust and protection models
- No unified notion of content governance in current peer-to-peer distribution models

We outline some of the possible approaches to achieving interoperability, and discuss related issues. We start in the next section by describing a basic reference model for DRM. The cause of interoperability is served by understanding and circumscribing what DRM is "in the whole." We then outline and contrast three different approaches to achieving interoperability. One approach relies on flexible network services to provide functionality where it is needed. Finally, we describe an experimental service orchestration system (NEMO) that enables such an approach.

II. Towards a DRM Basic Reference Model

Commercial practice across a variety of DRM systems has matured to a point where robust technical patterns can be identified as a basis for establishing a DRM Basic Reference Model[1]. In this section, we consider the architecture of current DRM systems in order to identify common technical elements and the requirements they try to address. Proceeding from this analysis, we then outline a reference model that may serve as a basis for coordinating evolution and interoperability of next-generation DRM systems. Establishing a general vocabulary and a set of reference concepts is the first step in building a framework for interoperability of heterogeneous systems.

A. Current DRM Architectures and Industry Practice

Fig. 1 illustrates an abstract system architecture based on DRM application and service elements representative of a variety of contemporary commercial DRM systems. Key concepts in this diagram are as follows:

- Content and associated usage rights enter the system through a packaging process, typically under the authority of the Content Licensor.
- Packaging services produce protected content and either full licenses, or rules and metadata as input to a Licensing and Reference Service. Licenses can usually be personalized based on the particular parameters of the license-requesting party [5].
- Consumers use a local Consuming Application to transact with the licensing and reference services for licenses, and interact with streaming or download services for acquisition of the protected content. Often, the licensing service provides the reference to the correct content and associated distribution source.
- The consumer may be licensed to transfer protected content to another peer system (*e.g.* other "full-featured hosts"), or to a portable device with DRM capabilities. Portable or "tethered" devices interact with the DRM system by proxy via a more capable upstream system (*e.g.* the "full-featured host"). The host may for example create a restricted form of the original license better suited to the capabilities the device, or may buffer or cache certain usage information on behalf of the less capable device.

Each of the elements in fig. 1 may consist of multiple systems in a real-world implementation. For example, licensing services may embody an entire distribution value chain consisting of retail, subscription or download services.

Each element may be hosted by different business entities,

[1] The CEN/ISSS Digital Rights Management Final Report [16] provides an overview of evolving DRM technical architectures with the goal of "identifying the current status of DRM usage and possible means to ensure effective implementation of DRM in the marketplace."

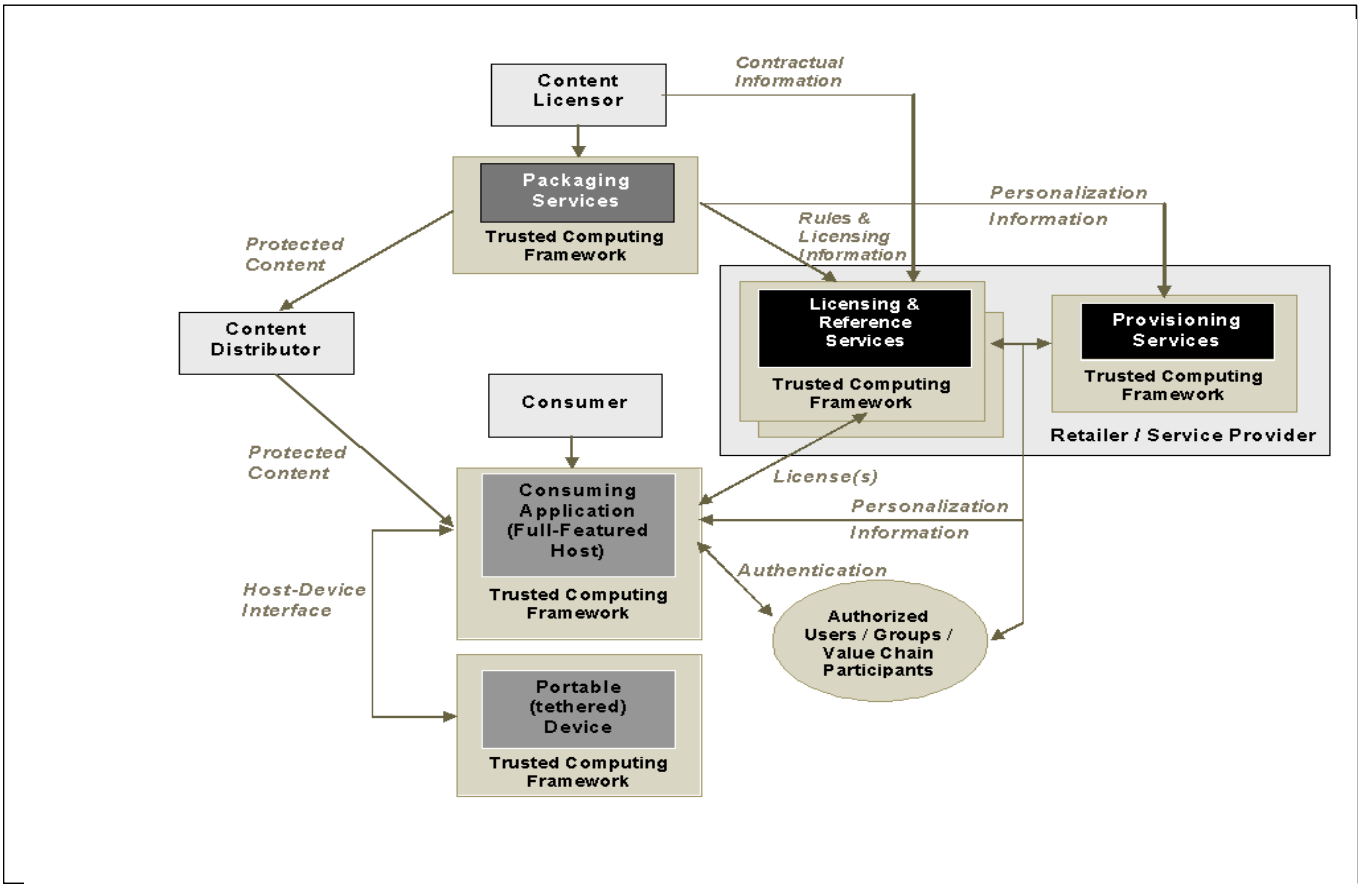

Fig. 1. Abstract DRM System Elements

acting in cooperation with other parties' systems based on contractual business relationships. Current deployment scenarios for DRM systems involve mutually well-known business partners, carefully architected technical responsibilities and negotiated business relationships. However, increased business automation and more dynamic business relationships create the need for flexible provisioning and management of DRM infrastructure.

DRM applications and services (consumption, packaging, license services, provisioning services, etc.) are all built on elements of the trusted computing framework, which includes secure software distribution and execution environments, trusted identity management, secure policy and rule processing and enforcement, supporting cryptographic functions and key management, and tamper resistance. Provisioning services support adding new participants and services, and supplying DRM systems with supporting software, certificates, *etc.*

The ability to programmatically configure and manage trusted and secure relationships between the participants and the underlying DRM technology is paramount [6]. All of the parties in the value chain must trust that distributed content or information and its source are authentic, is accessible only by intended or contracted receivers and is used by those receivers consistently with the contracted rights. Devices and services must be qualified as trustworthy and then maintained as such.

B. *Value Chains and DRM Systems*

Understanding roles in the commerce value chain and how these interact with DRM services is essential.

A detailed model of roles involved in electronic copyright management systems was developed by the European Commission-funded Imprimatur project. Completed in 1998, the goal of Imprimatur was to "understand and analyze the context in which Electronic Copyright Management Systems are to be developed," and which "reflect[s] current business practices for trading and licensing multimedia documents [by identifying] relevant roles, their relationships and corresponding transactions" [5]. Roles and responsibilities addressed by the Imprimatur model include:

- The Creator – the party responsible for delivering their creation to the Creation Provider.
- Creators may assign exploitation rights to a Rights Holder (*e.g.* a collection or licensing agency).
- The relationship between Creators and Rights Holders and associated contracts are maintained in an IPR Database.
- The Media Distributor is expected to pass appropriate royalties to the Rights Holder according to the current payment details stored in the IPR Database.
- The Purchaser (consumer) may use the creation, and if they generate a new composite document based on it then they also become a Creator. In order for the Purchaser to

perform functions associated with the Creator role, they must have obtained the required permission from the corresponding Rights Holder of the original creation. Rights Holders of original creations automatically have rights on composite creations – the flow of royalties is determined according to the IPR Database.

Few DRM systems take all of these types of roles, relationships, and activities directly into account as part of their intrinsic design, leaving contract management, auditing and accounting issues to a diverse array of largely unintegrated back office systems. With increased end-to-end systems automation and sophisticated digital content manipulation and aggregation services, models like Imprimatur will likely receive increased atttention in new architectures. Possibly the most thorough attempt to date in a single DRM system was undertaken by InterTrust in its Commerce system [7].

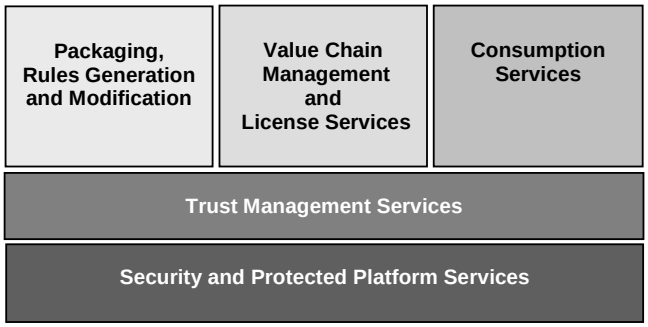

Fig. 1 – DRM Basic Reference Model

C. DRM Systems Functionality

The proposed basic DRM reference model is illustrated in fig. 2. We now frame the functional characteristics of the five main domains of our proposed basic reference model.

1) Packaging, Rules Generation and Modification

The point of entry to the DRM-managed content and governance lifecycle includes technologies supporting content packaging, specification of rights and associated data, and generation and modification of digital items.

a) Content Packaging

Content packaging is the process of preparing content for DRM protection – placing content into a secure container, usually by encrypting it, associating the necessary identifiers and metadata, and logging and cataloging the content, its identifiers and metadata, and its cryptographic material. Consumers and associated consumption processes may also be enabled to package their own content[2]

Content packaging can be closely associated with rules and license generation, or may be completely independent from it. Content identifiers couple the protected content with rules and content protection keys. Therefore, rules, packaged content, and content keys may be generated together or separately, at the same time or at different times. They may be delivered together, through the same channels, or separately, at different times, through different channels. In a production environment content may be packaged initially without rules. Alternatively, content may be packaged on-demand and immediately associated with rules.

The content may contain directions as to where licenses or offers associated with the content can be acquired, or other offer metadata that can be used to automate downstream distribution processes.

Content protection is typically accomplished using cryptographic processing, where content protection keys are made available to one value chain participant or consumer, and are not exposed in the clear to other value chain participants or consumers. Key management procedures can bind or associate a content package to any security principal, including individual consumers, devices, certain types of secure media, or content-sharing networks (*e.g.*, a network of home media devices). Associating content with a consumer allows the protected content and license to be transported to other systems on which the consumer is also authorized.

b) Rules Generation and Modification

Any authorized member of the value chain from packager to consumer may create rules to be associated with a content package. Rules may be used to govern consumer access to content as well as to govern the actions of other value chain members on the content or information associated with the content. For example, usage rules may require authentication on access or usage, or require license updates to be obtained before operating on the content[3].

Rules may specify consequences such as generation of audit records based on content usage actions or attempts at usage, such that the audit records are securely delivered to a designated authority prior to execution of the action governed by the rule.

Rules are often associated with the whole piece of content, but may also be managed at the granularity of a content sub-element (*e.g.* stream, component, etc.). Rules can also be associated with a class of content (*e.g.*, all content belonging to a particular owner, all audio content, all low-bitrate content, etc.) rather than a specific content instance.

Rules can be delivered as separate files (*e.g.*, a license), or combined with the protected content (integrated with the content data format itself), or both. Alternatively, the rules can be provided as input to value chain management and licensing services, or applied in conjunction with processes for resolving references to the content.

[2] The term "consumer" typically refers to retail end-users but may also apply to other value chain participants – regardless, consumers are participants of the managed value chain and may participate in a broader class of functions than strictly consumption and rendering.

[3] For example, expired rights might require license updates to enable access or usage.

Rules, terms and conditions, and consequences can be represented in a variety of different ways. For example, one approach is to use a standardized rights expression language such as the MPEG-21 Rights Expression Language (REL, [8]) or the Open Digital Rights Language (ODRL, [9]). Alternatively, rules may also be encoded in formatted text (such as XML or named key-value pairs), or possibly via compiled or interpretive code as part of an application.

In some systems, it is possible to modify or extend rules after their initial creation. For example, value chain management and licensing services may support the ability to select and apply rules that have been updated to reflect up-to-the-minute changes in business offers, regardless of when the content was packaged and placed into the system.

In the final stage of rules generation, rules are embedded into data structures that can be linked to the content. There are a variety of mechanisms available for packaging rules. For example, sets of rules may be organized into "offers" that describe the content and the associated license for presentation to a consumer or other value chain member. Offers may be delivered to a content distributor, who may choose to present some or all of the offers to other participants further down the value chain. Associated collateral information and promotional content can be included in a separate package for use in retail promotion and downstream distribution.

2) Value Chain Management and License Services

A common characteristic of systems that support non-trivial operational models (such as subscriptions, superdistribution, push-distribution, etc.) is the ability to produce, modify, assemble and aggregate rules, and negotiate conflicts involving rules from one or more sources..

Consumer licenses are sometimes the result of a collaboration of multiple value chain participants. Authorized value chain members may insert new rules into the licensing structures, using processes that are themselves governed. The rights of various services to interact with the content's distribution process may be encoded in rules delivered directly to the service or that are referenced using the same identifiers or references that are associated with the content.

Value chain management services may include post-transaction processing (*e.g.*, allocation of the value exchanged such as financial payment, usage data, etc.) per contractual obligations [5]. Such post-transaction processing rules can be included in the license associated with the content (whether packaged together with the license or separately), or created as an electronic contract covering specific offers or content and delivered separately.

Historically, the terms by which value chain participants are allowed to interact with the content and rights to its use are expressed via contractual relationships between creators (or creation providers) and other value chain participants. We anticipate that contractual relationships may be automated using similar mechanisms (*e.g.* electronic contracts) as those used to control access to content by consumer applications. Contracts may be encoded using a Contract Expression Language [10], similar to rights expression languages used for encoding content usage rights. Electronic contracts are then delivered to participating entities and used by trusted applications to manage content distribution rights. The ways in which these terms are delivered and managed are discussed in greater detail in the next section.

Frequently, rights and contractual obligations associated with a piece of content already exist as a result of prior interactions with the content (*e.g.* as part of prior distribution arrangements). Rights discovery refers to a set of functions provided either by technically automated or other means, such as conventional business processes, for referencing these existing rights and obligations.

a) Value Chain Management

Value chain management refers to those system facilities that track, serve and govern value chain participants. Value chain participants have interests in the distribution of products and provide decision-making, reporting, and other processing services affecting the digital content under their control. Just as rules govern the use of protected content, rules and policy govern the ways in which value chain participants interact with one another and with their associated content.

Static value chain management refers to approaches where offer and consumption rules are computed at content packaging time. An expression of rules can be distributed with content packages for examination or modification by other participants in the value chain.

In the static model, content packages are created for a particular set of distribution participants. The value chain management process is parameterized at packaging time with information about the known and identified participants, and the packager output conveys the necessary information in advance of actual participation. Once packaged, modification to the value chain information is governed by the associated rule set. The upshot of this early-binding approach is that unanticipated business model changes might necessitate content and/or rules repackaging from an original source.

The **dynamic value chain management** model is late-binding. In the dynamic model, rules governing the use of value chain information are accessed on demand through network services, rather than being carried as they were encoded at packaging in an early-bound and immutable configuration. Rather than copying packaged files to each value chain participant, content may be distributed by reference [10]. The rights to the content are distributed based on these references and the references may be incorporated in or used by other structures, such as licenses. Reference Services fulfill requests for content consumption by consulting their current rule sets [10].

Dynamic value chain management allows for modification of the value chain information as references to the content move through the distribution channel. The dynamic model allows content to be packaged without advance knowledge of distribution configurations. Distribution configurations can change in response to new contracts, law, or business models.

In addition to enabling greater adaptability and responsiveness to changes in the business environment, dynamic value chain management may provide better ways to accommodate complex rights management issues, such as fair use rights.

b) Licensing Processes

License services manage and distribute content licenses. DRM functions associated with license services commonly include:

- Management of data structures carrying rules (*e.g.* licenses or offers) and cryptographic information (*e.g.* content protection keys).
- Discovery, delivery, authentication, and management of offers
- License request processing, license generation, license association (binding) and delivery of resulting licenses to requesting entities (devices, services, applications or security principals associated with authenticated user identities) consistent with the requirements of the rights holders and governing contracts.
- Validation of trusted status of entities requesting services of the system (*e.g.* authentication of value chain participants and the business relationships between them).
- Validation of transactions from peer value chain systems authorizing generation and association of licenses on behalf of a third party.
- Processing and validation of any rules required for delivery of the license, such as enforcement of geographic restrictions; enforcement of time restricted offers; and validation of credentials from the requesting party.
- Management and enforcement of subscription data.
- Event reporting for payment functions (or any other exchange of value).
- Event reporting for usage tracking and overall system assurance.

3) Consumption Services

Consumption services are functions through which consumers interact with DRM content according to some governed action (*e.g.* playback rendering, editing, printing, annotation, aggregation, etc.). Consumption services are typically associated with consumer client systems, but may also be associated with any value chain participant that accesses or processes protected content, metadata, or rules. Systems incorporating DRM consumption services can take a variety of forms including:

- Application software incorporating DRM functions for protected media services running on a general purpose OS using PC hardware.
- Consumer electronics devices such as set-top boxes, multi-media appliances or game consoles, etc.
- Wireless or personal digital appliances, including those capable of participating in online transactions with value chain management and license services, and supporting operational and trust management services.

Supporting elements of distributed DRM systems, such as value chain management services and license services, must be able to establish and maintain trust with systems that host consumption services. Trusted consumption hosts must protect their operation against circumvention of local DRM processing functions, must enforce rules governing access to packaged data, and must render and otherwise use protected content. Systems that consume protected content typically employ a variety of security mechanisms and may interact with local or distributed security services.

Consuming systems request and acquire protected content through transactions with licensing and potentially other services. These transactions may include information about the requesting system environment and user context – including possibly personalization data, locale, system capabilities, security level or evidence of current certification, and information about the content. Due to the potentially sensitive nature of some of this information, privacy protection is a paramount concern in these functions [11].

Although many systems associate protected content, using cryptographic techniques, to the identity of the requesting system (*e.g.* using a fingerprint based on characteristic attributes of the specific system, or an indelible identifier or key), it is also possible (and increasingly desirable) to license the protected content to an identity associated with an authenticated security principal, (*e.g.* the user or a role associated with the user). Establishing this type of association allows the protected content and license to be transported to other systems on which the user is also authorized.

Once the license is received, the consuming system is able to manipulate the content according to the specified rules.. Rules may express, for example, limitations on the number of plays, time-based usage or expiration, requirements for enrollment in a subscription service, budget transactions with a local stored-value database, authorization from a content management system within a business or between business partners, etc.

The consuming system's DRM components are responsible for enforcing the rules and maintaining any state associated with them. State information must be protected in order to assure integrity against circumvention for purposes such as unauthorized replay or redistribution.

If the rules specify consequences, the consuming system's DRM components are responsible for any required local or distributed transactions such as usage auditing, event reporting[4] or metered payment. Unsuccessful event reporting or auditing may result in prohibitions against further access until such records can be successfully processed.

Rules may also specify whether the consuming system has the right to copy content to another peer or portable device. In this case, the system's DRM components must support device interfaces and non-volatile state (such as copy and check-in / check-out counts) used to maintain compliance with the rules. A device or application to which the content is being

[4]Event reporting includes activities such as successful download notification.

transferred must be able to enforce the applicable content usage rules to a required level of conformance.

a) Consumption and Portable Devices

In many ways, portable devices are just another class of consuming system. Examples of portable devices include personal digital music players and various types of imaging, games, or electronic book devices. The primary characteristic of a portable device is that it is usually managed by a more capable system, what we might call a "full-featured host," that is capable of direct transactions with distributed value chain management and license services. Portable devices typically rely on a secure communications channel managed by the host system for functions such as copying and (re-)associating protected content to the portable device (or a removable secure memory) for offline usage and rendering.

Portable devices typically incorporate many of the following functions:

- The device and/or its portable media may be registered by its unique identity with the host system or consumer electronics appliance, personal area network, license service, or a personalization service.
- They are certified as compliant with applicable security and/or DRM specifications. This is typically represented by a certificate associated with the device.
- A device may register with a host at the time of content transfer, by presenting credentials or a unique identity, allowing content to be secured to the device.
- The device may be re-registered to a new network, service, etc. in which case it is unregistered from the previous attachments. (Content may be keyed to the original network and would cease to be renderable.)
- The device may be disabled, or denied service ("revoked"), based on its status as a trusted element of the comprehensive DRM system. Revocation may involve the invalidation of the certificates mentioned above, or invalidation of the software component used to transfer content in the host device.
- The host system may require information about the portable device including its unique identity, the unique identity of any removable secure memory or media associated with it, and other functional and/or security-related capabilities of the device (including certificates).
- If authorized, the host system may support format translation to a different protected representation that can be consumed by the portable device.
- The host system may support (re-)associating rules with the portable device or its removable storage.
- The host system may support (re-)packaging, re-encryption, and securely associating rules to a representation supported by the portable device.

Host system support for a given type of portable device typically depends on the security level of the device and its DRM capabilities. Usually, rules associated with protected content will determine whether the host can undertake transactions with a portable device, and. hence the portable device may be required to include functionality such as:

- A secure clock or time source
- Dynamic device binding to a personal area network or home gateway
- Secure counters for counted plays and subscriptions
- Device or removable storage hardware unique IDs for secure association of rules, content encryption keys, and associated protected content

4) Trust Management Services

Trust management services are primarily responsible for functions supporting provisioning, certification, secure operation and renewability of elements in the distributed DRM system [12]. Trust management services are relied upon by features in virtually all components of the DRM system and typically entail functions including:

- Support for different trust management topologies (*e.g.* peer-to-peer, web of trust, hierarchical trust models)
- Certification of software and/or hardware
- Registration of software and/or hardware
- Registration of business relationships
- Support for personalization functions provided by the security and protected platform services
- Security lifecycle maintenance including renewability and revocation as required for maintenance of distributed trust and authorized participation across all system elements This applies to software and hardware components of all value chain participants and their relationships
- User certification and credential management services,
- Centralized audit and event logging services.

Trust management subsystems use authorization techniques to regulate activities with risk potential within and between DRM system components. A regulated activity is approved or denied according to a predefined trust policy, and according to credentials and other evidence surrounding the activity. Trust policy can be as simple as a list of trusted activity partners, or it can be a complex decision procedure based upon activity parameters, such as the value being transacted, or the kind of security safeguards in place.

Trust management topologies arise from the ability of relying parties to accept third party recommendations or certifications of the relied-upon party. In social contexts trust is rarely transitive. (A's trust of B and B's trust of C does not necessarily imply A's trust of C.) Nonetheless, the most useful automated trust negotiation systems implement some form of (possibly limited) transitive trust model.

Trust management enables risk management by implementing trust policy. Although trust management systems may make use of authorization technologies to regulate activities with potential risk, trust management should not be confused with enforcement of content usage rules. Content usage rules implement business models, while

trust policy implements risk management. Different people with different expertise and concerns will author trust policy and content usage rules. Trust policy and content usage rules have different life cycles, and are distributed and managed differently. Trust policy and content usage rules are conditioned on different criteria, and have different vocabularies. While content access is an activity with risk potential, trust policy will apply to other activities as well. Despite these differences, content usage rules and trust policy may overlap, in vocabulary and in effect.

5) Security and Protected Platform Services

A trusted environment for persistent governance of rules and content is built on a foundation of security functions. The required security functions may leverage trusted hardware if it is available (*e.g.* smart cards, hardware cryptographic processors, or evolving standards for trusted hardware in general purpose PC and PDA platforms [13]). Security and protected platform services and technologies include:

- *Software tamper resistance*, whereby host and device software and/or firmware is designed to provide protection of content buffers, persistent state, key stores; the program code may itself be obfuscated to reduce potential for reverse engineering; and techniques may be employed that detect and disable attacks against the software itself [14][15].
- *Execution environment security*, whereby host and device software and/or firmware can be validated with various integrity checks to ensure that it is legitimate and has not been modified. Some hosts may provide isolated processing compartments that protect sensitive processes from other processes running on the same platform.
- *Software personalization and individualization services* that support creation or upgrade of DRM components, licenses and information such as certificates required for operation of the host or device system.
- *Authentication* of the user or other elements to the application or local operating system, or with associated distributed value chain management and license services.

Authorization to perform rules processing and actions on governed content typically requires that the system performing the functions prove the integrity of its security mechanisms to other services and applications with which it interacts. (This would be enforced by the trust policies of those services.) Proving integrity usually involves proving compliance with various published security certification criteria developed by the relevant stakeholders. Evidence of compliance with the applicable criteria is strongly associated with the client or device, typically through certificates that are cryptographically bound to the system, and integrity protected and digitally signed by the certifying party. These compliance certificates are then used both by the DRM software as part of its runtime validation as well as in various protocols involved in the acquisition of protected content, rules and licenses.

D. DRM Interoperability and the Reference Model

The DRM reference model described here should be useful for building an interoperability framework capable of accommodating a heterogeneous universe of DRM systems. The breadth of topics involved in standardizing DRM interoperability can be seen to be both systemic and cross-cutting. The functionality is cross-cutting in the sense that it intersects technical concerns spanning application integration, component software and mobile code, operating systems, distributed services, devices, content management and security services. The functionality is systemic from the standpoint that it must support operational and business concerns spanning diverse value chain relationships [5], establishment and management of trust relationships between distributed participants, and governance of valuable digital goods throughout their lifecycle.

If the basic domain modeling underlying the shape of the DRM RM is robust (we believe it is from our work with it over the last year), it will support additional detail design work in each given domain, including the basis for mapping and reusing applicable standards that may already exist in those domains. In this capacity, we believe the DRM RM can provide particular value as a tool for better coordinating interoperability across a variety of different standardization projects (*e.g.* MPEG-21, OMA, OASIS, W3C, IETF, DVB, CR Forum, DOI, etc.). Indeed, we anticipate that other "planes" can (and may need to) be defined and mapped onto the current two-dimensional modeling. One category of work that may benefit from modeling as a separate but coordinated peer level plane is the design of ontologies[5] for DRM actions, consequences and policies, since such topics tend to defy confinement to a single layer and may interact with multiple domains.

Fig. 3 illustrates how functionality described in section 2.3 can be mapped onto the proposed basic reference model.

Summarizing this section, we have outlined the structure of a Basic Reference Model for DRM that embodies dominant architectural patterns derived from observation of practice in a variety of current systems. DRM systems are evolving in a rapidly changing technical environment and thus exhibit a significant range of diversity in the design of interfaces and protocols, formats, trust and security mechanisms, protection mechanisms, governance semantics, and supporting functionality. Reference models assist in the formulation of common design vocabularies and concepts, which can become a further basis for motivating improved interoperability and usability. While the current model as presented is only a starting point, our experience is that it is an effective tool for factoring and organizing technical issues in DRM architectures.

[5] The MPEG-21 Rights Data Dictionary (RDD – ISO21000 part 1) is an example of one such standardized ontology

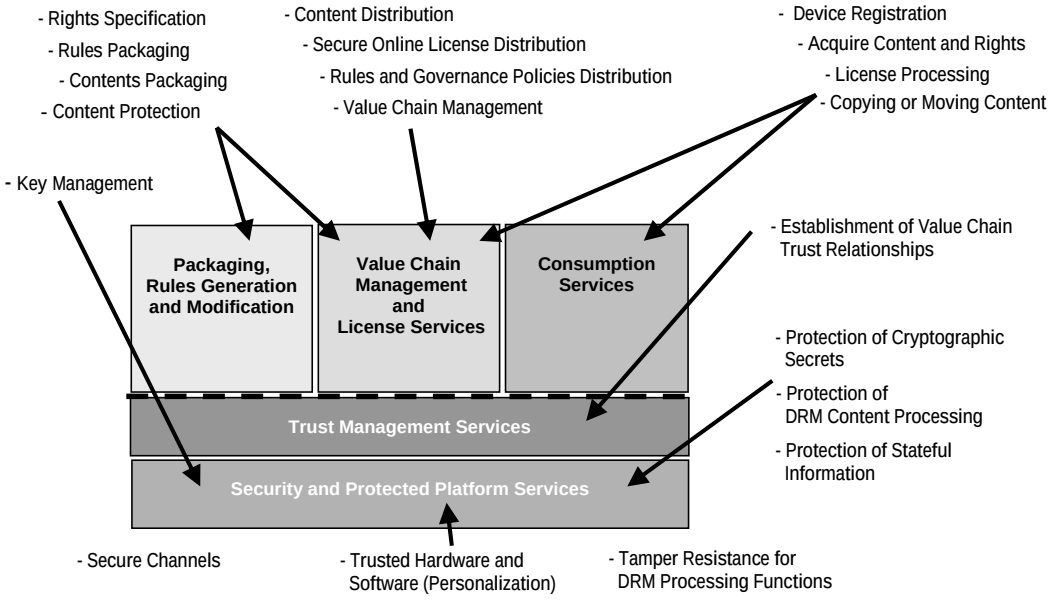

Fig. 2 – Example Functionality Mapping Onto DRM Basic Reference Model

III. Interoperability and Directions in Next Generation DRM Systems

This section focuses on interoperability in DRM systems, and especially the role of interoperability standards. We ask the reader to keep the DRM reference model in mind, as technical analysis of interoperability strategies and the discontinuities underlying systems' non-interoperability also tend to highlight the most important interfaces, services and subsystems, and formats and processing points to be addressed in ongoing development of the model.

A. Approaches to Interoperability in DRM Systems

End-to-end format, services, and device interoperability are desirable goals, both for end-users and others involved in the digital content lifecycle. Normative specifications for end-to-end interoperability are the province of various standards bodies – many such efforts are in progress at this time [1]. Due to the variety of standards used for different modes of distribution, and the diversity of proprietary technologies in the absence of any authoritative standard, full interoperability is unlikely any time soon through standards-setting activities alone. Even where degrees of end-to-end interoperability are possible within a particular industry segment, consumer demand and new business opportunities frequently introduce a requirement for interaction with systems built on different agreements and standards.

Full interoperability can be addressed in several different ways, which we explore in the following sections.

1) Full Format Interoperability

Full format interoperability expects that the interchange representation of the digital content can be consistently processed based on agreement between all participants in the value chain. The audio CD and the DVD are good examples. All participants (creators, distributors, manufacturers, etc.) use the same data representation, encoding, protection scheme, trust management, key management, etc. Usually there is some renewability infrastructure in place to cope with security breaches, but in severe cases, the associated standard may need to be updated. Full format interoperability usually entails robustness criteria and a certification regime to establish trustworthiness and security of conformant implementations.

Pros: Makes it very easy to produce, distribute, and use digital content. Also makes it easy for diverse participants to economically build mass market applications and devices. Promotes efficiency by discouraging redundancy.

Cons: The approach is rigid. Developing industry standards is a long process, to be undertaken when industry requirements are understood. Adoption requires a critical mass in industry which can take years to establish.. The mechanisms specified in a standard may not be the optimal choice in any particular market niche. Also, the approach can be vulnerable: an attack on one system can compromise all simultaneously, perhaps by simply distributing "cracks" on the Internet. Some parts of most DRM security systems (*e.g.*, obfuscation, aspects of tamper resistance) rely upon "security by obscurity" for effectiveness, which is vulnerable to a lack of diversity. Most standardization processes support neither the required certification processes nor the short response cycles in case of breaches. Industry segments will need to provide renewability services and certification infrastructures.

2) Connected Interoperability

Connected interoperability builds on the expectation that consumers will have online access, and relies upon online services, some of them possibly transformative or capable of complex negotiation, to solve interoperability problems in a transparent way. While different parties may do things in different ways, translations or bridges exist between the ways different parties perform DRM functions, and that mutually trusted parties can perform these translations transparently, as

long as devices are connected at least some of the time.

Pros: Makes it possible for different parties to choose their own solutions, and to ensure that security can be renewed, while still providing a certain type of interoperability to end-users. Shifts the burden of responsibility for interoperability from coalitions of manufacturers to on-line service providers.

Cons: Lacks the straightforward simplicity of full format interoperability. Semantic mismatch between different systems must be overcome so that cross-boundary content flows appear natural. Rights may be restricted to a least common denominator when crossing boundaries, for technical and business reasons. Some classes of products will not meet the requirement for online connection. This still requires a direct trust relationship between the coordinating parties to govern the domains between which transforms should take place, or a trusted third party that can negotiate the transform.

3) Configuration-driven Interoperability

Configuration-driven interoperability assumes that system components ("tools", possibly from different vendors) can be downloaded and/or configured in real-time at e.g. the consumer's device or software application. This allows consumer systems to effectively "acquire" functionality on demand in order to accommodate new formats, protocols, and so on. Ideally the consumer need not even be aware that the dynamic configuration is occurring. This is the main concept underlying the MPEG-4 IPMP-X (eXtensions) approach [17]. It emulates the behavior of many software music players that can host downloadable compression codecs.

Pros: Presumes a very late-binding, on-demand model that pushes capability to the consumer's system. Depending on how the interoperability tools are delivered to the consumer, online access may not be required, or at least may be minimized once the configuration is installed.

Cons: Expects that the target platform on which the configurable and downloadable tools execute is capable of hosting and running an arbitrary number of different configurations. This may be troublesome for small consumer electronic devices. Multiple formats can be processed with multiple hosted tools, but the establishment of trust between these components and the host environment still remains – a problem that is not yet addressed by MPEG's IPMP work.

B. Which Approach to Take?

DRM systems interoperability is a challenging problem and there are no easy, universal solutions. While the approaches outlined above have individual merits, they are most likely to find utility in combination. In order to understand how to weigh the trade-offs involved in applying them, we outline a set of supporting principles.

1) Apply Full Format Interoperability Wherever Possible

Standardization efforts should apply full format interoperability as far as they can – things should not be done differently just because it is difficult to establish a common agreement. Standards should only accommodate alternative options when there is a valid technical or business reason to do so. Otherwise, the resulting specification may take on so much overhead that it becomes too costly or complex for any set of parties to achieve an interoperable result. The success of the MP3 and MPEG-2 standards is instructive.

2) Do Not Hide Non-Interoperability, but Minimize It

When full interoperability is not possible, this should not be hidden this under layers of abstraction or indirection. They do not bring the solution closer, but will make implementations more costly and less feasible. In particular, the goal should be to put all the non-standard elements in a single, monolithic block, and to make these monolithic blocks as small as possible, leaving very little to be defined elsewhere. Of all the elements left in such a block, DRM interoperability standards should carefully consider whether they can be agreed upon without compromising security – arguably the only valid reason not to agree on a common way to solve a problem. As mentioned earlier, the security of some parts of most DRM systems depends upon diversity and obscurity.

3) Employ Transform Services to Bridge Non-Interoperability

Consumer demand and new business opportunities frequently introduce the need for interaction with other systems built on different agreements and standards. Following from the previous principle, non-interoperable elements should be defined so that transform tasks (*e.g.* formatting, encoding or license transformations) between different systems are as straightforward as possible. This will further interoperability in a number of ways:

- It will reduce the technical issues, thus facilitating design and development of transform services
- It will facilitate the provision of content in different systems simultaneously
- It will facilitate creation of consumer solutions that support multiple systems.

C. Technologies for Full Format Interoperability

We examine a few pieces of the DRM reference model and discuss specific issues around standardization.

1) Transport Formats, Compression, and Bulk Encryption

Music and especially video presentations are large. For the foreseeable future, it will continue to require a noticeable amount of network bandwidth, storage space, and computing power to move, manipulate, or transform media content. Perhaps the greatest advantage from full format interoperability can be had from standardizing processes that directly manipulate content bits:. This includes transport (file) formats, compression (codec) formats, and bulk encryption of media bits, for file download, Internet streaming, and broadband broadcast. Currently, these technologies are often bundled together in proprietary combinations.

2) Content Key Distribution

The security Achilles' heel of most cryptographic systems is key management [18]. While encrypted content may be secured with military-grade algorithms, the content is no more secure than are the content encryption keys. What's worse, content encryption keys are small and can more easily be distributed across the Internet, singly or in bulk, than content itself. While DRM-enabled clients are responsible for verifying usage rules associated with protected content, any security analysis has to consider the possibility that clients will be compromised – which is where the art of designing key distribution schemes comes into play. The goal is to limit the potential damage resulting from a security attack by limiting the value exposed to individual clients. One simple strategy is to encrypt a commercial movie with many different sets of content keys, so that if an attacker obtains keys to one copy of the movie, he cannot reliably redistribute keys to all circulating copies of the same movie. In the extreme case, content keys can be unique for each copy of a digital work.

Standardizing key distribution is problematic. Security demands that key distribution schemes be obscure (so that observers cannot reverse engineer key handling paths), diverse (so that successful attacks on one client do not compromise the entire infrastructure), and renewable (so that compromised systems can recover from attacks). Even the means of association between content keys and content files may be considered sensitive or renewable information.

In addition to keeping content keys confidential, DRM clients must also worry about keeping identity credentials (*e.g.*, private keys) secret. An attacker can obtain content keys if it knows how legitimate clients authenticate themselves, or prove they are legitimate. It benefits attackers if these mechanisms are well known, widespread, and stable.

One notable key distribution scheme that has been proposed for multimedia content protection in home networks is IBM's xCP protocol [19]. xCP uses a symmetric key distribution scheme ("broadcast encryption" [20]) originally developed for multicast applications. Each content key is locked inside a large Media Key Block, which is made available to devices on the network. Each device protects a set of secrets, which it can use to unlock Media Key Blocks and obtain content keys. If a device leaves a network, perhaps by being "revoked", then the revoked device's keys will no longer unlock newly generated Media Key Blocks. Standardized multicast key distribution schemes would be beneficial for distribution of content keys on fixed media, such as CD's or DVD's, since for this distribution mechanism there is no possibility of negotiation between the consuming client and the key provider.

3) Provisioning and Security Services

Every consuming device or application that provides or consumes cryptographic services must be provisioned or initialized with its individual set of secrets, trust anchors, and credentials. Many consuming devices will need to use other security services, in order to receive security upgrades, to refresh keys and certificates, to contact secure time servers, and to report suspected security attacks.

There are two related difficulties with standardizing such security services. The first difficulty relates to client privacy issues. The second difficulty stems from the requirement for clients to authenticate and prove their integrity to remote services. Platform integrity is further related to tamper resistance, which in practice is based on proprietary techniques. New computing platform security and integrity standards [13] may help.

4) Trust Management

No multi-vendor multi-component system can operate without a means for establishing and verifying trust among the components and the entities served by the components. The best known standardized trust management system is the hierarchical PKI model first put forward in the X.509 standard [21]. The basic X.509 system uses hierarchical certifications and anchors trust in a root certification authority. Public-key cryptography provides the technical means of verifying the integrity of certificates.

Trust management decisions must be made on client devices, and may be required more often than content access control decisions. Hence, the complexity of trust management operations is of concern. Public key cryptography operations, especially producing digital signatures, are computationally expensive. Trust models for backend services can be more elaborate, although performance is still an issue there.

Hierarchical trust models with top-level "roots of trust" are convenient for closed-system deployments, but are problematic for dynamic, global deployments. First, a root of trust takes at least delegated responsibility as a certification authority for his descendent entities, but it cannot be expected that a single authority will be competent or willing to take such responsibility universally for all credential-issuing entities. Some entities should be trusted independently of others. This makes sense also when trust relationships are dynamic. Second, a system that relies on a root of trust becomes a slave to its own success. The number of parties relying on the root of trust makes it very difficult for the root to modify its policies and procedures. This seriously inhibits the system from growing and innovating. In the most general case, each stakeholder in a transaction or activity should act as his own ultimate root of trust. He may delegate trust to well-known authorities, but the system should support independent selection of trust authorities and trust anchors [12].

While trust itself can not be standardized, standardization of trust *management* for media DRM should be possible, but the first steps must be identification of a dictionary or vocabulary. It must be decided who needs to be trusted, to perform what activity, under what conditions. This is probably more important to standardize than the exact language for computing decisions based upon trust policy and evidence.

5) Usage Rule Expression

Producers and consumers of content need to share a common license language for expressing the usage rules attached to content. As with trust management, the

standardization of vocabulary and identifiers is probably more crucial than the choice of a specific language. Every DRM system in deployment has a means of expressing usage rules. Probably the simplest is the 4C Entity's Copy Control Information (CCI [22],[23],[24]), used in the 5C DTCP/Firewire access control standard [25] and the DVD/CSS access control system [26]. The Copy Control Information for a digital work consists of two bits, indicating whether the work can be copied one generation, copied never, no more, or freely. A copy of "copy once" content is per force "copy no more" content. "Copy never" content differs from "copy no more" content, in that "copy never" content should only appear in its original form, never in a copied form.

At the other end of the spectrum is the MPEG-21 REL [8], which defines a vocabulary of media-related concepts, and inherits by extension a larger vocabulary through the MPEG-21 Rights Data Dictionary. REL is purely declarative, resembling a logic programming language in that it supports "for all" quantified variables, assertions of fact, and recursive computation. REL supports delegation and chaining of licenses (through the "issue" right), although the way REL chaining interacts with the time line differs from other chained assertion languages like SPKI [27] or X.509. These features give REL much of its expressive power and also much of its complexity. REL's syntax is expressed in XML, which makes typical licenses much larger than the two bits required for 4C Copy Control Information.

There are many other rights expression formats in use. Paul Kocher advocates a system that uses a virtual machine bytecode to express licensing, security, and trust conditions, written against a standard API of host-supplied services [28]. Similar concepts were also explored by the OPIMA standards project [29]. Kocher argues that the delivery of bytecode means that security implementations can be renewed with each new piece of content, but other techniques must still be used to renew the implementation of the virtual machine, and to prevent reverse engineering of the virtual machine.

IV. NEMO – an Approach to Connected Interoperability

Some interoperability problems, especially those involving heterogeneous DRM systems, can be addressed by employing on-line negotiations. This is becoming more feasible in today's climate of ubiquitous always-on or frequently-on network access. NEMO (Networked Environment for Media Orchestration [30]) is an experimental framework being developed at InterTrust for the discovery, access, composition, and orchestration of media-related on-line services.

A. An Introduction to NEMO

NEMO allows service access across multiple network tiers – wide area networks, local area networks, home networks (*e.g.*, over UPnP [31] or Rendezvous [32]), and personal area networks (*e.g.*, over Bluetooth [33]). NEMO supports multiple local and remote interface bindings (*e.g.* WS-I [34], Java RMI, DCOM, C, .Net, etc.) allowing integration with applications. NEMO allows the use of multiple discovery protocols (UDDI [35], JINI [36], UPnP, Rendezvous) for finding NEMO services. NEMO supports (and encourages) the active composition and orchestration of services to perform complex tasks on-line. The idea is that orchestration and composition will allow service configurations to adapt and optimize to changing needs, such as when a personal area network moves into range of new home network devices.

An instance of the NEMO framework consists of a logically connected set of nodes that interact in a peer-to-peer fashion. NEMO nodes interact by making service invocation requests and receiving responses. The format and payload of the request and response messages is defined in XML. The NEMO framework supports the construction of diverse communication patterns ranging from direct interaction with a single service provider to a complex aggregation of a choreographed set of services from multiple service providers. The framework supports the basic mechanisms for using existing service choreography standards and allows service providers to use their own conventions.

A service interface may have one or more service bindings. A NEMO node may invoke the interface of another node as long as that node's interface binding is described and the requesting node can support the conventions and protocols associated with the binding. E.g., if a node supports a web service interface, a requesting node may be required to support SOAP, HTTP, WS-Security, etc. Any service interface may be controlled in a standardized fashion directly providing aspects of rights management. All interactions between NEMO nodes can be viewed as governed operations.

The *Workflow Collator (WFC)* helps fulfill most non-trivial NEMO service requests by coordinating the flow of events of a request, managing any associated data including transient and intermediate results, and enforcing the rules associated with fulfillment. Other examples of this type of functionality can be seen in the form of transaction coordinators ranging from simple transaction monitors in relational databases to more generalized monitors as seen in Microsoft MTS/COM+. The Workflow Collator is a programmable mechanism through which NEMO nodes orchestrate the processing and fulfillment of service invocations.

We use the concept of profiles as an organizing principle in NEMO. A profile is the set of thematically related data types and interfaces defined in WSDL for the NEMO framework. Currently we have two profile categories: "Core", which includes the foundational set of data types and service messages necessary to support fundamental NEMO framework interaction patterns and functionality, and "DRM" which describes the Digital Rights Management services that can be realized with NEMO.

Some services defined in the NEMO Core profile include:

- Authorization – services related to authorization of a participant (such as node) to use a resource (service).
- Peer Discovery – services related to the discovery of NEMO nodes.
- Notification – services related to the delivery of targeted

messages to a given set of nodes.
- Service Discovery – services related to the discovery of services provided by some set of service providing nodes.

Some basic services defined in the NEMO DRM profile include:
- Provisioning – services for obtaining the necessary credentials, policy, and other objects necessary for a consumer electronics device, software application, etc to participate within a specific context that uses DRM.
- Licensing – services for obtaining DRM licenses.
- Membership – services for obtaining objects that establish membership within some designated domain.

We have not yet tried to define a formal categorization of NEMO peer roles based on service type groupings. However, based on existing functionality and observed patterns we have defined a preliminary set of roles that may be formalized over time. These include:

Client - this is the simplest role where no services are exposed and the peer simply uses services of other peers.

Authorizer - this role denotes a peer acting as a Policy Decision Point (PDP) determining if the requesting principal has access to a specified resource with a given set of pre-conditions and post-conditions (consequences, obligations) [37].

Gateway - in certain situations a peer may not be able to directly discover or interact with other service providers, for reasons including: transport protocol incompatibility, inability to negotiate a trusted context, or lack of the processing capability to create and process the necessary messages associated with a given service. The Gateway role denotes a peer acting as a bridge to another peer in order to allow interaction with a service provider. From the perspective of identity and establishing an authorized and trusted context for operation, the requesting peer may actually delegate to the Gateway peer its identity and allow that peer to negotiate and make decisions on its behalf. Alternatively, the Gateway peer may act as a simple relay point forwarding or routing requests and responses.

Orchestrator - in situations where interaction with a set of service providers involves some type of non-trivial coordination of services possibly including transactions, distributed state management, *etc*, it may be beyond a peer's capability to participate in such a scenario. The Orchestrator role is a specialization of the Gateway role. A peer requests an Orchestrator peer to act on its behalf, intervening with one or more services. The orchestrating peer may use certain additional NEMO components such as an appropriately configured Workflow Collator in order to satisfy the orchestration requirements.

B. Consumer Media Applications

Since our ultimate goal is to enable the oft-repeated "instant gratification of request for any media, in any format, from any source, to any place, at anytime, on any device complying with any agreeable set of usage rules," we developed an informal model that helps us demonstrate how we use NEMO to achieve this goal. This model helped us in the separation of concerns process in system architecture discovery. The model is roughly aligned with the Basic DRM Reference Model outlined previously, but is more specialized for our networked application. The model spans heterogeneous network tiers, and illustrates some realistic interoperability problems. We explain the highest level of the model, and then we show how NEMO allows low level services from different tiers in the model to be assembled into richer end-to-end media services

1) A Media Distribution Model

In this model there are four tiers of service components:

1) Content Authoring, Assembly, and Packaging services,

2) Web-based Content Aggregation and Distribution services,

3) Home Gateway services, and

4) Consumer Electronics (CE) devices.

Each of these four tiers clearly has significantly different requirements for security, rights management, service discovery, service orchestration, user interface complexity, and other service attributes. The first two tiers fit very roughly into the models that we see for "traditional" web services, while the last two tiers fit more into what we might call a personal logical network model, with certain services of the home gateway being at the nexus between the two types of models. However, services of CE devices could occasionally appear in any of the tiers. Thus, we have the dilemma where we want to specialize parts of the framework for efficiency of implementation, while being general enough to encompass an end-to-end solution.

For relatively static and centralized web services, a UDDI directory and discovery approach may work well, but for a more dynamic transient merging of personal networks, discovery models such as found in UPnP and Rendezvous are more appropriate. Thus, we need to be able to include multiple discovery standards in our framework.

When rights management is used for media distribution through wholesale, aggregator, and retail distribution subtiers, there can be many different types of complex rights and obligations that need to be expressed and tracked. This requires a highly expressive and complex rights language, sophisticated content governance and clearing services, and a global trust model. However, rights management and content governance for the home gateway and CE device tier generally requires a different trust model and needs to emphasize fair use rights that are straightforward from the consumer's point of view. Peer devices in a personal logical network want to interact using the simple trust model of that network, and they need to interact with peers across a wide area network using a global trust model perhaps through proxy gateway services.

At the consumer end, complexity arises from automated management of content availability across devices, some of which are mobile and intermittently intersect multiple networks. Thus, our approach to rights management, while

enabling end-to-end distribution, is heterogeneous, supporting a variety of rights management services, including services that interpret distribution rights expressions and translate them, in context, to individual consumer fair use rights in a transaction that is orchestrated with a sales transaction or another event where a subscription right is exercised.

2) NEMO Solutions

We are currently using NEMO to link various consumer devices to a number of different services in the multi-tiered environment described above. We have successfully demonstrated interoperability in one interconnected system using cell phones, game platforms, PDAs, PCs, web-based content services, discovery services, notification services, and update services. We support multiple media formats (*e.g.* MP4, Windows Media, and others), multiple discovery protocols (over Bluetooth and through registries such as UDDI, LDAP, and Microsoft Active Directory), and IP-based notification and wireless SMS notification on the same device. We use the orchestration feature to help the consumer overcome interoperability barriers. When there is a query for content, orchestration coordinates the required services in order to fulfill the request, including, discovery, search, matching, update, rights exchange, and notification services.

We are attempting to converge on a state where a consumer can use most any device, make a wish for content, and be instantly fulfilled with the content and the rights and rendering capabilities (within obvious limits of the hardware) to both use and share the content. The orchestration capability allows the consumer to view all home and internet-based content caches from any device at any point in a dynamic multi-tiered network. We are extending this capability to more advanced services that promote sharing of streams and play lists, making impromptu broadcasts and narrowcasts easy to discover and connect to, using many different devices, while ensuring rights are respected.

3) Connected Interoperability

Beyond the consumer-centric side, we are looking at ways to provide an end-to-end interoperable media distribution system that does not rely on a single set of standards for media format, rights management, and fulfillment protocols. Value chains that include content originators, distributors, retailers, service providers, device manufacturers, and consumers, exhibit a number of localized needs in each segment. This is especially true in the case of rights management, where content originators need to express rights of use that may apply differently in various contexts to different downstream value chain elements. A consumer gateway has a much more narrow set of concerns, and an end user device has a yet simpler set of concerns, namely just playing the content.

With a sufficiently automated system of dynamically self-configuring distribution services, content originators can produce and package content, express rights, and confidently rely on value added by other service providers to instantly provide the content to all interested consumers, no matter where they are or what kind of device they are using. We use NEMO to fulfill this goal and provide means for multiple service providers to innovate and introduce new services that benefit both consumers and service providers without having to wait for or depend on a monolithic set of standards.

This approach allows digital rights management to be decomposed into components with a more natural separation of concerns that focus on policy management of service interfaces rather than on copy protection. This has the potential to change the tension between consumers and content providers in the digital content domain as the NEMO enriched infrastructure provides consumers with better information, more useful services and instant gratification. Policy management can limit the extent to which pirates can leverage those legitimate services. NEMO allows the network effect to encourage the evolution of a very rich set of legitimate services providing better value than pirates can provide.+

V. Conclusions

We have argued that interoperability is very important to the success of DRM. We have explained the concepts behind digital rights management, and provided a basic reference model that embodies patterns representative of current architectures, and which can serve as a basis for coordination of interoperability solutions in next generation DRM systems. We have enumerated the pros and cons of three separate approaches to DRM interoperability, and discussed issues related to standardization of specific DRM-related technologies. We have presented an experimental system that supports interoperability of heterogeneous DRM systems via on-line services and service orchestration.

Looking forward, DRM systems must continue to evolve in order to achieve interoperable, secure, media-related ecommerce in a world of heterogeneous consumer devices, media formats, communication protocols and security mechanisms. Today significant barriers exist to the goal of an interoperable and secure world of media related services. Standardization can solve some interoperability problems, but standards are not always universally applied. Where heterogeneous systems exist, dynamic late-bound network services can supply required functionality, including bridging services. But, in the end, DRM is about protection. A DRM system will not interoperate if it does not *want* to.

Acknowledgment

The authors gratefully acknowledge valuable discussions and input from David Maher, William Bradley, Talal Shamoon, and Albhy Galuten.

References

[1] G.A. Lyon, “A Quick-Reference List of Organizations and Standards for Digital Rights Management”, *NIST Special Publication 500-241*, October 2002.

[2] J. Borland, “Apple Unveils Music Store”, in *CNET News.com*, April 28, 2003. http://news.com.com/2100-1027_3-998590.html?tag=rn

[3] J. Graham, "Downloaders dance to Apple's iTunes", in *USA Today*, December 15, 2003. http://www.usatoday.com/tech/news/2003-12-14-apple2_x.htm

[4] DRM Watch Staff, "More Paid Download Music Services to Launch in Early 2004", *DRM Watch*, December 11, 2003. http://www.drmwatch.com/ocr/article.php/3287471

[5] Esprit WP4, *The IMPRIMATUR Business Model Version 2.1* http://www.imprimatur.net/IMP_FTP/BMv2.pdf

[6] N. Friesen, *Towards a Digital Rights Expression Language Standard for Learning Technology*,

[7] O. Sibert, et. al., *Securing the Content and Not the Wire*. Technical Report, InterTrust Technologies, Inc. 1996.

[8] ISO/IEC JTC1 SC29 WG11 (MPEG),*ISO/IEC 21000-4 Rights Expression Language*, to be published by ISO in 2004

[9] R. Ianella, *Open Digital Rights Language Specification v 1.0*, see www.w3.org/TR/odrl/

[10] Content Reference Forum, *Content Reference Forum Introduction.* CRF-004, 2003. http://www.crforum.org/crfreppub/CRF004_002_cr_forum_overview.pdf

[11] J. Feigenbaum, et. al., *Privacy Engineering for Digital Rights Management Systems.* http://citeseer.nj.nec.com/cache/papers/cs/25597/http:zSzzSzwww.star-lab.comzSzsanderzSzpublicationszSzspdrm01.pdf/feigenbaum01privacy.pdf

[12] S. Weeks, et. al., *Understanding Trust Management Systems*. InterTrust STAR Lab, Technical Report STAR-TR-01-02, March, 2001.

[13] Trusted Computing Group, *TCG Main Specification, Version 1.1a.* September 2001. https://www.trustedcomputinggroup.org/home

[14] B. Horne, et. al.. "Dynamic self-checking techniques for improved tamper-resistance", in *Proceedings of Workshop on Security and Privacy in Digital Rights Management 2001*, Association of Computing Machinery. http://citeseer.nj.nec.com/horne01dynamic.html

[15] W. Shapiro, et. al., *How to Manage Persistent State in DRM Systems*, InterTrust STAR Lab, Technical Report STAR-TR-01-06, August, 2001.

[16] Chr. Barlas (ed.), *Digital Rights Management Final Report*, http://europa.eu.int/comm/enterprise/ict/policy/doc/drm.pdf

[17] ISO/IEC JTC1 SC29 WG11 (MPEG), *ISO/IEC-14496-13 Final Draft International Standard*, (MPEG-4 IPMP Extensions).

[18] B. Schneier, *Applied Cryptography Second Edition*, John Wiley & Sons, Inc., 1996.

[19] F. Prestoni, *xCP Cluster Protocol – IBM Response to DVB-CPT Call for Proposals for Content Protection & Copy Management*, International Business Machines Corporation, October, 2001, http://www.almaden.ibm.com/software/ds/ContentAssurance/prez/xCP_DVB.ppt

[20] A. Fiat, and M. Naor, "Broadcast Encryption", in *Advances in Cryptology* --- CRYPTO '93, Springer-Verlag, Berlin, 1994, pp. 480--491. 16

[21] International Telecommunications *Union. ITU-T recommendation X.509 (08/97) – information technology – open systems interconnection – the directory: Authentication framework*, August 1997.

[22] M. Ripley, C. Traw, S. Brendan, S. Balogh, and M. Reed, "Content Protection in the Digital Home", in *Intel Technology Journal*, November 2002. http://developer.intel.com/technology/itj/2002/volume06issue04/

[23] 4C Entity LLC, *Content Protection System Architecture: A Comprehensive Framework for Content Protection*, *Revision 0.81*, February 2000, http://www.4centity.com/data/tech/cpsa/cpsa081.pdf.

[24] J. A. Bloom, I. J. Cox, T. Kalker, J.-P. Linnartz, M. L. Miller, and B. Traw. Copy protection for DVD video. *Proceedings of the IEEE*, 87(7): 1267—1276, 1999. http://citeseer.nj.nec.com/bloom99copy.html.

[25] *5C Digital Transmission Content Protection White Paper*, *Revision 1.0*, July 1998, http://www.dtcp.com/data/wp_spec.pdf

[26] DVD Copy Control Association, http://www.dvdcca.org

[27] C.M.,Ellison, B. Frantz, B. Lampson, R.L. Rivest, B.M. Thomas, and T. Ylonen, "SPKI certificate theory". *IETF RFC 2693*, September 1999.

[28] P. Kocher, J. Jaffe, B. Jun, N. Lawson, *Self-Protecting Digital Content*, Cryptography Research, Inc., 2003. http://www.cryptography.com/resources/whitepapers/SelfProtectingContent.pdf

[29] IEC ITA, *OPIMA Specification, Version 1.1, June, 2000*, http://leonardo.telecomitalialab.com/opima/

[30] W. Bradley D. Maher, "The NEMO P2P Service Orchestration Framework", *In Proceedings of the 37th Annual Hawaii International Conference on System Sciences* (HICSS-37), January 2004.

[31] UPnP Forum, Basic Device V 1.0 Profile, *MediaServer V 1.0 and MediaRenderer V 1.0 Profile, Internet Gateway Device (IGD) V 1.0 Profile*, http://www.upnp.org/

[32] Apple/Darwin Group, *Rendezvous*, http://developer.apple.com/darwin/projects/rendezvous/

[33] Bluetooth.org, *Bluetooth Core Specification, 1.0B*, January 2003 , https://www.bluetooth.org/docman2/ViewProperties.php?group_id=53&document_content_id=330

[34] S. Werden, C. Evans, M. Goodner, *WS-I Usage Scenarios*, Web Services Interoperability Organization (WS-I), http://www.ws-i.org/

[35] T. Bellwood, (ed.), *Universal Description, Discovery and Integration (UDDI) V2*, OASIS Standard, 19 July 2002. http://www.uddi.org/specification.html

[36] B. Joy, and J. Waldo, *Jini Network Technology*, Sun Microsystems, March 1999. http://wwws.sun.com/software/jini/

[37] R. Yavatkar, D. Pendarakis, R. Guerin, "A Framework for Policy-based Admission Control," RFC2753, http://www.ietf.org/rfc/rfc2753

Rob Koenen (M'97–SM'00) holds an MS degree in electrical engineering from the Technical University of Delft, the Netherlands, '89.

He works with InterTrust Technologies in Santa Clara, CA, US. Before joining InterTrust, he was a research director at KPN Research in The Netherlands for 10 years. His numerous projects with KPN included: image coding research, audio/visual communication for people with special needs, interactive broadband multimedia for residential users, mobile multimedia, the strategic development of new multimedia services, audio/visual quality assessment and multimedia standardization.

Mr. Koenen is Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology, chairman of MPEG's Requirements Group, and founder and current President of the MPEG Industry Forum, MPEGIF.

Jack Lacy received his MS degrees from the University of Wisconsin in Physics, 1979 and from New York University in Computer Science, 1987.

Prior to joining InterTrust, he spent 18 years as a researcher at Bell Laboratories, Bell Labs and AT&T Labs working in a variety of areas related to networking and computer security, including systems for sending voice over IP networks, cryptography, and secure systems architecture. He is a co-inventor of Cryptolib, a widely distributed cryptographic library, and Policymaker, an AT&T developed approach to specifying and interpreting security policies, credentials, and relationships. Mr. Lacy has also been active in intellectual property protection and management through his involvement in standards setting organizations, such as MPEG, OPIMA and SDMI. He chaired the SDMI Portable Device Working Group from March – September 1999.

At InterTrust he is responsible for media standards activities, technical requirements for advanced development projects, development of system architectures and prototypes, particularly around Media technologies, and determination of InterTrust's interfaces to open technology standards.

Michael MacKay is Executive Vice President for Standards, Policy and Specifications at InterTrust Technologies in Santa Clara, California. He has held several executive engineering positions with InterTrust since 1999. As Senior VP, Engineering he led research and development of the Rights|System, DRM Platform. Prior to joining InterTrust, Mr. MacKay was VP, Corporate Architecture for Novell, Inc. where he led design projects and standards development in support of the NetWare 5 platform. Prior to joining Novell, Mr. MacKay worked for Taligent where he was responsible for class framework architecture. Mr. MacKay also spent 15 years with Xerox Corporation where he worked on a variety of technologies including distributed printing services, structured document-processing systems, printing languages, and a variety of printing systems and services products.

He has been a contributor in multiple standards bodies including IETF, DMTF, and ISO, and is a former member of the board of directors for the Object Management Group. Mr. MacKay is a member of the ACM.

Steve Mitchell (M'84) (b. Toronto, 1961) BS electrical engineering 1984 (Georgia Tech), MS applied mathematics 1986 (Georgia Tech), MS computer science 1989 (Cornell U.) has studied theory of computation, signal processing, and bioacoustics.

Since 1999 he has worked as a member of the technical staff of InterTrust Technologies in Santa Clara, California, on projects relating to multimedia content protection, enterprise policy administration and analysis, and digital multimedia watermarking.

Mr. Mitchell is a member of several professional societies, including the IEEE.