www.tv-anytime.org

TV-ANYTIME RMP GROUP WD550

RMP Specification Drafting Process

Specification Workbook

AWAJI, MARCH 2002

This document comprises the drafts of those various sections that will ultimately make the TV-Anytime RMP Specification, document S-5. It is not a draft in the sense that: -

It includes unapproved text, including multiple proposed wordings for the same sections or paragraphs.

It is in a format that is not appropriate for a proper specification document.

Some critical sections are still missing or incomplete.

It is given for information purposes only. Text that is yet to be approved is grayed out.

1. Guiding Principles and Objectives

1. Provide end-to-end persistent protection of content, associated data and user rights
2. Enable interoperation with other systems and the flow of commerce
3. Provide sufficient level of effectiveness to:
    - Prevent widespread, easy hacks, not professional piracy
    - Prevent unfettered re-distribution
    - Protect content related rights and user rights and data through a domain scheme
4. Specify a single mandatory baseline system, allowing interaction with proprietary/legacy RMP systems and optional extensions
5. Specify a complete baseline specification plus optional (specified) features
6. Specify an API between baseline and proprietary RMP systems
7. Specify an API between baseline RMP system and applications
8. Specify a secure communications layer for the RMP Device I/O
9. The APIs and interface above shall support a set of security tools , which may include:
    - Cipher
    - Signature
    - Watermarking
    - Pseudo-Random Number Generator
    - Authentication
    - Hash
    - Other elements

PROPOSED REVISED TEXT, SECTION 1.

This list of principles is in support of TVA-RMP initiative to arrive at a specification for a RMP **baseline** system consistent with the TVARMP commercial requirements and the system reference model.

.

The focus of our work is to **enable and enforce** the business of secure content distribution and consumption by providing robust content protection and copy management against a wide-spread consumer hack. Moreover, the baseline specifies mechanisms for interaction between the proprietary and baseline systems for controlling content as the initial point of content entry to the home is via a proprietary gateway.

To achieve high adoption of this specification, it is recognized, that the system must support a heterogeneous mix of devices and benefit the various stakeholders (e.g., content creators, content distributors, chip and CE device manufacturers, and consumers).

Note, robust implementation guidelines and their legal enforcement are assumed outside the scope of this document

Enabling the business of content distribution and consumption. This has two components: Sustain present business models – allow for end2end and persistent protection of proprietary content and baseline content

Enforce copy and move protection within authorized domain context

Threat model. The baseline RMP system is to protect against easy to use, inexpensive, widespread software and/or hardware hack that **results** in one or more of the following:

a.) Circumvention of use rights – violation of authorized copy permissions and consumption rules of the content.

b.) Illegal redistribution of content by (e.g., copying, retransmission)
c.) Illegal distribution of content keys.

Content Interoperability. To enhance the value of the CE devices to the consumer while maintaining their low cost, it is crucial to provide an efficient baseline RMP system that allows content to flow amongst a heterogeneous mix of compliant devices and content protection systems. This necessitates some mandatory features whose reference architecture allows it to potentially interact with proprietary RMP systems. For example, for content interoperability and a baseline cipher should be chosen for mandatory implementation. Additional approved ciphers may be optionally implemented, e.g., for descrambling the broadcast content.

Full Specification vs. freedom to implement optional features. To provide interoperability for devices with the baseline RMP system optional features should be **specified** so interaction between devices by different manufacturers with such optional features is possible. For example, if a smart card is an optional feature, its interface must be specified by the RMP system.

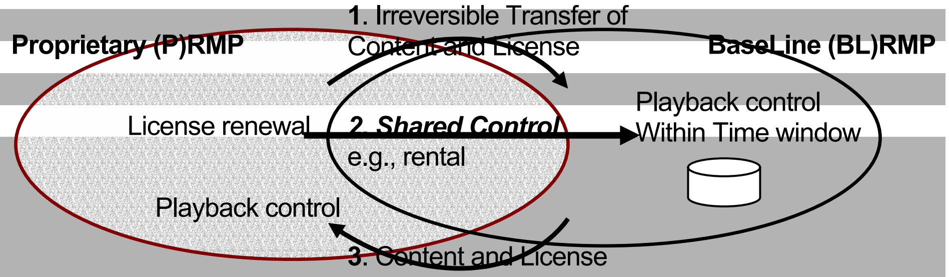
Content interoperability and the Interaction between baseline and proprietary RMP systems. Protected content arrives at home controlled by a proprietary RMP system. It is thus highly desirable to allow consumers to consume/copy such proprietary content using their baseline-RMP-only devices. To do so the proprietary RMP systems may provide time and otherwise-limited license for their content consumption on the baseline system. The baseline system need potentially delegate some security decisions related to authorized use of such content to the proprietary RMP system. This is achieved by specifying an **API** for communication between the RMP baseline RMP system and the proprietary ones, ***RMP-API (or P← →BL).*** The connective tissue between proprietary and RMP BL (BL) systems on same or different devices, will enable content interoperability along the following content control paradigms (1 through 3 are depicted below):

1. Control of content is irreversibly transferred from Proprietary to RMP BL.

2. Shared control of content between Proprietary and RMP BL,

3. Content **and** its control (license) is transferred back from the BL to the Proprietary RMP – *this is not a RMP security enforcement issue*.

4. Control of content is transferred from one Proprietary RMP to another Proprietary RMP by the means of the baseline RMP facilitating the inter proprietary systems pipe.



End to End protection. Whenever possible trust should be only given to necessary nodes: content source device and final rendering ***device*** e.g., STB and TV, respectively. The intermediate links or devices should not be given the content keys as it may only compromise security. Going beyond link encryption is an important tenet of security. The baseline system should facilitate, for example the passing of keys and control data between the source and destination transparently through the intermediary devices that are pass-through devices.

Consumption paradigm. Content X on Device(s) Y(s) for time-window Z. Device may be STB, Digital TV, DVCR, Portable DVCR. Time-window is Always, Now, Rental window

.

Security box. The baseline RMP system architecture should include a toolbox of crypto, security, and content license control functions that may be used by different applications/interfaces in the device. It should include, e.g.,: cipher, signature (data integrity) functions, watermarking function, pseudo random number generator, authentication and hashing algorithms, etc.

Secure Authenticated channels. A basic mechanism for passing control data (e.g., keys, rights, license) should be done via a secure channel using encryption and authenticated key exchange. The devices' authenticated key exchange can be done via locally stored certificates relying on protected device secrets, and/or relying on a trusted third party.

Persistent control. The API and the baseline system shall enable persistent control of the proprietary content in devices that implement only the baseline system. In such cases, the baseline system shall delegate control of such content to its proprietary protection system. There should be no mandatory (forced) handoff of content control from the proprietary to the baseline.

Trusted Community. For any RMP system to work trust mechanisms and policies among the entities - modules, devices, and applications - must be specified, implemented, and enforced. For example, secure authenticated channels must protect all interfaces. Moreover, for a level of trust to be established, the security-relevant features implemented in the device should be certified.

End-2-end Analysis. A protection system is as good as its weakest link There is a clear need to assess the proposals in the context of a full system analysis given it is under attack.

Security in hardware – Hardware-protected entity (e.g. secure video processor). A device is divided into a protected area and its outside; the outside is assumed insecure. To establish trust certain security axioms must be adhered to. For example, security/crypto functions must be done inside a silicon chip since software protection is inherently weak(reflecting present commercial reality). This is to enable trust for content interoperability. (Note, robust implementation guidelines and their enforcement are assumed outside the scope of this document.). This secure entity is protected by licensable IP in accordance with TVA IP licensing statement.

Content license supported by the baseline RMP The content license shall provide the information for copy control and validity reserved fields for (external) system.

2. Architecture

Figure 1 is a logical diagram illustrating the high level elements of the TVA RMP system, including its relationships to enhanced and proprietary RMP components. It is derived from the conceptual diagram, Figure 7.1 in the R-5 Requirements document.

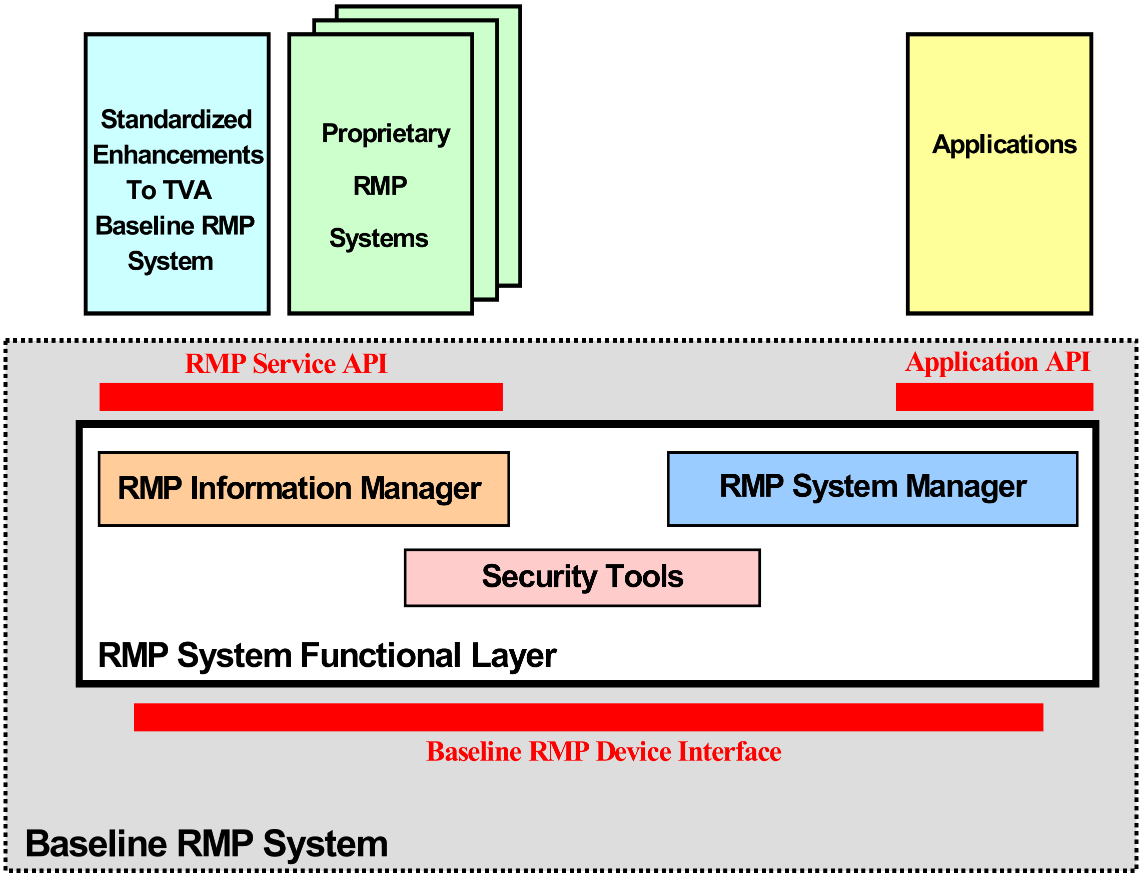
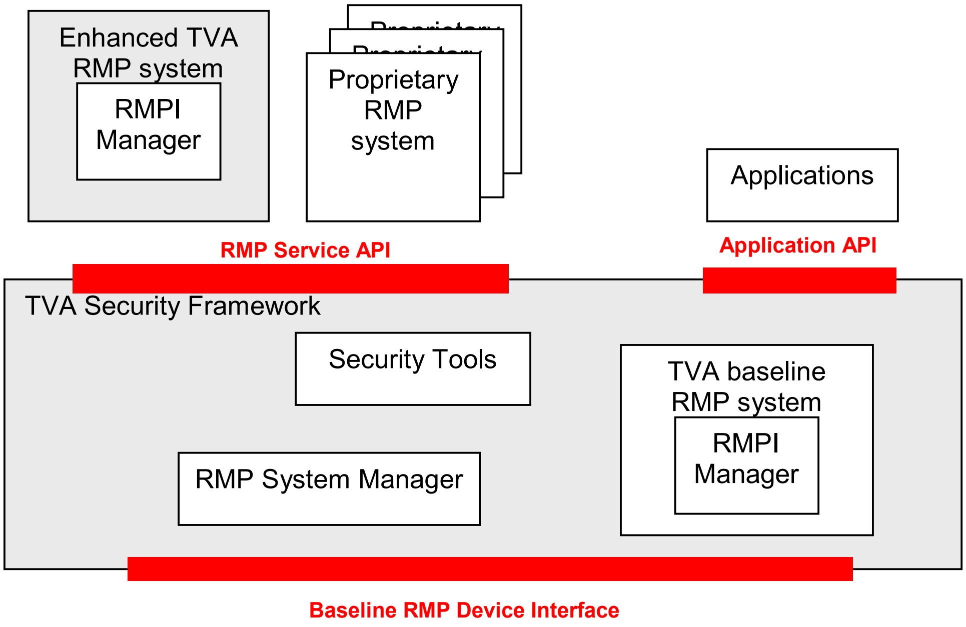


Figure 1: Logical Diagram

Figure 1 Legend

Application API: Allows applications to communicate in an interoperable way with the RMP system.
Application: Software and/or services enabling user access to content and PDR features in accordance with RMP conditions
Baseline RMP System: The functionality conformant with TV Anytime RMP baseline specification.
Proprietary RMP systems: Proprietary content protection systems interfacing with the TVA RMP baseline system through the RMP service API.
RMP Information Manager: Decides what kind of actions are allowed on content, e.g. play, copy, move, etc. and may pass cryptographic keys to security tools.
RMP Service API: Allows an RMP system to communicate in an interoperable way with the RMP Baseline security functions.
RMP System Functional Layer: Collection of functions implementing the Baseline System.
RMP System Manager: Manages operation of the Baseline System.
Security tools: Possibly contains: de-scrambler, watermark detector / embedder, signature verifier, etc.
Standardized enhancements to TVA baseline RMP system: optional extensions to the TVA RMP baseline system
TVAF RMP Baseline Device Interface: A secure communications layer between TVA compliant devices.

Figure 2 shows the basic architecture of the TVA Security Framework.



- Figure 1, TVA security framework architecture

In this figure the following system elements are identified:

Application API: Allows applications to communicate in an interoperable way with the RMP system.

Application: Software and/or services enabling user access to content and features in accordance with RMP conditions.

TVA baseline RMP System: The baseline RMP systems that is present in all devices implementing the TVA Security Framework and that defines the basis for device interoperability.

Security tools: Tools and algorithms used by the different RMP systems: examples are (de)scrambler, watermark detector / embedder, signature verifier, etc.

RMP System Manager: Manages operation of the TVA Security framework.

RMP Service API: Allows an RMP system to communicate in an interoperable way with the RMP Baseline security functions.

Proprietary RMP systems: Proprietary content protection systems interfacing with the TVA RMP baseline system through the RMP service API.

Enhanced TVA RMP system: optional extension to the TVA Security Framework to allow the processing and enforcement of more advanced functionality.

RMPI Manager: This component is the part of the RMP systems that interprets and processes the RMPI of a RMP system.

RMP Service API: Allows an RMP system to communicate in an interoperable way with the TVA Security Framework security functions.

TVA security framework: This framework controls the transfer to and the access of information protected by the TVA RMP system.

TVAF RMP Baseline Device Interface: A secure communications layer that allows communication between TVA security framework compliant devices to interoperate.

The architecture identifies three different RMP systems: proprietary RMP systems, the enhanced TVA RMP system and the TVA baseline RMP system.

The TVA baseline RMP system provides basic content protection functionality and supports a minimum number of business models. It will provide a basic, small and simple system, which can be easy to incorporate into CE devices. The TVA baseline RMP systems are intended to provide basic functionality for exchanging content between devices. As such the system will then provide the basis for inter device interoperability.

The RMPI defined for the TVA baseline RMP system will cover a limited set of usage states (view_only, copy_never, copy_one_generation). A more flexible and elaborate set of RMPI will allow expressing more complex business models. Such RMPI could allow content providers to explore more complex business models without the need for proprietary elements within the end devices. This more advanced RMPI will be used by the so-called Enhanced TVA RMP systems.

The third RMP system type concerns the proprietary RMP systems. Within the scope of the TVA security framework, all non-TVA RMP systems are considered as proprietary RMP systems. The TVA security framework will interface with such systems using the RMP Service API. This will allow that content can move from and to proprietary RMP systems. Examples of proprietary RMP systems are the conditional access systems used for digital broadcasts, disc based RMP systems like the system used for DVD, and analogue protection systems (e.g. Macrovision).

Not all devices implementing the TVA Security framework will support all three RMP system types. As the TVA baseline RMP system is intended to provide basic device interoperability it shall be present in all implementations of the TVA Security Framework. On some devices, the TVA security framework will also support one or more proprietary RMP systems and/or the enhanced TVA RMP system. Communication between these RMP systems and the TVA baseline RMP system will be handled by the RMP Service API.

3. Business Functions

The following are examples of participants in the TVA RMP process and their activities.

Content provider Generates content, generates metadata, gets content identifier, sets usage rules and licensing terms, add promotional material

Service provider Aggregates content, aggregates content information, packages and delivers content, applies content provider rules and adds own rules, processes transactions, captures and aggregates usage information, applies persistent protection, advertises content

Consumer Gets certified equipment, activates service, searches for services, consumes content according to usage rules (watching, viewing, copying, passing on to a friend), sets preferences, generates usage information, initiates transactions

Manufacturer Makes certified equipment, replace broken software

Maintenance issues Service revocation, certificate revocation, content revocation, rights revocation, security renewal

PROPOSED REVISED TEXT, SECTION 3.

The following are examples of participants in the TVA RMP process and their activities.

Content creator Generates content

Rights owner Owns rights to content, sets usage rules and licensing term.

Content provider	Gets content identifier, sets usage rules and licensing terms
Advertisers	Generates ads to be bound to content at various points in the chain.
Distributors	Provides channel to deliver content to Consumers
Service provider	Aggregates content, aggregates content information, packages and delivers content, applies content provider rules and adds own rules, processes transactions, captures and aggregates usage information, applies persistent protection, advertises content
Portal owners	tbd
Network operators	tbd
Equip manufacturer	Makes certified equipment, replace broken software
Equip retailer	Sells certified equipment to consumers
Consumer	Gets certified equipment, activates service, searches for services, consumes content according to usage rules (watching, viewing, copying, passing on to a friend), sets preferences, generates usage information, initiates transactions
Maintenance issues	Service revocation, certificate revocation, content revocation, rights revocation, security renewal
Software developers	Generates value added capabilities to devices after devices have been deployed
Metadata providers	Creates metadata to be bound with content

How the RMP Baseline relates to TVA Phases

Ed: Rework (it was perfect but they want to mess it up)

The TVA RMP maps to TVA Phases as follows:

TVA Baseline RMP: TVA Phase 1 RMP

TVA 2.0 RMP: TVA Phase 2 RMP

TVA x.y RMP: TVA Phase X RMP

The TVA Baseline RMP system is the minimum RMP system that each TVA component must implement in order to become TVA Phase 1 compliant. A TVA RMP device will be able to deliver content to, and receive content from devices based on earlier and later versions of the TVA RMP.

Newer versions of the TVA RMP will support increasingly diverse business models. Each version of the TVA RMP will support the business models of the previous versions of the TVA RMP.

TVA RMP devices will not deliver content to devices that implement a previous version of the TVA RMP specification, if the receiving device is incapable of supporting the business model associated with the content.

Business Models Supported by TVA Baseline RMP

Purchase Restrictions:

- Bound to device
- Bound to personal network of devices
- Rent to own
- Time release (cheaper the older it is)
- Only for a limited duration of time (eg. Pay per day)
- Subscription
    - Season passes
    - Series

- Broadcast with a base set of rights (Free)
    - Additional rights can be acquired

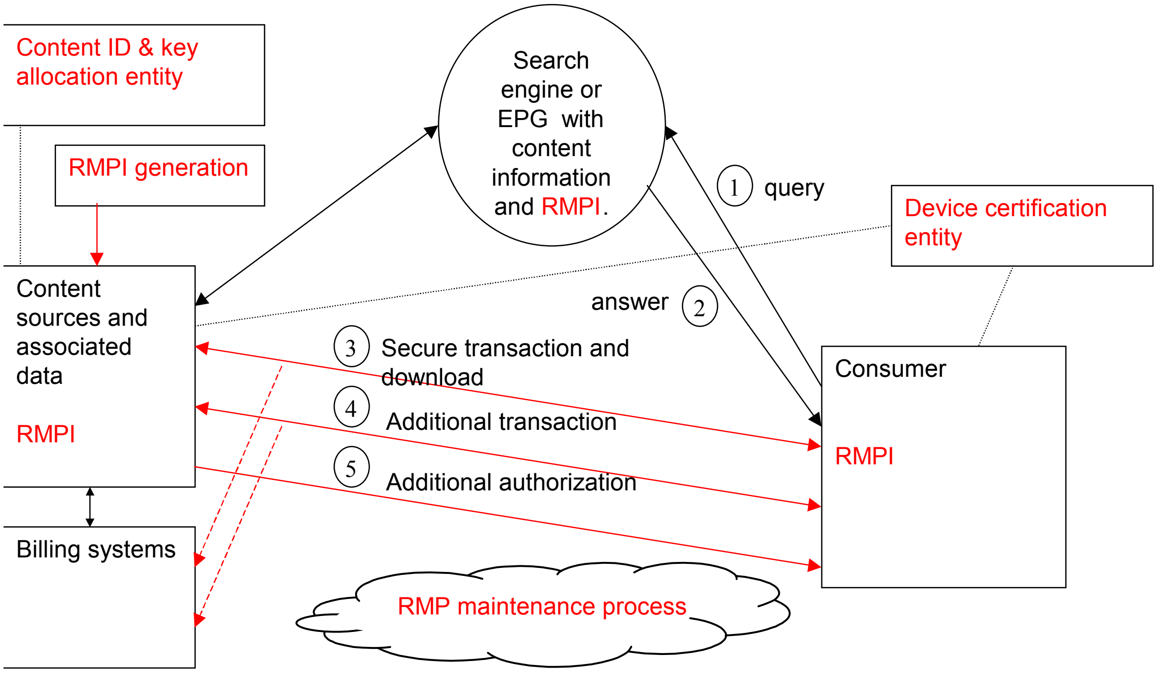
What can be purchased:

- Unlimited play
- Purchase right to make single generation removable copies
- Number of plays
- Access to Segments
    - Commercial Skip
    - Segment Replacement (Targeted Advertising)
    - Highlights
- Trick Play
- Limited Sharing
- Ability to transfer ownership

Regulated distribution

- Territorial restrictions (local vs state vs country vs continent)
- Controlled domains, personal communities (interest and groups)
- Generations of copies
- Number of copies

4. System Process


Content ID & key allocation entity
RMPI generation
Search engine or EPG with content information and RMPI.
1 query
Device certification entity
answer 2
Content sources and associated data
RMPI
3 Secure transaction and download
4 Additional transaction
5 Additional authorization
Consumer
RMPI
Billing systems
RMP maintenance process

• Figure 2: TVA RMPI Business Sequence

The sequence illustrated above exclusively aims to explain the RMP aspects of TV Anytime usage. Content referencing and metadata technologies are presumed to be available to the system to make it work, but it is not explained how this is done. The elements colored red in the above diagram are those within the scope of the RMP system.

To enable his TV Anytime business model, a service provider has made an EPG, pointing to content that should be appealing to consumers. The content has rights associated to it (RMPI), e.g. the content can be viewed a specified number of times, it can be acquired for unlimited use, it can be distributed to others etc. Each right has its own price for the consumer.

The sequence starts with a consumer who activates the EPG (1,2), selects desired content and requests to acquire it for viewing at his/her leisure, i.e. to download it into a PDR. A secure session (3) is set up for the transactions between consumer and service provider. The session can be electronic via an interaction channel, or it can be handled via mail and telephone as a subscription, concluded e.g. by sending a customized Smart Card with access rights to the consumer.

An appropriate entity has certified the consumer's PDR to be TV Anytime RMP compliant. The service provider has ways to exclude equipment that is not compliant, either by direct checking via an electronic interaction channel or by having ensured that content will only play on equipment with valid certificates.

On successful conclusion of the session, the service provider transfers the content with the associated authorizations to the consumer. Although they are bound to each other by some mechanism, the authorizations and the content need not be transferred at the same time; it may be that the content is part of a scheduled broadcast, in which case the authorization is pre-delivered. Finally, content and associated rights are available from the PDR.

On viewing, the consumer may decide that he wants to have unlimited use of the content at his premises by transferring it to a digital archive. Another possibility is that he wants to send the content to a friend as a present. Of course, this requires the friend to have a compliant PDR.

To realize this, the consumer sets up another session (4) with the content provider or further discharges the Smart Card.

On successful conclusion of this session, the service provider sends additional authorizations to the consumer (5) as a post-delivered license.

To enable end-to-end persistent protection of content and management of the associated rights, several entities are required, as shown in the diagram. The content must be uniquely identifiable by the CRID as a non-removable stamp to enable secure binding of the RMPI to the content. The authority that does this normally holds the basic rights to the content. The rights owner also is responsible for the keys to the content protection system.

Another entity may be needed for maintenance of the RMP system.

PROPOSED ALTERNATIVE VERSION OF SECTION 4

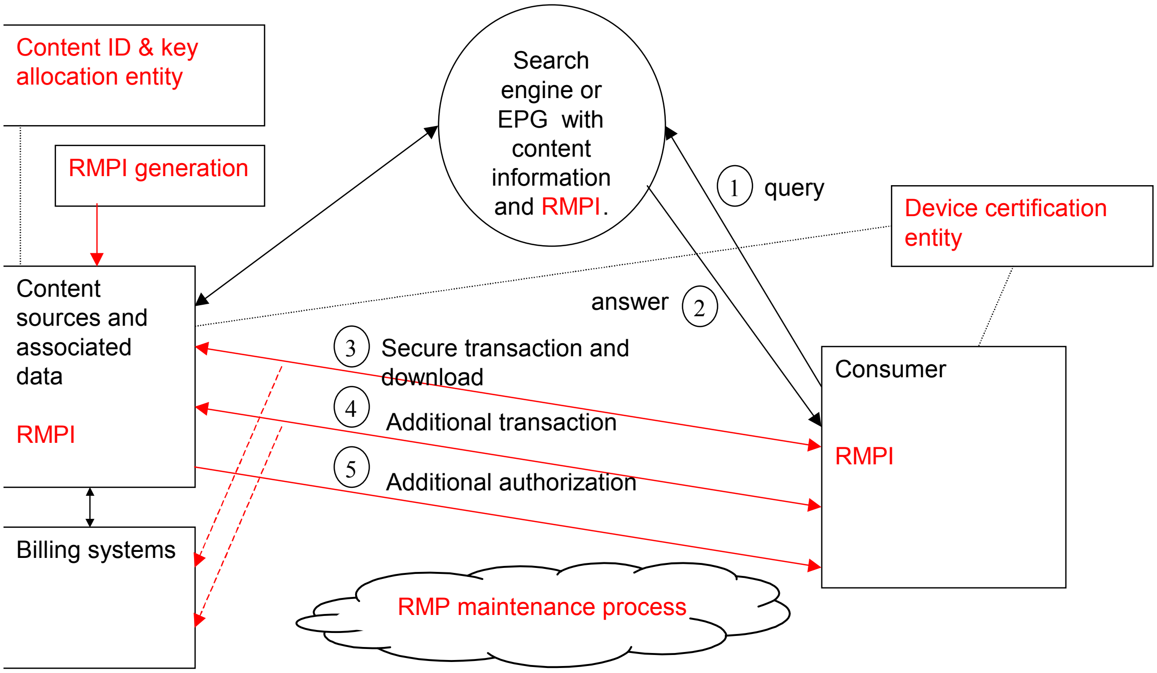

Content ID & key allocation entity
RMPI generation
Search engine or EPG with content information and RMPI.
1 query
Device certification entity
Content sources and associated data
RMPI
answer 2
3 Secure transaction and download
4 Additional transaction
5 Additional authorization
Consumer
RMPI
Billing systems
RMP maintenance process

Figure 2: TVA RMPI Business Sequence

The sequence illustrated above exclusively aims to explain the RMP aspects of TV Anytime usage. Content referencing and metadata technologies are presumed to be available to the system to make it work, but it is not explained how this is done. The elements colored red in the above diagram are those within the scope of the RMP system.

To enable his TV Anytime business model, a service provider has made an EPG, pointing to content that should be appealing to consumers. The content has rights associated to it (RMPI), e.g. the content can be viewed a specified number of times, it can be acquired for unlimited use, it can be distributed to others etc. Each right has its own price for the consumer.

The sequence starts with a consumer who activates the EPG (1,2), selects desired content and requests to acquire it for viewing at his/her leisure, i.e. to download it into a PDR. A secure session (3) is set up for the transactions between consumer and service provider. The session can be electronic via an interaction channel, or it can be handled via mail and telephone as a subscription, concluded e.g. by sending a customized Smart Card with access rights to the consumer.

An appropriate entity has certified the consumer's PDR to be TV Anytime RMP compliant. The service provider has ways to exclude equipment that is not compliant, either by direct checking via an electronic interaction channel or by having ensured that content will only play on equipment with valid certificates.

On successful conclusion of the session, the service provider transfers the content with the associated authorizations to the consumer. Although they are bound to each other by some mechanism, the authorizations and the content need not be transferred at the same time; it may be that the content is part of a scheduled broadcast, in which case the authorization is pre-delivered. Finally, content and associated rights are available from the PDR.

On viewing, the consumer may decide that he wants to have unlimited use of the content at his premises by transferring it to a digital archive. Another possibility is that he wants to send the content to a friend as a present. Of course, this requires the friend to have a compliant PDR.

To realize this, the consumer sets up another session (4) with the content provider or further discharges the Smart Card.

On successful conclusion of this session, the service provider sends additional authorizations to the consumer (5) as a post-delivered license.

To enable end-to-end persistent protection of content and management of the associated rights, several entities are required, as shown in the diagram. The content must be uniquely identifiable by the CRID as a non-removable stamp to enable secure binding of the RMPI to the content. The authority that does this normally holds the basic rights to the content. The rights owner also is responsible for the keys to the content protection system.

Another entity may be needed for maintenance of the RMP system.

5. RMPI

The RMPI is segmented into the following four logical layers.

5.1.1. Semantic layer

This section is to deal with CP rules, usage rules, agents, actions etc. (A first level of copy-protection use cases is to be defined by the RMPI Semantics Ad-Hoc Group.)

5.1.2. Syntactic layer

This section is to address rights expression language and dictionary, namespaces, message protocols etc. It is anticipated that XML will be used. (AN331, AN332, AN338 and AN351 will be adapted to TV-Anytime requirements, as recommended by the RMPI Syntactic Ad-Hoc Group.) This syntactic layer should be compatible with that used by the portions of TV-Anytime Metadata requiring protection.

5.1.3. Serializations

This section is to address the binary representation of the RMPI syntax. This serialization approach should be compatible with that used by TV-Anytime Metadata.

5.1.4. Transport bindings

This section is to address transport requirements for encoding / transmitting serializations.

5.2. RMPI and Metadata relationship

The RMPI employed to enforce the usage rules may take a different form from the "informative" RMPI messaging to user interfaces. Wherever applicable, RMPI data and TVA Metadata should use the same language and structure.

5.3. RMPI Protection

This section is to address protection of RMPI.

REVISED PROPOSED TEXT

The goal of this section is to describe the meaning and use of rights management and protection information in a TV Anytime context using models and terminology (semantics) sufficient for human understanding, and to refine those descriptions into explicit syntax using XML that was precise enough to construct conformance tests, and could be machine parsed with sufficient precision to control TV Anytime devices according to the requirements specified in documents WD460 (working draft S-5 specification) and TV039r7 (RMP R-5 Requirements), and RMPI contribution documents.

The scope of this work is primarily to provide a mechanism for describing the use of video content. It is not the intent of this document to decide what uses are appropriate. It is assumed that representatives of content rights holders, broadcasters, equipment manufacturers, governments, and consumers will determine what uses of content are appropriate, and RMPI will provide the means to define and express those uses. The RMP system, in cooperation with RMP information and the RMP information management system described here, will control those uses in accordance with the RMPI provided.

The RMPI is segmented into the following four logical layers.

5.4. Semantic layer

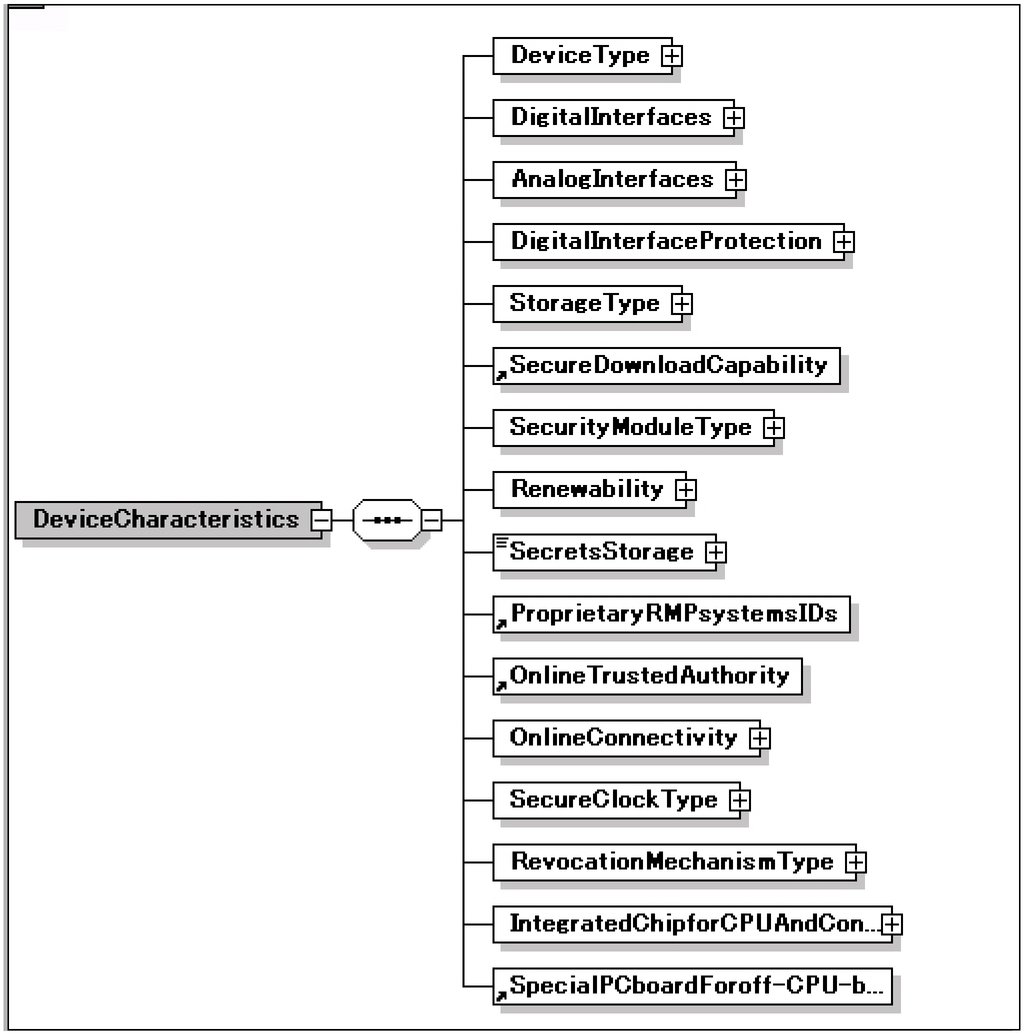
This section is to deal with CP rules, usage rules, agents, actions etc. (A first level of copy-protection use cases is to be defined by the RMPI Semantics Ad-Hoc Group.)

5.4.1. Device Characterization Mask

(Ed: need to be discussed in more detail whether this information is to be included in the RMPI vocabulary.)

Information concerning device characterization mask is relevant to RMPI and the possible terms to be incorporated in the RMPI semantic layer are listed herein.

The certified mask characterizes the type and cardinal features of the device that are relevant and helpful in assessing trust level. It is part of the device certificate. The purpose is to enable the content source protection system to make an informed decision about its interaction with a destination device by assessing the destination device mask. If for example, we wish not to allow some content to be sent to and rendered on a device with removable storage, there is a need to know the device storage capability. The following diagram shows the overall structure of the possible elements to be considered. The details of each of these elements are given in the attached XML file *“NDS-devices.xsd”*.


DeviceCharacteristics
DeviceType
DigitalInterfaces
AnalogInterfaces
DigitalInterfaceProtection
StorageType
SecureDownloadCapability
SecurityModuleType
Renewability
SecretsStorage
ProprietaryRMPsystemsIDs
OnlineTrustedAuthority
OnlineConnectivity
SecureClockType
RevocationMechanismType
IntegratedChipforCPUAndCon...
SpecialPCboardForoff-CPU-b...

The descriptions of the terms in this category are given below:

Name	*Description*
`DeviceType`	(Details to be filled in). Examples are PC, STB, PVR, DTV, cell phone, PDA, ,
`DigitalInterfaces`	Digital Interfaces of the device for example USB, ethernet, 1394, 1394+5C, DVI, BlueTooth, etc, wireless802.11a/b/e, Modem, other. (Details to be filled in).
`AnalogInterfaces`	*See the attached XML file*
`DigitalInterfaceProtection`	*See the attached XML file*
`StorageType`	*See the attached XML file*
`SecureDownloadCapability`	*See the attached XML file*
`SecurityModuleType`	*See the attached XML file*
`Renewability`	*See the attached XML file*
`SecretsStorage`	*See the attached XML file*
`ProprietaryRMPsystemsIDs`	This is the initial – out of factory or pre-certificate installation - set of RMP systems on the device. Additional RMPs sytems may be downloaded later to the device but the old certificate is not changed but may be extended.
`OnlineTrustedAuthority`	*See the attached XML file*
`OnlineConnectivity`	*See the attached XML file*
`SecureClockType`	*See the attached XML file*
`RevocationMechanismType`	*See the attached XML file*
`IntegratedChipforCPUAndContentProcessing`	*See the attached XML file*
`SpecialPCboardForoff-CPU-busvideohandling`	*See the attached XML file*

5.4.2. Terms for various entitlement management information

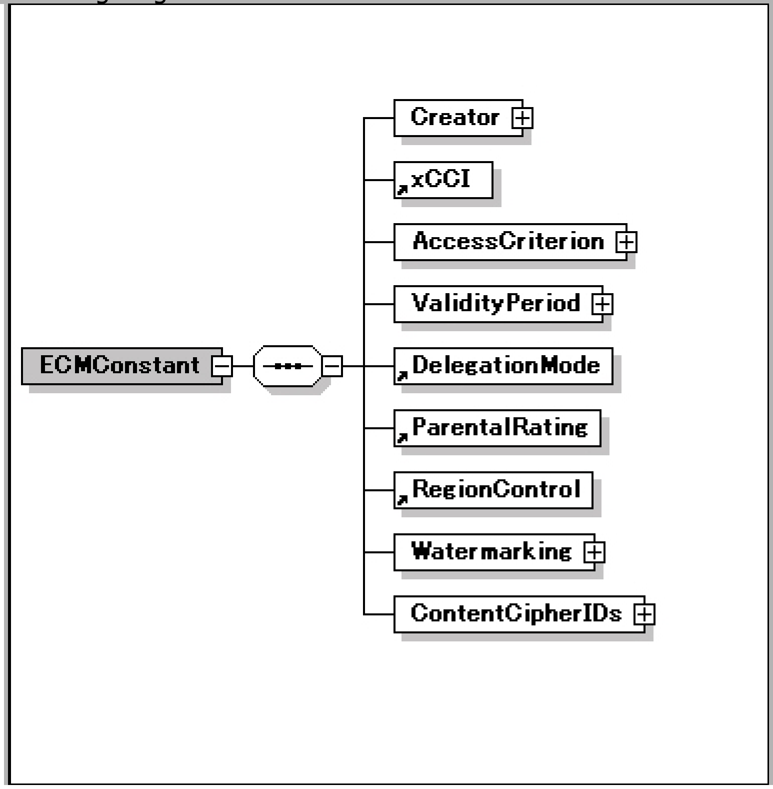
5.4.2.1. BL-ECM (license, voucher)

It is noted that broadcast ECM is often not suited for use over a long period of time. It also represents a global content-decryption-key. It is necessary to create a local ECM that can provide the content key that exists for a long period of time within the user network. This local ECM – BL-ECM - is to be used within the specified authorized domain to control content usage states and house content decryption key in a protected manner. The BL-ECM is comprised of two parts:

2.2 **Constant part** – same for the the whole content, (e.g., FECM, or metadata), and

2.2 **Changing part** – encrypted content CW and other info - varies with crypto-period

The following diagram illustrates the basic elements contained in the Constant ECM.



The descriptions of each element is given below (See the attached XML file *NDS-ecm.xsd* for details.)

Name	*Description*
`Creator`	*Details to be filled in later*. Examples are P or BL system ID; Device ID/URL; Domain-ID;*See the attached XML file*
`xCCI`	*See the attached XML file*
`AccessCriterion`	*See the attached XML file*
`ValidityPeriod`	*See the attached XML file*
`DelegationMode`	*See the attached XML file*
`ParentalRating`	*See the attached XML file*
`RegionControl`	*See the attached XML file*
`Watermarking`	*See the attached XML file*
`verificationSystemID`	*See the attached XML file*
`FingerprintingSystemID`	*See the attached XML file*
`ContentCipherIDs`	*See the attached XML file*
`ControlcipherID`	*See the attached XML file*
`EncryptionMode`	*See the attached XML file*
`Signature`	*See the attached XML file*

5.4.2.2. ExCCI

ExCCI terms are taken from the preliminary Technical Committee Working Document, which is only an input to SMPTE's standardization process. It may or may not be incorporated in whole or in part in a resulting SMPTE standard, but it is being considered by SMPTE, and is recommended for TV Anytime's consideration.

The following table gives the description of ExCCI terms.

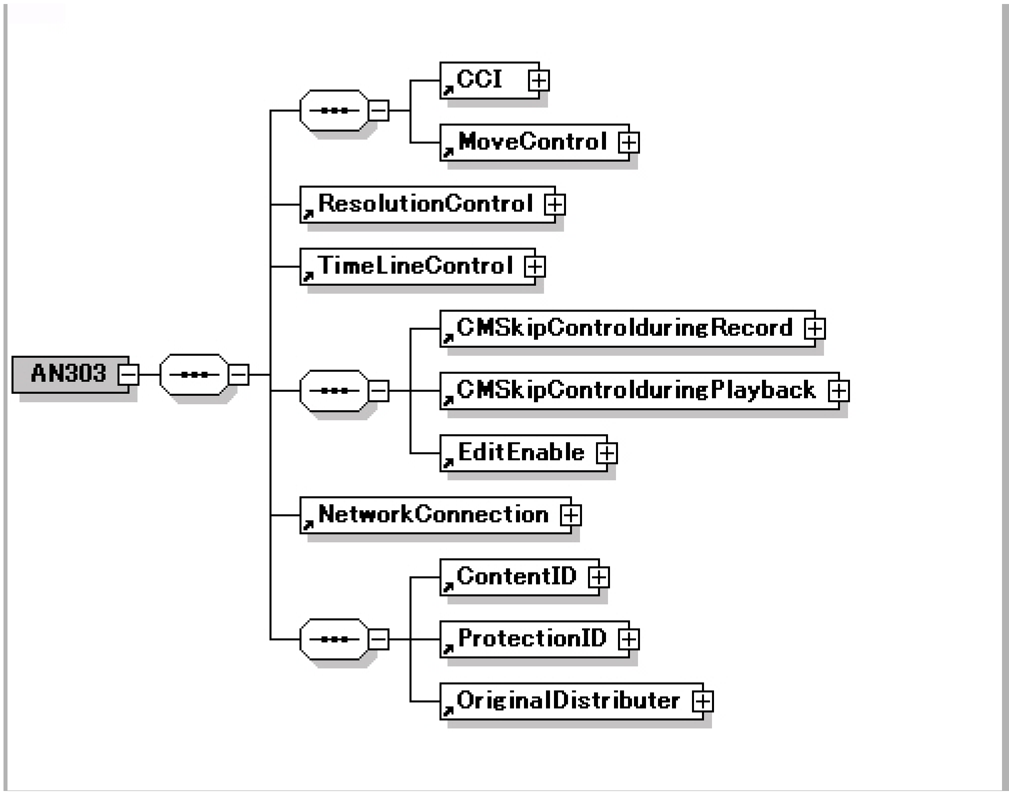
Name	*Description*
`ExCCIVersion`	a binary number representing the Version of the ExCCI data packet table.

ContentControl		encompasses the copy control information bits associated with the Copy Generation Management System, Analog Protection System, etc.
AnalogComponentOutput		conveys whether the content can be output using the device's Analog Component Output
		Analog component output not allowed
		Analog component output allowed.
DigitalComponentOutputs		convey whether the content can be output using the device's digital outputs
		No digital signal output allowed.
		Only uncompressed digital signal output allowed.
		Only compressed digital signal output allowed.
		Both compressed and uncompressed digital signal outputs allowed.
CopyGenerationManagementSystemForAnalog		
		Copy control not asserted, unauthorized retransmission outside the home not permitted.
		No further copying is permitted.
		One generation copy is permitted.
		Copying is not permitted.
AnalogProtectionSystem		convey the type of analog copy protection system to be generated by the device, but only if both CGMS-A bits are equal to 1. In the case where either of the CGMS-A bits are equal to 1, the APS will be turned off. Analog copy protection can utilize Pseudo Sync Pulse (PSP) and/or an inverted split color burst signal generation.
		Analog copy protection is not used.
		PSP is on & split color burst is off.
		PSP is on & 2-line split color burst is on.
		PSP is on & 4-line split color burst is on.
ImageConstrain		conveys information about whether High Definition resolution content must be image-constrained to no greater than 600 vertical pixels (standard definition resolution) when being transmitted over unprotected analog component outputs.
		High Definition resolution content need not be image constrained when being transmitted over unprotected analog component outputs.
		High Definition resolution content must be image constrained to no greater than 600 vertical pixels when being transmitted over unprotected analog component outputs.
Retransmission		allows content owners to mark their content to restrict its retransmission beyond the Home.
		The content may not be retransmitted beyond the Home.
		The content may be retransmitted beyond the Home

CopyControlAssertion		allows content owners to mark their content using existing 2-bit CCI data to trigger content encryption. But once the content is decrypted and the content's ExCCI read, the 'CCA' bit conveys information concerning whether copy control is being asserted by the content owner.
		Copy control is not asserted, regardless of CGMS data.
		Copy control is asserted.
CopyGenerationManagementSystemForDigital		the equivalent of the CGMS-A bits and correspond to the definitions defined above for the CGMS-A bits. It should be noted that the CGMS-D bits can differ from the CGMS-A bits in order to allow different copying restrictions based on what type of signal (whether analog or digital) is being fed to the recording device.
CopyMoveCount		define an integer value 'M' that represents the number of times a copy made of Copy Once content can be moved to another recording medium with the original recording being erased and/or no longer accessible.
		A copy made of Copy Once content can not (or can no longer) be moved.
		Number of times a copy of Copy Once content can be moved.
		A copy made of Copy Once content can be moved an unlimited number of times.

5.4.2.3. Fuji et al.

Nippon Television Network Inc., Tokyo Broadcasting System, and Fuji Television Network System have contributed a list of possible RMPI terms. The following diagram illustrates the range of categories included in the list.



The descriptions of the terms included in each of the above categories are given below. (See the attached XML file *AN303.xsd*).

Name	*Description*
`CCI`	Control number of personal copies *See the attached XML file*
`MoveControl`	Control number of personal copies *See the attached XML file*
`ResolutionControl`	Control the resolution of playbacked material, irrelevant to the number of copies or *e the attached XML file*
`TimeLineControl`	Control playback start date of recorded protected material. *See the attached XML file*
`CMSkipControlduring`	Control Automatic skip of CM material during playback. *See the attached XML file*
`CMSkipControlduring`	Control Automatic skip of CM material during recording. *See the attached XML file*
`EditEnable`	Control segmentation of contents and editing. *See the attached XML file*
`NetworkConnection`	Control content transmission beyond Home Network. *See the attached XML file*
`ProtectionTypeID`	Identify encryption or protection technology applied to the content, for contents to be l beyond the Home Network or distributed through removable media e.g. 5C-DTCP, DVB-RMP. *See the attached XML file*
`ContentID`	Unique ID to define content, e.g. ISAN, cIDf.

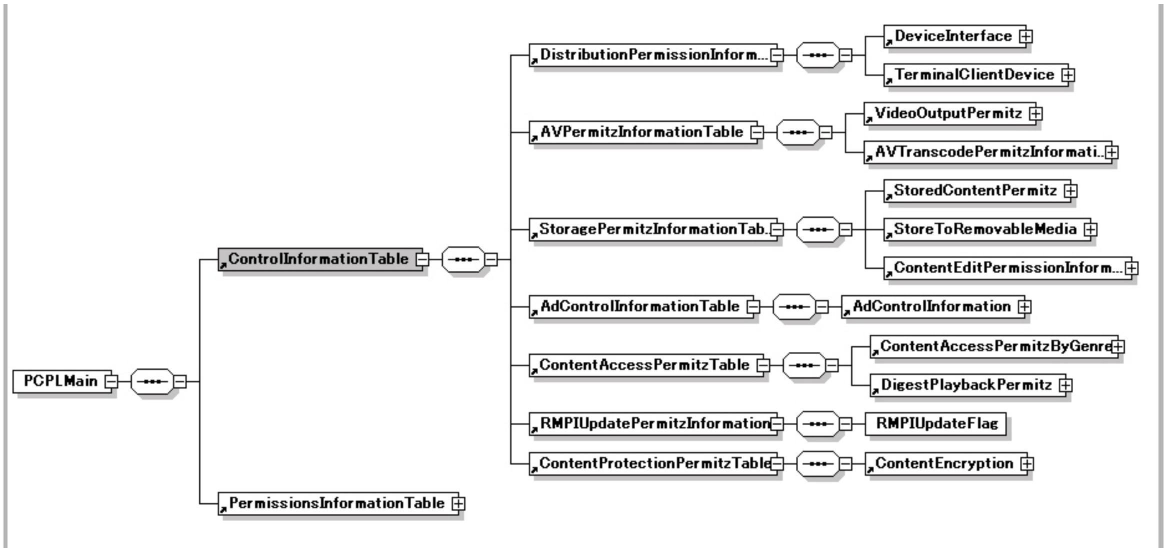
Name	*Description*
`OriginalDistributer`	Unique ID for entity who brought the contents originally onto the market. Whereas D provides ID for contents holder, This ID will provide information of the Content Distributor

The following is an additional terms proposed by ARIB (AN366). See the attached XML file AN366.xsd.

Name	*Description*
`RetentionMode`	This information indicates that the received content can be stored , even if the copy control information indicates copy never.
`RetentionState`	This information is applied with the RetentionMode, and indicates that me of temporarily stored content.
`EncryptionMode`	This information indicates the necessity of the protection when outputting ontent to the high-speed digital output interfaces.

5.4.2.4. Permissions Language

See the attached XML file permis*sion_language.xsd.*


PCPLMain
ControlInformationTable
PermissionsInformationTable
DistributionPermissionInform...
DeviceInterface
TerminalClientDevice
AVPermitzInformationTable
VideoOutputPermitz
AVTranscodePermitzInformati...
StoragePermitzInformationTab...
StoredContentPermitz
StoreToRemovableMedia
ContentEditPermissionInform...
AdControlInformationTable
AdControlInformation
ContentAccessPermitzTable
ContentAccessPermitzByGenre
DigestPlaybackPermitz
RMPIUpdatePermitzInformation
RMPIUpdateFlag
ContentProtectionPermitzTable
ContentEncryption

General Attributes

Name	*Definition*
`version`	Indicates the version of the specification used to describe the permissions information.
`ResourceID`	Indicates the IDs associated with content to be granted permissions.
`ResourceAttrs`	Indicates the attributes of the content
`ServerSideAttrs`	Indicates the attributes that the “server” side has.
`DistributionAttrs`	Indicates the attributes inherent in the distribution environment

Name	*Definition*
ClientSideAttrs	Indicates the attributes relevant to client
ServiceAttrsType	Indicates the attributes relevant to both server and client
ContentType	Indicates the type of content.

Audio-Visual Attributes

Name	*Definition*
AudioCompression	Indicates the conditions concerning the method(s) that can be or must be used to compress the audio part of the content (if any).
VideoCompression	Indicates the conditions concerning the method(s) that can be or must be used to compress the video part of the content (if any).
AudioSamplingRate	Indicates the conditions concerning the applicable sampling rate of the audio part of the content (if any).
AudioBitRate	Indicates the conditions concerning the applicable bit rates of the audio part of the content (if any).
VideoBitRate	Indicates the conditions concerning the applicable bit rates of the video part of the content (if any).
ScreenSize	Indicates the conditions concerning the screen size when the content is displayed.
PrimaryRecordingMedia	Indicates the conditions concerning the media on which the content can be recorded for the first time when it is received.
SecondaryRecordingMedia	Indicates the conditions concerning the media on which copies of the content (if copying is allowed) can be recorded after it has been received

Distribution Attributes

Name	*Definition*
ExtendedLanguageTypeSpokenPurpose	Indicates the purpose of use (usage type) for the content in question, such as individual uses, profit-pursuing business uses, non-profit-making business uses, educational uses, and broadcasting uses.

Name	*Definition*
`SalesMethod`	Indicates the sales method permitted to the content.
`TransmissionMode`	Indicates the type of communication/broadcast network that can be used for delivery of the content.
`Territory`	Indicates the place where the content can be sold or bought. The values of this type are currently derived from DVD Regional Management Information Code.
`CompilationUnit`	Indicates the conditions under which the content is combined with other contents and edited.

Usage Control Attributes

Name	*Definition*
`UserLimitation`	When the user is limited with regard to the use of the content, indicates the details in free format. For instance, the "R specification" may be pertinent for video content.
`PlayBackControl`	Indicates the conditions concerning the playback of the content.
`CopyPermission`	Indicates the number or the generations of copies that may be made.
`AlteringPermission`	Indicates the range for which altering of the content is allowed. This may be altering disabled, free, partial extraction, contents combining, size change, color change, speed change, translation, etc.

Protection Attributes

Name	*Definition*
`DRM`	Indicates the DRM (Digital Rights Management) method(s) that can be or must be used with regard to the delivery of the content
`EncryptionFlag`	Indicates the encryption method(s) that can be or must be used with regard to the delivery of the content.
`WaterMarkingFlag`	Indicates the watermarking method(s) that can be or must be used with regard to the delivery of the

Name	Definition
	content.

5.5. Syntactic layer

This section is to address rights expression language and dictionary. The language used in couching the information defined in the previous section is XML. XML is used here for representational purpose. (Ed: Make clear that it is not required that the box is actually implemented to handle XML directly.) This syntactic layer should be compatible with that used by the portions of TV-Anytime Metadata requiring protection, such as consumer metadata, targeting metadata, and segmentation metadata.

In each of these cases, the content owners can regulate the rendering, distribution and use of their content by expressing their intent in an XML License. Sophisticated business models can be created by allowing RMPI in the form of licenses to be added to the system from a variety of sources. Each of the TVA compliant devices will adhere to the intent of the RMPI.

5.5.1. Definition of the Rights Language

This section defines the syntax of the language that will express the Rights Management and Protection Information, as described in the previous sections.

1. Schema

3.1 Rights

```
<xsd:element name="segment" type="tvax:Segment"
substitutionGroup="r:right">
 </xsd:element>
 <xsd:complexType name="Segment">
    <xsd:complexContent>
       <xsd:extension base="r:Right"/>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="trickPlay" type="tvax:TrickPlay"
substitutionGroup="r:right">
 </xsd:element>
 <xsd:complexType name="TrickPlay">
    <xsd:complexContent>
       <xsd:extension base="r:Right"/>
    </xsd:complexContent>
 </xsd:complexType>
```

Name	Description
`segment`	A Derivitave Work Right giving the permission to partition one piece of content into any number of subpieces in such a way that the sum of the subpieces is equal to exactly the whole content, without duplication.
`trickPlay`	right to play with various tricks (ff, pause, rw, etc., as listed in the allowedTricks condition).
`encryptionMethod`	If present, specifies that the indicated encryption method must be used.

3.2 Resource

```
<xsd:element name="subscription" type="tvax:Subscription" substitutionGroup="r:resource">
</xsd:element>
<xsd:complexType name="Subscription">
   <xsd:complexContent>
      <xsd:extension base="r:Resource">
         <xsd:sequence minOccurs="0">
            <xsd:element name="id" type="xsd:anyURI"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Name	Description
`subscription`	A Property Resource indicating the property of being subscribed to a certain subscription.

3.3 Conditions

```
 <xsd:element name="destinationMediaInternal"
type="tvax:DestinationMediaInternal"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="DestinationMediaInternal">
    <xsd:complexContent>
       <xsd:extension base="r:Condition"/>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="recordingMedia" type="tvax:RecordingMedia"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="RecordingMedia">
    <xsd:complexContent>
       <xsd:extension base="r:Condition">
          <xsd:choice minOccurs="0" maxOccurs="unbounded">
             <xsd:element name="cd-r"/>
             <xsd:element name="md"/>
             <xsd:element name="dvd-r"/>
             <xsd:element name="hd"/>
             <xsd:element name="sdCard"/>
             <xsd:element name="magicGate"/>
             <xsd:element name="memoryStick"/>
             <xsd:element name="smartMedia"/>
             <xsd:any namespace="##any"/>
          </xsd:choice>
       </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="distributionMedium"
type="tvax:DistributionMedium" substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="DistributionMedium">
    <xsd:complexContent>
       <xsd:extension base="r:Condition">
          <xsd:choice minOccurs="0" maxOccurs="unbounded">
             <xsd:element name="terrestrial"/>
             <xsd:element name="sattelite"/>
             <xsd:element name="catv"/>
             <xsd:element name="internet"/>
             <xsd:element name="mobile"/>
             <xsd:element name="removable"/>
             <xsd:any namespace="##any"/>
          </xsd:choice>
       </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="dvdRegion" type="tvax:DvdRegion"
substitutionGroup="r:condition">
```

```
 </xsd:element>
 <xsd:complexType name="DvdRegion">
    <xsd:complexContent>
       <xsd:extension base="r:Condition">
          <xsd:sequence minOccurs="0">
             <xsd:element name="region1" minOccurs="0"/>
             <xsd:element name="region2" minOccurs="0"/>
             <xsd:element name="region3" minOccurs="0"/>
             <xsd:element name="region4" minOccurs="0"/>
             <xsd:element name="region5" minOccurs="0"/>
             <xsd:element name="region6" minOccurs="0"/>
          </xsd:sequence>
       </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
```

Name	*Description*
`destinationMediaInternal`	Satisfied when all the destination media (if any) are not removable and has no external interface.
`recordingMedia`	Satisfied if the recording media is one of the specified kinds
`distributionMedium`	Satisfied if all the distribution media in question are one of the specified kinds.
`dvdRegion`	Satisfied if all the regions in which distribution and/or use is conducted is one of the specified regions.

3.3 Quality Conditions

```
<xsd:element name="bitRate" type="tvax:BitRate"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="BitRate">
    <xsd:complexContent>
       <xsd:extension base="r:Condition">
          <xsd:sequence minOccurs="0">
             <xsd:element name="minBitsPerSec"
type="xsd:nonNegativeInteger" minOccurs="0"/>
```

```
           <xsd:element name="maxBitsPerSec"
type="xsd:nonNegativeInteger" minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="colors" type="tvax:Colors"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="Colors">
   <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
           <xsd:element name="minBits"
type="xsd:nonNegativeInteger" minOccurs="0"/>
           <xsd:element name="maxBits"
type="xsd:nonNegativeInteger" minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="compression" type="tvax:Compression"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="Compression">
   <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
           <xsd:element name="none" minOccurs="0"/>
           <xsd:element name="mpeg1" minOccurs="0"/>
           <xsd:element name="mpeg1layer2" minOccurs="0"/>
           <xsd:element name="mpeg1layer3" minOccurs="0"/>
           <xsd:element name="mpeg2" minOccurs="0"/>
           <xsd:element name="mpeg4" minOccurs="0"/>
           <xsd:element name="zip" minOccurs="0"/>
           <xsd:any namespace="##any" minOccurs="0"
maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="frameRate" type="tvax:FrameRate"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="FrameRate">
   <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
```

```
           <xsd:element name="minFramesPerSec"
type="xsd:nonNegativeInteger" minOccurs="0"/>
           <xsd:element name="maxFramesPerSec"
type="xsd:nonNegativeInteger" minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="resolution" type="tvax:Resolution"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="Resolution">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
           <xsd:element name="minPixelWidth" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
                   <xsd:element name="absolute"
type="xsd:nonNegativeInteger"/>
                   <xsd:element name="relativeToSource"
type="xsd:float"/>
                </xsd:choice>
              </xsd:complexType>
           </xsd:element>
           <xsd:element name="minPixelHeight" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
                   <xsd:element name="absolute"
type="xsd:nonNegativeInteger"/>
                   <xsd:element name="relativeToSource"
type="xsd:float"/>
                </xsd:choice>
              </xsd:complexType>
           </xsd:element>
           <xsd:element name="maxPixelWidth" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
                   <xsd:element name="absolute"
type="xsd:nonNegativeInteger"/>
                   <xsd:element name="relativeToSource"
type="xsd:float"/>
                </xsd:choice>
              </xsd:complexType>
           </xsd:element>
           <xsd:element name="maxPixelHeight" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
```

```
                   <xsd:element name="absolute"
type="xsd:nonNegativeInteger"/>
                   <xsd:element name="relativeToSource"
type="xsd:float"/>
                 </xsd:choice>
              </xsd:complexType>
           </xsd:element>
         </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="screenSize" type="tvax:ScreenSize"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="ScreenSize">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
                                         <xsd:element
name="minInchesDiagonalViewable" type="xsd:nonNegativeInteger"
                         minOccurs="0"/>
           <xsd:element name="maxInchesDiagonalViewable"
type="xsd:nonNegativeInteger" minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
```

Name	*Description*
`bitRate`	Satisfied if the bit rate of the destination content or rendered content is within the specified range. That is, when minBitsPerSec is present, the bit rate MUST be greater than or equal to that value, and when maxBitsPerSec is present, the bit rate MUST be less than or equal to that value.
`colors`	Satisfied if the bits/pixel for color of the destination content or rendered content is within the specified range. That is, when minBits is present, the bit/pixel for color MUST be greater than or equal to that value, and when maxBits is present, the bit rate MUST be less than or equal to that value.
`compression`	Satisfied if the final destination content or

Name	Description
	rendering process uses a compression algorithm listed herein.
`frameRate`	Satisfied if the frame rate of the destination content or rendered content is within the specified range. That is, when minFramesPerSec is present, the frame rate MUST be greater than or equal to that value, and when maxFramesPerSec is present, the frame rate MUST be less than or equal to that value.
`resolution`	Satisfied if the resolution of the destination content or rendered content is within the specified range. That is, when minPixelWidth is present, the pixel width of the resolution MUST be greater than or equal to that value, when minPixelHeight is present, the pixel height of the resolution MUST be greater than or equal to that value, when ,maxPixelWidth is present, the pixel width of the resolution MUST be less than or equal to that value, and when maxPixelHeight is present, the pixel height of the resolution MUST be less than or equal to that value.
`screenSize`	Satisfied if all of the screens have a viewable diagonal whose size in inches is within the specified range. That is, when minFramesPerSec is present, the frame rate MUST be greater than or equal to that value, and when maxFramesPerSec is present, the frame rate MUST be less than or equal to that value.

3.4 Quantity Conditions

```
 <xsd:element name="minRenderQuantity"
type="tvax:MinRenderQuantity" substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="MinRenderQuantity">
```

```
    <xsd:complexContent>
      <xsd:extension base="tvax:Quantity"/>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="maxRenderQuantity"
type="tvax:MaxRenderQuantity" substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="MaxRenderQuantity">
    <xsd:complexContent>
      <xsd:extension base="tvax:Quantity"/>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="maxNewMaterialQuantity"
type="tvax:MaxNewMaterialQuantity"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="MaxNewMaterialQuantity">
    <xsd:complexContent>
      <xsd:restriction base="tvax:Quantity">
         <xsd:choice minOccurs="0">
           <xsd:element name="frames"
type="xsd:nonNegativeInteger"/>
           <xsd:element name="seconds"
type="xsd:nonNegativeInteger"/>
         </xsd:choice>
      </xsd:restriction>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="maxRemovedMaterialQuantity"
type="tvax:MaxRemovedMaterialQuantity"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="MaxRemovedMaterialQuantity">
    <xsd:complexContent>
      <xsd:extension base="tvax:Quantity"/>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:complexType name="Quantity">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:choice minOccurs="0">
           <xsd:element name="entirety"/>
           <xsd:element name="frames"
type="xsd:nonNegativeInteger"/>
           <xsd:element name="seconds"
type="xsd:nonNegativeInteger"/>
         </xsd:choice>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
```

Name	Description
minRenderQuantity	Specifies the minimum amount of the digital resource that can be rendered during each exercise of the right.
maxRenderQuantity	Specifies the maximum amount of the digital resource that can be rendered during each exercise of the right.
maxNewMaterialQuantity	Specifies the maximum amount of new content that can be added to the resource during each exercise of the right.
maxRemovedMaterialQuantity	Specifies the maximum amount of the digital resource that can be removed during each exercise of the right.
Quantity	Specifies a quantity as either the entirety of the digital resource or some fraction thereof.

3.5 Other Conditions

```
    <xsd:complexType name="EncryptionRequirement">
<xsd:complexContent>
      <xsd:extension base="r:Condition">
        <xsd:sequence>
          <xsd:element name="encryptionMethod"
type="enc:EncryptionMethodType">
          </xsd:element>
          <xsd:element ref="r:principal"/>
          <xsd:element name="cipherData"
type="enc:CipherDataType">
          </xsd:element>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
```

```
 </xsd:complexType>
 <xsd:element name="encryptionRequirementAtDestination"
type="tvax:EncryptionRequirement"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:element name="encryptionRequirementAtSource"
type="tvax:EncryptionRequirement"
substitutionGroup="r:condition">
 </xsd:element>







<xsd:element name="linkSecurity" type="tvax:LinkSecurity"
substitutionGroup="r:condition">
 </xsd:element>
 <xsd:complexType name="LinkSecurity">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
           <xsd:element name="analog" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
                   <xsd:element name="all"/>
                   <xsd:element name="none"/>
                   <xsd:choice maxOccurs="unbounded">
                      <xsd:element name="macrovision"/>
                      <xsd:any namespace="##any"/>
                   </xsd:choice>
                </xsd:choice>
              </xsd:complexType>
           </xsd:element>
           <xsd:element name="digital" minOccurs="0">
              <xsd:complexType>
                <xsd:choice>
                   <xsd:element name="all"/>
                   <xsd:element name="none"/>
                   <xsd:choice maxOccurs="unbounded">
                      <xsd:element name="cci">
                         <xsd:complexType>
                           <xsd:choice>
                              <xsd:element name="free"/>
                              <xsd:element name="copyOnce"/>
                              <xsd:element name="copyNoMore"/>
                              <xsd:element name="copyNever"/>
                           </xsd:choice>
```

```
                          </xsd:complexType>
                       </xsd:element>
                       <xsd:any namespace="##any"/>
                    </xsd:choice>
                 </xsd:choice>
              </xsd:complexType>
            </xsd:element>
            <xsd:any namespace="##any" minOccurs="0"
maxOccurs="unbounded"/>
         </xsd:sequence>
       </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
```

Name	*Description*
EncryptionRequirement	Specifies an encryption that must be used.
encryptionMethod	If present, specifies that the indicated encryption method must be used.
cipherData	If present, specifies that the actual result of the encryption must be the value indicated here.
encryptionRequirementAtDestination	Satisfied if all the encryptions used for the destination content are the one specified here.
linkSecurity	Satisfied if every link used for busing the content around during the exercise of this right is listed herein.

```
<xsd:element name="allowedTricks" type="tvax:AllowedTricks"
substitutionGroup="r:condition">
</xsd:element>
 <xsd:complexType name="AllowedTricks">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:sequence minOccurs="0">
            <xsd:element name="fastPlayback"/>
            <xsd:element name="fastForward"/>
            <xsd:element name="rewind"/>
            <xsd:element name="pause"/>
            <xsd:element name="jump"/>
            <xsd:element name="skipNext"/>
            <xsd:element name="skipForward"/>
            <xsd:element name="skipBackward"/>
            <xsd:element name="skipToEnd"/>
            <xsd:element name="skipToStart"/>
            <xsd:element name="loopRepeat"/>
            <xsd:element name="slowMotion"/>
            <xsd:element name="stepForward"/>
            <xsd:element name="stepBackward"/>
            <xsd:any namespace="##any"/>
         </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
 <xsd:element name="startAt" type="tvax:OffSet"
substitutionGroup="r:condition"/>
 <xsd:element name="endAt" type="tvax:OffSet"
substitutionGroup="r:condition"/>
 <xsd:complexType name="OffSet">
    <xsd:complexContent>
      <xsd:extension base="r:Condition">
         <xsd:choice minOccurs="0">
            <xsd:element name="afterStart" type="xsd:duration"/>
            <xsd:element name="beforeEnd" type="xsd:duration"/>
         </xsd:choice>
      </xsd:extension>
    </xsd:complexContent>
 </xsd:complexType>
```

5.5.2. An example XML License: (Informative)

The following is an example of a license generated for content coming in on digital input with “copy-once” cci bits set.

It has provisions to allow output from the RMP Domain on analog outputs, protected by Macrovision, with the CCI bits set to “copy no more”. It can be viewed, copied, etc. freely within the RMP domain. (See the attached xml file.)

```
<license licenseId="http://www.xrml.org/examples/2001/11/tva-
copyonce/1">
         <inventory>
               <keyHolder licensePartId="domainKey">
                     <info>
                           <dsig:KeyValue>
                                 <dsig:RSAKeyValue>

   <dsig:Modulus>oRUTUiTQkMknAxv6qdF11LKtdToQQmjX0j9q9OIpgOyM4c1T
34rcnDTf84hre4NP7RZYPNnae05g7cQ2W00yzA==</dsig:Modulus>

   <dsig:Exponent>AQABAA==</dsig:Exponent>
                                 </dsig:RSAKeyValue>
                           </dsig:KeyValue>
                     </info>
               </keyHolder>
               <digitalResource licensePartId="theContent">
                     <nonSecureIndirect URI="foo:bar"/>
               </digitalResource>
         </inventory>
         <!-- The content can be rendered by the domain (to the
outside world) over analog w/ macrovision or digital with cci set
to copy no more. -->
         <grant>
               <keyHolder licensePartIdRef="domainKey"/>
               <cx:play/>
               <digitalResource licensePartIdRef="theContent"/>
               <tvax:linkSecurity>
                     <tvax:analog>
                           <tvax:macrovision/>
                     </tvax:analog>
                     <tvax:digital>
                           <tvax:cci>
                                 <tvax:copyNoMore/>
                           </tvax:cci>
```

```
                      </tvax:digital>
                </tvax:linkSecurity>
         </grant>
         <!-- The content can be freely copied (buffered/etc)
within the domain to non-removable media with either no
compression or mpeg2 compression. -->
         <grant>
                <keyHolder licensePartIdRef="domainKey"/>
                <cx:copy/>
                <digitalResource licensePartIdRef="theContent"/>
                <allConditions>
                      <cx:destination>
                            <keyHolder
licensePartIdRef="domainKey"/>
                      </cx:destination>
                      <tvax:destinationMediaInternal/>
                      <tvax:compression>
                            <tvax:none/>
                            <tvax:mpeg2/>
                      </tvax:compression>
                </allConditions>
         </grant>
         <issuer>
                <dsig:Signature>
                      <dsig:SignedInfo>
                            <dsig:CanonicalizationMethod
Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                            <dsig:SignatureMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                            <dsig:Reference>
                                  <dsig:Transforms>
                                        <dsig:Transform
Algorithm="http://www.xrml.org/schema/2001/11/xrml2core#license"/
>
                                  </dsig:Transforms>
                                  <dsig:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>

   <dsig:DigestValue>PB4QbKOQCo941tTExbj1/Q==</dsig:DigestValue>
                            </dsig:Reference>
                      </dsig:SignedInfo>

   <dsig:SignatureValue>AYmqOhSHbiP9JadD2GLBweJdGzNNbwDgFDBtjpRn2
aeW0MGXFF9zmSaN46kylPb7ZQAPozk8Cf5V5u9kQrk5QQ==</dsig:SignatureVa
lue>
                      <dsig:KeyInfo>
                            <dsig:KeyValue>
                                  <dsig:RSAKeyValue>
   <dsig:Modulus>oRUTUiTQkMknAxv6qdF11LKtdToQQmjX0j9q9OIpgOyM4c1T
34rcnDTf84hre4NP7RZYPNnae05g7cQ2W00yzA==</dsig:Modulus>

   <dsig:Exponent>AQABAA==</dsig:Exponent>
                                  </dsig:RSAKeyValue>
                            </dsig:KeyValue>
                      </dsig:KeyInfo>
                </dsig:Signature>
                <details>
                      <timeOfIssue>2001-01-
01T04:03:02</timeOfIssue>
                </details>
         </issuer>
   </license>
```

More specifically, there are two grants contained in the license above. The first is that which grants the content in question can be rendered by the domain (to the outside world) over analog with macrovision or digital with cci set to *copy-no-more.* With the TV Anytime specific extension, expressed with the namespace `tvax`, the grant is given as follows:

```
<grant>
                <keyHolder licensePartIdRef="domainKey"/>
                <cx:play/>
                <digitalResource licensePartIdRef="theContent"/>
                <tvax:linkSecurity>
                      <tvax:analog>
                            <tvax:macrovision/>
                      </tvax:analog>
                      <tvax:digital>
                            <tvax:cci>
                                  <tvax:copyNoMore/>
                            </tvax:cci>
                      </tvax:digital>
                </tvax:linkSecurity>
</grant>
```

The second grant – that the content can be freely copied (buffered/etc) within the domain to non-removable media with either no compression or mpeg2 compression – is given as follows:

```
<grant>
                <keyHolder licensePartIdRef="domainKey"/>
                <cx:copy/>
                <digitalResource licensePartIdRef="theContent"/>
                <allConditions>
                      <cx:destination>
                            <keyHolder
licensePartIdRef="domainKey"/>
                      </cx:destination>
                      <tvax:destinationMediaInternal/>
                      <tvax:compression>
                            <tvax:none/>
                            <tvax:mpeg2/>
                      </tvax:compression>
                </allConditions>
</grant>
```

This grant allows Alice to play the Content given by the crid for 3 weeks.

```
<grant>
                <keyHolder licensePartIdRef="Alice"/>
                <cx:play/>
                <cx:digitalWork>
                      <cx:metadata>
                            <xml>
<tva:TVAMain>
   <tva:ProgramDescription>
         <tva:ProgramInformationTable Version="0.0">
                <tva:ProgramInformation
ProgramID="crid://www.broadcaster.com/theContent" Version="0.0"/>
         </tva:ProgramInformationTable>
   </tva:ProgramDescription>
</tva:TVAMain>
                            </xml>
                      </cx:metadata>
                </cx:digitalWork>
                <validityInterval>
                      <notBefore>2001-11-15T04:03:02</notBefore>
                      <notAfter>2001-12-06T04:03:02</notAfter>
                </validityInterval>
   </grant>
```

This grant also illustrates the ability to carry TV Anytime Metadata in a License. It may be descriptive metadata, or control metadata that interacts with the RMP system, such as metadata to allow segmented playback, or permission to skip commercials. All the information in the License can be made tamper-evident using a signed hash, and some or all of the information can be obscured by encryption.

5.5.3. Serializations

This section is to address the binary representation of the RMPI syntax. This serialization approach should be compatible with that used by TV-Anytime Metadata.

TBD

5.5.4. Transport bindings

This section is to address transport requirements for encoding / transmitting serializations.

TBD

5.6. RMPI and Metadata relationship

The RMPI employed to enforce the usage rules may take a different form from the "informative" RMPI messaging to user interfaces. Wherever applicable, RMPI data and TVA Metadata should use the same language and structure.

TBD

5.7. RMPI Protection

This section is to address protection of RMPI.

TBD

6. Application services API

ORIGINAL TEXT

It is suggested to use the DAVIC (Digital Audio-Visual Council), 1998. DAVIC 1.4.1 specification, part 9, Annex I. See http://www.davic.org/. Please also refer to document AN345 Appendix E.

PROPOSED REPLACEMENT TEXT FOR 6

A standardized API is needed when software from independent third parties is used. So a standardised application API is required only on platforms with this requirement. Examples of such platforms are platforms that support downloaded applications. Devices that do not support such applications do not need to implement this API.

The DAVIC (Digital Audio-Visual Council), 1998. DAVIC 1.4.1 specification, part 9, Annex I is used as the application API. See http://www.davic.org/. Please also refer to document AN345 Appendix E.

The DAVIC CA API addresses the majority of the functionality required for using protected content from an application. It is however likely that some extensions are required to address issues related to storage and networks.

Ed: The original (complete) API (converted from Java to IDL) should be added to the document (in her or in a appendix).

7. RMP services API

ORIGINAL TEXT

It is suggested to use a subset of OPIMA (Open Platform Initiative for Multimedia Access), 2000. OPIMA Specification Version 1.1. http://www.cselt.it/opima/. Please refer to section 2.2 of document AN345.

NEW TEXT

Ed: Replace with the IDL definition of the API as indicated by OPIMA incorporating the changes as indicated below.

The RMP Service API allows an RMP system to communicate in an interoperable way with the RMP Baseline security functions. The RMP Service API consists of the subset of methods from OPIMA[1] as are given in this section. In the following sections, the OPIMA Methods for the RMP API are grouped according to functionality. Note that in OPIMA an RMP system is called and IPMP system.

[1] OPIMA (Open Platform Initiative for Multimedia Access), 2000. OPIMA Specification Version 1.1. http://www.cselt.it/opima/

7.1. Access to content

This part reflects the interface definition of the 'Abstract Access to Content' interface, section 3.3.4.7 of the OPIMA standard. Via this interface, an application can indicate the desired action on the content.

Note:
In OPIMA, the RMP system has little control over the stop-action of the content when the RMP decides that access to the content is no longer allowed (e.g. because a content rule transfers a change in access rights). The only mechanism that is available for the RMP system is to send a wrong decryption key to the TVA-SF. It depends on the implementation of the TVA-SF whether this action will result into a crash of the system. A more graceful shutdown of the content access is necessary as an additional method.

The following methods shall be used for access to content:

- installCallbackContentAccess
- AbstractContentAccess
- replyToContentAccess

Additional methods:

- stopContent(ContentId)

In these methods the following (re)definition of Purpose is used :

Purpose class	Sub class	Description
RELEASE	RENDER	Release the content to another RMP system, only allowing rendering on a device (no storage).
	MOVE	Transfer this content completely to another RMP system.
	COPY	Transfer a copy of this content to another RMP system.
RECEIVE		Receive content from another RMP system.
ACCESS	PROCESS	Process the content without changing the rights (eg. bitrate or content transcoding).
	RENDER	Render the content (includes PROCESS).
	STORE	Store this content on some storage device (includes PROCESS).
	EDIT	Make a copy of the content and edit it (includes STORE).
	DELETE	Delete the content.
OTHER		Other accesses defined in the compartment.

Table 1, Replacement of the OPIMA defined purposes.

7.2. Access to rules/keys

This part reflects the interface definition of the 'Abstract Access to Rules' interface, section 3.3.4.8 of the OPIMA standard. Via this interface the RMP system can access/replace RMPI data.

The following methods shall be used for user interaction:

- obtainUserRules
- obtainContentRules
- newRules
- updateContentRules

7.3. Smart cards

This part reflects the interface definition of the ' Smart Cards' interface, section 3.3.4.6 of the OPIMA standard. The RMP system can access smart cards via this system and send/receive standard ISO 7816 APDUs.

The following methods shall be used for smart card interaction:

- addCTListener
- removeCTListener
- cardInserted
- cardRemoved
- getSlotId
- isCardPresent
- openSlotChannel
- closeSlotChannel
- getATR
- reset
- sendAPDU

7.4. Encryption & Decryption

This part reflects the interface definition of the 'Encryption and Decryption Engines' interface, section 3.3.4.3 of the OPIMA standard. The RMP system can control via this interface both the content cryptography as well as cryptographic actions on miscellaneous data.

The following methods shall be used for encryption and decryption:

- queryEncryptionAlgorithms
- encrypt
- initEncryption
- updateEncryptionKeys

- stopEncryption
- decrypt
- initDecryption
- updateDecryptionKeys
- stopDecryption

7.5. Signatures

This part reflects the interface definition of the 'Signature Engines' interface, section 3.3.4.4 of the OPIMA standard. Via this interface, the RMP system can check and generate both signatures over the content as well as signatures over miscellaneous data.

The following methods shall be used for signatures:

- querySignatureAlgorithms
- verifySignature
- verifyContentSignature
- generateSignature
- generateContentSignature

7.6. Watermarks

This part reflects the interface definition of the "Watermark Engine" interface, section 3.3.4.5 of the OPIMA standard. Via this interface, the RMP system can detect and embed watermarks in the content.

The following methods shall be used for watermarks:

- queryWatermarkAlgorithms
- extractWatermark
- stopWatermarkExtraction
- insertWatermark
- stopWatermarkInsertion

7.7. Access to RMP systems

This part reflects the interface definition of the 'Abstract Access to OPIMA Peers' interface, section 3.3.4.9 of the OPIMA standard. Via this interface RMP systems can interact with each other and the baseline RMP system.

The following methods shall be used for interaction between RMP systems:

- openConnection
- closeConnection
- addConnectionListener
- sendMessage

- newConnection
- receiveMessageFromPeer

7.8. User interaction

This part reflects the interface definition of the 'User Interface', section 3.3.4.1 of the OPIMA standard. Using this interface the user can exchange information with the RMP system.

The following methods shall be used for user interaction:

- sendMessageToUser
- receiveMessageFromUser

The receiveMessageFromUser method only allows for the transfer of strings of characters between the RMP system and the user. The RMP system has no control over the formatting and presentation of the information. To support such formatting in the receiveMessageFromUser method, the MessageText value(s) shall be according to the Common Interface high-level MMI messages [2].

Ed:Make sure reference to CommonInterface is available,Distribute specification or location to TVA.

7.9. Application interaction

This part reflects the interface definition of the 'Abstract Access to Applications', section 3.3.4.10 of the OPIMA standard. This interface defines a transparent bit channel between the application and the RMP system.

Note:
In the TVA framework multiple applications and multiple RMP systems can be present. Therefore this interface will be enhanced with some specific methods to enable the interoperability between applications and RMP systems for some basic functionality.

The following methods shall be used for application interaction:

- installCallbackApplication
- replyMessage
- receiveMessageFromApplication

Additional:
The receiveMessageFromApplication method shall contain the additional Message Type 'QUERY_ENTITLEMENT'. As response to this message type the RMP system shall

[2] CENELEC EN 50221 : 1997, *Common Interface for Conditional Access and other Digital Video Decoder Application*s; and CENELEC R 206-001 : 1997, *Guidelines for the Implementation and Use of the Common Interface for DVB* 15 *Decoder Application*s.

return the list of available entitlements for the current user, via the standard 'replyMessage'.

7.10. Life cycle control

This part reflects the interface definition of the 'Life-cycle Control' interface, section 3.3.4.11 of the OPIMA standard.

The following methods shall be used for life cycle control:

- initialize
- terminate
- update
- remove

8. Security tools

It is suggested to use the tools proposed in section 3.3 (security tool-box) of document AN349. One mandatory common cipher will be specified: AES. An agreed method for the inclusion of additional ciphers, including, but not limited to Camellia, and with specific consideration of ciphers recognized as required by legacy systems (for instance DVB-CSA, M2, 3DES) will be included in the specification.

Editor's Note: the following material is essentially taken from the security toolbox section of AN349 (Section 3.3). Additional sections have been included to:

1. ***Present the security functional requirements of the TVA RMP system,***
2. ***Derive the type of security functions required in the toolbox to address these security requirements,***
3. ***Specify the algorithms and protocols providing the required security functions.***

8.1. Security Functional Requirements

Editor's Note: this section should present the security functional requirements of the TVA RMP system based on the guiding principles and objectives (e.g. end-to-end persistent protection, content interoperability,…).

8.2. Security Functions

Editor's Note: this section should present the type of security functions required in the toolbox to address the identified security requirements of the RMP system (e.g., ciphers, hash algorithms, authentication schemes,…). The actual specification of algorithms and protocols will be addressed in the next Section.

8.2.1. Cipher

A robust cipher is fundamental component of any protection system for hiding content and sensitive data. To enhance device utility and content interoperability for the consumer, a common cipher is desired. This mandates the use of a common cipher among all devices. Other ciphers are not precluded.

8.2.1.1. Content Channel

8.2.1.2. Control Channel

8.2.2. Signature

Digital signature of sensitive data (e.g., certificate) by trusted authority is the common method for ensuring integrity of the data, and verifying that the data has not been tampered by unauthorized entities.

8.2.3. Certificate

Certificates are used to carry and ensure authenticity for data contained within them and certified – signed – by a trusted authority. A hierarchy of certificates and the Certificate-based Authenticated key Exchange (CAKE) may be used in establishing a Secure Authenticated Channel.

8.2.4. Hash

Hash functions are used to condense large data into small one through a unique and (in general, one way) transformation. Further, contrary to CRC, hash functions are designed to resist malicious data modifications (e.g., collision attacks).

8.2.5. Pseudo-Random Number Generator

Pseudo random number generation (PRNG) is pervasively used throughout cryptography as keys, seeds, unpredictable and non-repeating values, etc.

8.2.6. Authentication

The process by which an identity of an entity, e.g., device/module, can be ascertained.

8.2.7. Authenticated Key Exchange

The process by which symmetric or asymmetric keys are securely established or exchanged.

8.2.8. BL-ECM Parser

Baseline-ECM (BL-ECM) is the local ECM to be used within the user network to control content usage states and content decryption key in a protected manner. BL-ECM is by definition processed – parsed and resolved - by the BL-RMP system.

8.2.9. Watermarking

Watermarking (WM) is used to hide important information within the content that needs to be preserved through some subsequent processing and remain in the clear content for future detection. In general content-bound rights such as copy control information bits are embedded as a watermark in the content.

8.3. Algorithms and Modes of Operation

Editor's Note: this section should specify the actual algorithms along with their modes of operations and exposed interfaces which shall be implemented in the security toolbox. These interfaces have to be consistent with the RMP Services APIs (and potentially with the Application Services API).

8.4. Protocols

Editor's Note: this section should specify the actual protocols along with their exposed interfaces which shall be implemented in the security toolbox. These interfaces have to be consistent with the algorithms' interfaces and with the RMP Services API (and potentially with the Application Services API).

8.5. RMP Security Reference Architecture

Below is a pictorial representation of the RMP secure enforcement model and its relationship to RMP Logical architecture in Figure 1.

Editor's note: Eli Hibshoosh to provide diagram in Word friendly format for insertion here.

RMP Security Reference Architecture

8.6. RMP Security Axioms

The following is a list of the security axioms that shall guide the security module implementation.

a.) No clear keys outside the hardware-protected environment (secure chip)

b.) No clear compressed content outside secure HW environment

Ed: potential constraint on proprietary RMP systems when releasing content

c.) Local content license issuance only in response to a given license ffrom a BL or approved proprietary entity; No autonomous generation of license.

d.) All security operations on content or control data must be done inside the hardware-protected environment. Examples:

Input content license checking and modification

Output content license creation and protection

Data integrity check (signature verification)

Encryption/scrambling

Decryption/descrambling

Water-marking (WM) check, WM embedding

Signing sensitive data for exporting outside the security module

Content-license generation

Secure data for external caching

Maintain relative time (session keys update, buffered viewing,)

Key renewability

e.) All sensitive control data that is externally cached must be secure under the control of the security module.

f.) The relative-time counter in the security module cannot be controlled from outside the secure module; exceptions: power reset, special commands by certified entity.

g.) Security module driven by commands (requests) from the controller; the only autonomous actions taken by the security module are in response to hardware reset, timer events (forcing key generation), and time packets from trusted source.

h.) Security module may refuse the controller requests.

i.) The security module must be serialized with an embedded: unique public ID, and a unique secret, sufficiently robust to withstand brute force attack within the next 10 years. (Ref: Rivest - Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security; Lenstra – Selecting Cryptographic Key Sizes)

8.7. Security Functions

Editor's Note: this section should present the type of security functions required in the toolbox to address the identified security requirements of the RMP system (e.g., ciphers, hash algorithms, authentication schemes,…). The actual specification of

algorithms and protocols will be addressed in the next Section. The following list constitutes a set of possible functions. The ones required are yet to be determined.

8.7.1. Pseudo-Random Number Generator

8.7.2. Asymmetric Signature Generation/Verification

8.7.3. Asymmetric Encryption/Decryption

8.7.4. Symmetric Signature Generation/Verification

Ed: Message Authentication Code

8.7.5. Symmetric Encryption/Decryption

8.7.6. Life Period

8.7.7. Check Time

8.7.8. Check Certificate

8.7.9. Select Key

8.7.10. Import Data

8.7.11. Export Data

8.7.12. Create Session Key

8.7.13. Cipher

Editors note: A robust cipher is fundamental component of any protection system for hiding content and sensitive data. To enhance device utility and content interoperability for the consumer, a common cipher is desired. This mandates the use of a common cipher among all devices. Other ciphers are not precluded.

8.7.13.1. Content Channel

Ed: define the list of both mandatory and optional adopted ciphers and the requirement of their implementation.

8.7.13.2. Control Channel

8.7.14. Signature

Digital signature of sensitive data (e.g., certificate) by trusted authority is the common method for ensuring integrity of the data, and verifying that the data has not been tampered by unauthorized entities.

8.7.15. Certificate

Certificates are used to carry and ensure authenticity for data contained within them and certified – signed – by a trusted authority. A hierarchy of certificates and the Certificate-based Authenticated key Exchange (CAKE) may be used in establishing a Secure Authenticated Channel.

8.7.16. Hash

Hash functions are used to condense large data into small one through a unique and (in general, one way) transformation. Further, contrary to CRC, hash functions are designed to resist malicious data modifications (e.g., collision attacks).

8.7.17. Pseudo-Random Number Generator

Pseudo random number generation (PRNG) is pervasively used throughout cryptography as keys, seeds, unpredictable and non-repeating values, etc.

8.7.18. Authentication

The process by which an identity of an entity, e.g., device/module, can be ascertained.

8.7.19. Authenticated Key Exchange

The process by which symmetric or asymmetric keys are securely established or exchanged.

8.7.20. Content License Parser

Baseline-ECM (BL-ECM) is the local ECM to be used within the user network to control content usage states and content decryption key in a protected manner. BL-ECM is by definition processed – parsed and resolved - by the BL-RMP system.

Ed: reword with content license.

8.7.21. Watermarking

Watermarking (WM) is used to hide important information within the content that needs to be preserved through some subsequent processing and remain in the clear content for future detection. In general content-bound rights such as copy control information bits are embedded as a watermark in the content.

8.8. Algorithms and Modes of Operation

Editor's Note: this section should specify the actual algorithms along with their modes of operations and exposed interfaces which shall be implemented in the security toolbox. These interfaces have to be consistent with the RMP Services APIs (and potentially with the Application Services API).

8.9. Protocols

Editor's Note: this section should specify the actual protocols along with their exposed interfaces which shall be implemented in the security toolbox. These interfaces have to be consistent with the algorithms' interfaces and with the RMP Services API (and potentially with the Application Services API).

9. Certification

It is suggested to use what is proposed in sections 3.4 (device characterization mask), 3.8 (certification and keys), 3.10 (revocation), 3.11 (secrets – autonomy and minimum risk for the manufacturer) of document AN349.

10. Device interface

NOT YET PRESENT: THOMSON PRESENTATION PENDING

Philips and NDS are to suggest a solution with reference to AN345 and / or AN349.

11. Glossary