

Forming a Coral Ecosystem

October 2007



THIS DOCUMENT IS PROVIDED "AS IS". THE CORAL CONSORTIUM CORPORATION ON BEHALF OF ITSELF AND ITS MEMBERS MAKES NO REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY OTHER WARRANTY CONCERNING THE COMPLETENESS, ACCURACY, OR APPLICABILITY OF ANY INFORMATION CONTAINED IN THIS DOCUMENT AND DISCLAIMS ALL LIABILITY OF ANY KIND WHATSOEVER, EXPRESS OR IMPLIED, ARISING OUT OF OR RESULTING FROM THE RELIANCE OR USE BY ANY PARTY OF THIS DOCUMENT OR ANY INFORMATION CONTAINED HEREIN.

THE CORAL CONSORTIUM CORPORATION ON BEHALF OF ITSELF AND ITS MEMBERS MAKES NO REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, CONCERNING THE APPLICABILITY OF ANY PATENT, COPYRIGHT OR OTHER PROPRIETARY RIGHT OF ANY THIRD PARTY TO THIS DOCUMENT OR ITS USE, AND THE RECEIPT OR ANY USE OF THIS DOCUMENT OR ITS CONTENTS DOES NOT IN ANY WAY CREATE BY IMPLICATION, ESTOPPEL OR OTHERWISE, ANY LICENSE OR RIGHT TO OR UNDER ANY CORAL CONSORTIUM CORPORATION MEMBER COMPANY'S PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS.

Copyright © 2007 Coral Consortium Corporation. All Rights Reserved.

Coral Consortium Corporation
48377 Fremont Blvd., Suite 117
Fremont, CA 94538
USA

Website: <http://www.coral-interop.org>

Email: info@coral-interop.org

0 Table of Contents

0	Table of Contents	2
1	Overview.....	3
2	Ecosystem Formation	5
3	Ecosystem Specification.....	6
3.1	Content Usage Model	6
3.2	Ecosystem Design	7
3.3	DRM selection.....	8
4	Ecosystem Trust Management.....	9
5	Ecosystem Legal Support	9
6	An Ecosystem Example.....	10
6.1	Ecosystem Model	10
6.2	Licensing Use of Coral Specifications	10
6.3	Ecosystem Specification	10
6.4	Creation of Ecosystem Legal Agreements.....	14

1 Overview

The Coral Consortium has created a framework that allows multiple different DRM systems to coexist in a way that can provide consumers with the kind of interoperability that they have come to expect of digital content distribution systems. In order to create such a framework, Coral has:

- Created a set of specifications that allow trusted, secure communications between different entities in the digital content distribution value chain
- Identified the home network and rights management functionalities that appear in media value chains and defined a set of standard interfaces between them
- Created a strategy for integrating DRM systems with the interoperability framework that can be implemented with minimal (if any) changes to the DRM systems themselves.

The technical components are described in detail in the Coral Consortium specifications, available from the Coral Consortium website, www.coral-interop.org. There are three parts to the specifications¹:

- The NEMO Specifications
 - Coral's trusted communication and authorization framework
- The Coral Core Architecture Specification
 - Fundamental functions and data structures used to build interoperable solutions.
- The Coral Domain Architecture Specification
 - Tools for supporting interoperable domains of consumer devices.

The central data element of the Coral solution is the Rights Token. Very simply stated, a Rights Token is a DRM-agnostic association of a content identifier, content usage model identifier, and a consumer identifier. Rights Tokens are converted into DRM-specific licenses that are managed by the participating DRM systems themselves. The rest of the Coral system focuses on managing the consumer's personal media Ecosystem of devices, service providers and content and defining their domain of devices, using Rights Tokens to represent their media rights and Rights Lockers² to enable persistence of those rights.

In general, Rights Tokens and any Coral interfaces are required only when communication occurs between Coral roles that are deployed within different security boundaries. For example, a single service provider may deploy Rights Locker, Domain Manager, and License Issuer roles within a well-defined security boundary. In this case, consumer content transactions need not be stored as Rights Tokens and communication

¹ These are the specifications as released on the Coral website in the Fall of 2007.

² A Rights Locker – defined in the Coral Domain Specification – is typically an online service that provides login for a consumer, storage of rights acquired by that consumer, and potentially a point of purchase.

among the various internal roles need not use Coral interfaces and trusted communication protocols to interact. If multiple Rights Locker, Domain Manager, License Issuer deployments need to communicate or share consumers' rights information, such as when a consumer seeks to change from one Rights Locker / Domain Manager Service Provider to another, the consumer's rights information is packaged as Rights Tokens and transferred using the standard Coral interfaces.

The Coral specification stack includes the 3 parts labeled 'Coral' as depicted below in Figure 1³.

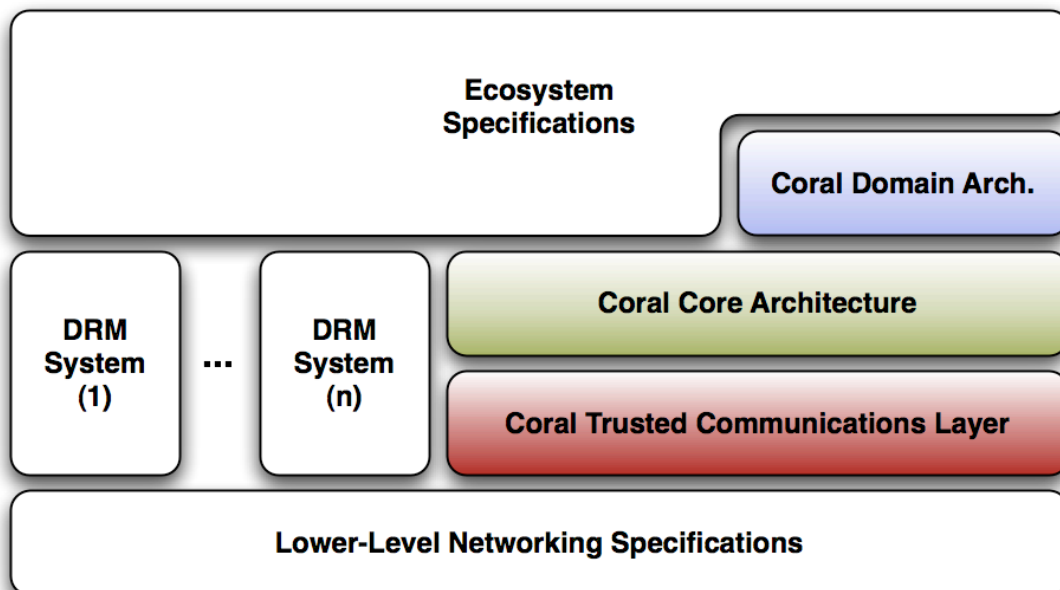


Figure 1 Coral Specifications Stack

These specifications define a generic framework that can be used to construct *Ecosystems* of devices, services, and content that operate under a well-defined set of usage models and that support many different DRM and content protection systems. The Coral framework may also be used to provide interoperability between such Ecosystems. The Coral Core Architecture is a necessary but not sufficient building block for constructing such Ecosystems — it does not, for example, specify or otherwise constrain the content usage models or policies that are necessary for interoperability. Rather, the Coral Core Architecture provides an underlying infrastructure that enables Ecosystem-specific policies to be built on top of multiple DRM systems with seamless interoperability. In addition to the Core Architecture Specification, Coral has also created a Domain

³ Coral currently supports a set of trusted communication protocols called NEMO. Coral is currently working to support additional protocols that maintain the same trust model but that may be more appropriate for different ecosystems of devices, service providers and content. For the rest of this document we refer to Coral trusted communication protocols with reference to NEMO as an example.

Architecture Specification that can be used by Ecosystem architects to enable the Ecosystem consumers to access their content on all of their devices – their device Domain – regardless of the DRM supported by those devices.

A Coral Ecosystem defines a unique configuration of the tools that are provided by the Coral Core Architecture and Domain Architecture, with enough detail to allow fully interoperable implementations. The Coral Specifications list and describe the elements that must be defined by a Coral Ecosystem to ensure interoperability. Ecosystems should define specifications and/or rules and policies that include the elements necessary for their Ecosystem. A Coral Ecosystem specification, in conjunction with the Coral Core Architecture, should comprise a completely interoperable set of specifications. That is, all independent implementations built according to that Ecosystem’s specifications will be able to interoperate if that Ecosystem’s policies allow.

This document explains the process of creating a Coral Ecosystem and its associated specifications, the legal relationship between Coral and Ecosystems, the technical and policy choices that must be made, and the support infrastructure that must be provided by Ecosystem Founders to their adopters. Finally, we provide an overview of an example Ecosystem.

2 Ecosystem Formation

Coral Ecosystems are expected to be part of content distribution solutions created or founded by one or more companies interested in supporting multiple DRMs. This entity may include combinations of content companies, service providers, infrastructure providers, and device manufacturers. For this document, we refer to this Ecosystem founding group as the Ecosystem Founders or simply, the Founders. Each group of Founders will define the rules of its Ecosystem and use the Coral specifications to make its Ecosystem interoperable across DRMs. Typically, the following steps must be followed to create an Ecosystem.

1. The Ecosystem Founders, acting as a business entity, define a content distribution model that includes interoperable support for multiple DRMs and for optional device domains. Such an arrangement might be an individual company, a Joint Venture (JV), or other legal entity.
2. The Founders may initially acquire the Coral Specifications under the click-through Coral Evaluation Agreement to begin the process of defining their Ecosystem.
3. After evaluating the Coral Specifications, the Founders enter into an agreement with Coral to formally acquire the Coral Specifications for the purpose of implementing and deploying a Coral Ecosystem and for the right to sub-license the Coral Specifications to adopters of this Ecosystem. This agreement – called the *Coral Ecosystem Agreement* – is available on the Coral website.
4. The Founders acquire the Coral Specifications.
5. The Founders create their Ecosystem Specifications. The Ecosystem Specifications describe how Coral Specifications are used within this particular

Ecosystem and define various content usage models, services, and associated parameters.

6. The Founders then license the Ecosystem Specifications and sublicense the Coral Specifications to their Ecosystem-specific adopters under an Ecosystem-specific adopter agreement (Ecosystem Adopter Agreement). This agreement, created by the Ecosystem Founders, will include a limited set of terms that must be passed on to all adopters of Coral-based Ecosystems. These mandatory terms are specified in the *Coral Ecosystem Agreement*. The Ecosystem Adopter Agreement may also include other terms that are suggested by Coral in a document called *Model Ecosystem Adopter Terms*, also provided on the Coral website.

Coral's responsibilities include updating and maintaining the specifications and any changes to them. Changes can be requested by the Ecosystem Founders on behalf of their adopters. Minor changes such as bug fixes and typos require only notification of Coral. Material changes must be authorized by Coral. Coral provides a template for change requests. The *Coral Ecosystem Agreement* defines and specifies the agreed protocols associated with both material and non-material changes to the specification. Coral remains open to new members, and Ecosystem Founders as well as adopters could choose to become a Coral member to work on any improvement.

3 Ecosystem Specification

Once the specifications are acquired from Coral, the Founders must design their Ecosystem and create their Ecosystem specifications. This includes the following steps.

3.1 Content Usage Model

The Ecosystem Founders define the way that content acquired within their Ecosystem may be used by consumers – the Ecosystem Usage Rules. This includes specifying the model under which content may be distributed and consumed. For example, the content distribution model may include one or more of the following models:

- A rental model, in which content access times out when the rental period expires,
- A subscription model, in which content is accessible for as long as the subscription is valid,
- A download to own model, in which content is accessible forever,
- Additional models.

The usage or consumption model specifies further whether or not content may be used on multiple devices – such as in a Domain – or whether use is restricted to a single device at a time.

Ultimately, when each group of Ecosystem Founders selects the DRMs for their Ecosystem, they must determine the degree to which those DRMs are capable of supporting and enforcing the selected model(s) and must define the compliance rules associated with the different elements of the system.

3.1.1 Domain Policy

The Coral Domain Architecture Specification provides support for domains that include some number of a consumer's devices – each of which may support a different (Ecosystem-supported) DRM. In particular, the Coral Domain Architecture Specification allows the domain parameters to be defined explicitly. Ecosystem Founders must choose whether or not to support domains. If such support is desired, the Founders must specify domain policy. Such policy includes:

- Number of devices allowed in a domain
- Number of domain memberships allowed per device
 - Can a device participate in more than one Ecosystem domain?
- Authentication policy
 - Must users authenticate themselves to the Domain Manager when
 - Adding or subtracting devices?
 - Interacting with the Domain Manager in any other way?
 - If so, what types of authentication are permissible?
- Proximity rules
 - Should they exist?
 - If so, what are they?
 - If users are local to their home environment, do authentication requirements change?
- May Ecosystem content play on all domain devices without restriction?
 - For example, for sell-through content it may be allowed that all content may be accessed on all of the devices in a user's domain. For content acquired under a rental model, there may be a limit to the number of devices on which the content may be accessed.

3.2 Ecosystem Design

Once the Ecosystem content usage models and domain policies are determined, Ecosystem Founders must decide which aspects of the Coral Specifications are required to support those models and policies. These decisions include:

- Are domains and Rights Lockers supported?⁴
 - If so, does the Ecosystem support multiple Domain Managers / Rights Lockers?
 - If so, can users change Domain Manager / Rights Locker service providers?
 - Must domain client devices contain Coral components or is it enough that they support Ecosystem-specified DRMs?
- What other Coral Roles are required for this Ecosystem?
- Which Roles are online, which are local, which may be either local or online, and which are not relevant to this specific Ecosystem?
- Is it required that some Roles be bundled in combination with other Roles?

⁴ These two Coral Roles are very obvious roles to consider but this is not to suggest that they are particularly special. Coral defines a number of Roles in the Coral Specifications; all of these must be considered carefully when designing an Ecosystem.

- Are some Roles deployed as standalone services?
- Are some Roles integrated with retail services?
- Are some Roles embedded in portable devices or general-purpose computers?
- Which policy variables are defined for the Ecosystem roles? Are some variables fixed? Are some static and only updatable via a system upgrade (e.g. hardware/firmware)? Are some of them dynamically updatable by an Ecosystem policy update service? How often?
- How are Ecosystem usage models mapped onto specific DRMs? Who determines whether those mappings are sufficient? Does the Ecosystem mandate some mappings? Is it up to the adopters/implementers?

The Ecosystem design process must also determine whether or not there are Ecosystem elements that will need to communicate using the Coral trusted communication protocols – that is, which, if any, elements must establish communication across trust boundaries. For example, if the Ecosystem includes more than one Domain Manager service, they may need to communicate. If so, they would do so using the Coral trusted communication protocols and each side would require associated credentials. The process associated with acquiring such credentials is discussed later in this document.

Other Roles that must be supported are associated with creating and managing Rights Tokens as well as deriving DRM-specific licenses from Rights Tokens. All of these Roles are described in detail in the Coral Architecture Specifications.

3.3 DRM selection

The Ecosystem Founders must decide which DRMs to support. As previously stated, Coral has specified an interoperability framework that supports development of Ecosystems that enable devices to acquire and access content from a set of content distribution services such that the content will play on all of those devices regardless of the DRM used by the device, as long as the DRM is one of those supported by the Ecosystem. The process of choosing DRMs is a combination of determining whether or not specific DRMs are capable of supporting the Ecosystem usage models and whether or not the DRMs are trusted by the Ecosystem Founders to meet their content protection requirements.

Once the DRMs are chosen, the Founders will need to determine policies for the ways that each usage model specified for the Ecosystem maps onto the different approved DRMs, and with what actual parameters. The License Server will implement these policies.⁵ The License Server is the role that is responsible for converting Rights Tokens to DRM-specific licenses when a consumer requests content for a device that supports that DRM. The License Server role may be deployed as part of a Rights Locker service. Alternatively, the License Server can be deployed as part of a content distribution retail service. The use of the rights token separates the acquisition of rights from the rights fulfillment events, which makes it possible to add support for new DRMs at any time.

⁵ As discussed in the Coral Core Architecture Specification, a single License Server may support multiple different DRMs.

4 Ecosystem Trust Management

Any Coral Role within an Ecosystem that must communicate across trust boundaries with other Coral Roles must support Coral's trusted communication protocols and therefore acquire related trust credentials. Coral-based Ecosystems must therefore support the creation and management of these credentials, which include two public key pairs – one for message signing and one for message encryption – and their associated public key certificates, as well as authorization statements for the Roles held by the node, signed by an Ecosystem-appointed authority (e.g., SAML assertions [SAML]). Once the elements of the Ecosystem are determined, the Ecosystem Founders must decide how, by whom, and under what circumstances credentials will be generated and managed.

Before Ecosystem credentials are granted to a participant implementation, the implementation must be certified to be compliant with Ecosystem compliance and robustness rules, which include rules stating that Coral portions of the implementations are compliant with the Coral Specifications. Such certification is determined according to procedures defined by the Ecosystem. For example, the implementer may use a suite of conformance tests provided by the Ecosystem Founder to determine conformance with both the Coral and Ecosystem specifications. Once successful execution of such tests is completed, the implementer or the certification authority signs a form attesting to the success. Certification with compliance and robustness rules may be tested by the implementer but can also be done by a third party organization. Decisions about self-certification and conformance testing are made by the Ecosystem Founders.

The Ecosystem Founders specify the credential generation procedure and entities responsible for credential generation and management. For some Ecosystems it may be adequate for the Ecosystem to create its own root certification authority and role authorization assertion generation and signing keys and to have those signed by a generic certification authority. For others, certified key and assertion management may be better handled entirely by an external organization on behalf of the Ecosystem. The Ecosystem certification authority is also responsible for managing Ecosystem credential revocation and renewal. Revocation may be done either using certificate revocation lists (CRLs) or online certificate status protocol (OCSP). Renewability strategy and policy must be formed by the Ecosystem Founders together with individual participating device makers and service providers.

While it is generally assumed that the scope of interoperability is within the bounds of a specific Ecosystem, if the Ecosystem Founders determine that they want to interoperate with other Coral-based Ecosystems, they may further decide to associate with a common certification authority, to cross-certify, or for each node to hold a trust anchor from each ecosystem.

5 Ecosystem Legal Support

Ecosystem Founders must create Ecosystem Adopter Agreements and optionally, Content Participant Agreements, if content companies participate in the Ecosystem. These must include terms from the *Sublicensing Authority* section of the [Coral Ecosystem Agreement](#).

The Ecosystem Adopter and Content Participant agreements may include additional Ecosystem-specific terms – some of which may be taken from the [Model Ecosystem Adopter Terms](#) – as well as an encoding of all Ecosystem-specific policies specified in section 3. Last, the Ecosystem Founders must specify policy (as well as an entity for implementing that policy) for specification change management as well as software and firmware renewability⁶.

6 An Ecosystem Example

6.1 Ecosystem Model

In this example there is a single Founder. This company (“Founder”) initiates a high-level Ecosystem design based on a download-to-own content distribution model, support for multiple DRMs, access to content on any device in a consumer’s home – i.e. Domain – that supports these DRMs. The Founder seeks to provide Domain Manager and Rights Locker services to participating retailers and consumers. Adopters are expected to include retail service providers, Domain Manager and Rights Locker service providers, content providers, and device manufacturers⁷.

Within the Founder’s Ecosystem, elements that need to communicate will want to do so across standard interfaces; this is critical if different service providers within the Ecosystem are competitors, which will surely happen if there are multiple retailers, Domain Managers, and Rights Lockers. Additionally, the Founder is aware that some of their participants – most notably retailers, service providers and device manufacturers – may participate in other Coral-based Ecosystems (e.g., a rental model) and would not want to have to adapt their devices and services differently for each Ecosystem.

6.2 Licensing Use of Coral Specifications

The Founder acquires the Coral Specifications under the Coral Ecosystem Agreement, which also gives the Founder the right to sublicense those specifications to the adopters of the Founder’s Ecosystem.

6.3 Ecosystem Specification

6.3.1 Statement of a typical Ecosystem Model in Coral Terms

The Ecosystem includes content retailers and their respective backend service providers and associated transaction databases as well as consumer devices that participate in the Ecosystem Domains. Each backend system will support Coral Domain Manager and Rights Locker functionality. Consumers will have one or more devices, each of which supports an Ecosystem-approved DRM. The consumer will establish a Domain with one

⁶ Software and firmware renewability are typically the responsibility of the Ecosystem Adopters themselves.

⁷ As stated earlier, not all devices will need to implement Coral roles. In particular, devices that use DRMs that support a DRM-specific domain model will likely require little, if any, Coral technology.

Domain Manager Service Provider. The consumer will establish accounts with multiple Retailers or Rights Lockers – typically one per Retailer.

A consumer will initiate a content transaction via a web-based interaction with a participating retailer and the transaction information will be stored in the associated backend transaction database. Upon verifying that the client is authorized to obtain a DRM license (e.g., by checking that it is a Domain member), the backend system returns a license trigger to the consumer. The consumer's client redeems the trigger by engaging in a DRM-specific protocol with the License Server that generated the trigger.

Any communication between Rights Lockers or Domain Managers that must occur across a trusted boundary will take place using Coral data structures (e.g., Rights Tokens) and trusted interfaces. Figure 2 below depicts the basic Roles and relationships for this Ecosystem.

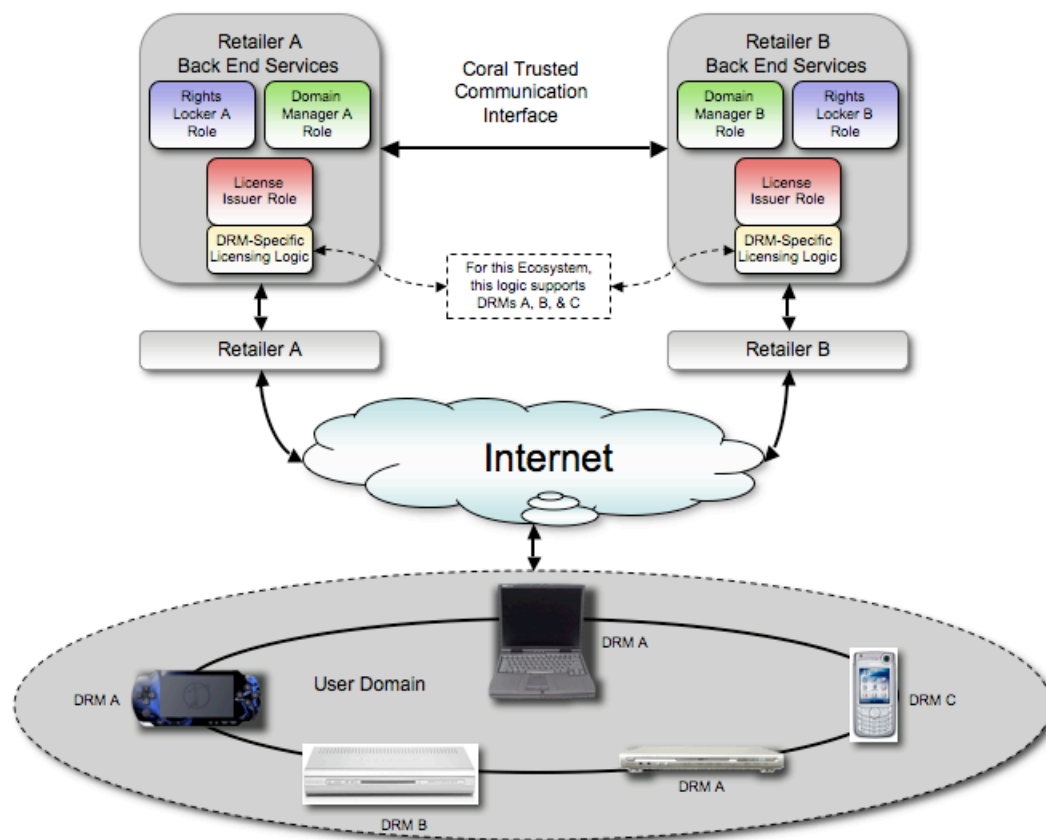


Figure 2

6.3.2 Choosing DRMs

The Founder specifies the DRMs to be used in the Ecosystem based on the following criteria:

- This decision is based in part on capabilities of different DRMs and their ability to support the Ecosystem usage model.
- It is also based in part on the degree to which content providers trust the DRMs' ability to mitigate risk (financial stability of the DRM provider, its ability to respond to breaches, etc.) DRMs must also be analyzed with respect to their ability to interface with Coral Services (e.g., Domain ID support, unique device IDs, secure clocks, etc.).

6.3.3 Specification of Ecosystem Policies and Underlying Processes

The Founder then specifies content usage and device policies and procedures in terms of Coral concepts. This process involves answering questions such as the following.

1. How and under what terms is content acquired from participating retailers / service providers?
 - a. Purchase to own
 - b. Rental
 - c. Subscription
 - d. Etc.
2. How can consumers establish and use their Domains?
 - a. Is there more than one Domain Manager envisioned but not initially required?
 - b. If the use of content is allowed on all devices in consumer's Coral Domain, what are the parameters of that Domain?
 - i. Size of Domain = N devices
 - ii. Other Domain policy
3. How do consumers establish accounts with a Rights Locker Service Provider?
 - a. Is there more than one Rights Locker envisioned but not initially required? (Domain Manager and Rights Locker Service Providers can be the same service provider but it is not required)
 - b. Are there policies to which Service Providers must adhere:
 - i. To provide a good Ecosystem-wide user proposition?
 - ii. To mitigate abuse?
4. What is the content licensing model?
 - a. When a consumer purchases a piece of content, Retailers communicate related transaction metadata to the consumer's Rights Locker Account using a standard format over standardized interfaces⁸. This information is formatted in the standard Rights Token format when it is communicated between services for any reason.
 - b. Consumer acquires DRM-specific licenses for their devices via interaction with the Domain Manager and Rights Locker
 - i. These DRM-specific licenses may be bound to the devices directly, or may be bound to a native DRM domain of which the device is a member. That is, the DRM itself may support domains, and so the DRM-specific licenses would be bound to the DRM-specific domain.

⁸ This account will typically be the same account as maintained by the Retailer in their retail transactions database.

- ii. The ability to bind to a native DRM domain may provide the opportunity for the Ecosystem design to dispense with much of the client-side Coral technology, because distributing a domain-bound license to a device does not require secure information about the device identity at license creation.

6.3.4 Determination of Required Coral Roles

Roles that are required by the Ecosystem specification include the following:

1. Domain Manager
 - a. Implemented by Domain Manager Service Provider
2. Rights Locker
 - a. Implemented by Rights Locker Service Provider
3. Domain Client and/or Virtual Client
 - a. Implemented by devices that don't support "domain-compliant" DRMs
4. License Issuer(s) for DRMs
 - a. Implemented as part of Rights Locker service
 - b. Must support DRM-specific license generation for each supported DRM

6.3.5 Specification of Ecosystem Parameters

The Founder specifies required Coral parameters for the Ecosystem. The parameters include:

1. Member Client Domain Limit (number of devices allowed in a domain)
2. Low-level role parameters (Rights Token timeouts, etc.)
3. Policies that govern device/domain misuse, such as quickly leaving and rejoining domains
4. Others

6.3.6 Specification of the Ecosystem Trust Model

The Founder (perhaps together with Content Participants) specifies the Ecosystem Trust Model

1. All elements of the Ecosystem that must communicate with one another across Ecosystem trust boundaries must use a Coral supported trusted communication protocols. These require identity and role credentials. In a typical Ecosystem the credentialed entities are:
 - i. Domain Manager Service Providers
 - Require X.509 ID credential and Domain Manager role assertion
 - ii. Rights Locker Service Providers
 - Require X.509 ID credential and Rights Locker role assertion
 - iii. Retailers that communicate with Domain Manager / Rights Locker
 - This communication is typically over retailer-proprietary protocols. Such protocols may have their own required credentials.

6.4 Creation of Ecosystem Legal Agreements

The Founder creates an Ecosystem Adopter Agreement and optionally, a Content Participant Agreement. The terms of these agreements are derived in part from the required sublicensee terms in the *Coral Ecosystem Agreement* and if desired, from the terms in the optional *Model Ecosystem Adopter Terms*. Other terms are Ecosystem-specific, such as Ecosystem-required terms for service level agreements among Ecosystem Service provider participants as well as security and robustness criteria.