



Automating Production of Cross Media Content for Multi-channel Distribution

www.AXMEDIS.org

DE12.1.3.1

Specification of AXMEDIS 4HOME Architecture

Version: 1.0

Date: 13/7/2007

Responsible: BBC (revised and approved by coordinator)

Project Number: IST-2-511299

Project Title: AXMEDIS

Deliverable Type: report

Visible to User Groups: yes

Visible to Affiliated: yes

Visible to the Public: yes

Deliverable Number: DE12.1.3.1

Contractual Date of Delivery: M30

Actual Date of Delivery: 13/7/2007

Title of Deliverable: Specification of AXMEDIS 4HOME Architecture

Work-Package contributing to the Deliverable: WP12.1.3

Task contributing to the Deliverable: WP12.1.3. (All tasks)

Nature of the Deliverable: report

Author(s): BBC, ETRI, PKU, sDae, TI

Abstract: this part includes the specification of components, formats, databases and protocol related to the AXMEDIS Framework area WP12.1:4HOME including the specification of the 4HOME architecture that will represent the 4HOME demonstrator including the demonstration of the utility of the IP Entity ontology.

Keyword List: AXMEDIS Domain, factory, mobile, PMS, Ontology

AXMEDIS Copyright Notice

The following terms (including future possible amendments) set out the rights and obligations licensee will be requested to accept on entering into possession of any official AXMEDIS document either by downloading it from the web site or by any other means.

Any relevant AXMEDIS document includes this license. PLEASE READ THE FOLLOWING TERMS CAREFULLY AS THEY HAVE TO BE ACCEPTED PRIOR TO READING/USE OF THE DOCUMENT.

1. DEFINITIONS

- i. "**Acceptance Date**" is the date on which these terms and conditions for entering into possession of the document have been accepted.
- ii. "**Copyright**" stands for any content, document or portion of it that is covered by the copyright disclaimer in a Document.
- iii. "**Licensor**" is AXMEDIS Consortium as a de-facto consortium of the EC project and any of its derivations in terms of companies and/or associations, see www.axmedis.org
- iv. "**Document**" means the information contained in any electronic file, which has been published by the Licensor's as AXMEDIS official document and listed in the web site mentioned above or available by any other means.
- v. "**Works**" means any works created by the licensee, which reproduce a Document or any of its part.

2. LICENCE

1. The Licensor grants a non-exclusive royalty free licence to reproduce and use the Documents subject to present terms and conditions (the **Licence**) for the parts that are own and proprietary property the of AXMEDIS consortium or its members.
2. In consideration of the Licensor granting the Licence, licensee agrees to adhere to the following terms and conditions.

3. TERM AND TERMINATION

1. Granted Licence shall commence on Acceptance Date.
2. Granted Licence will terminate automatically if licensee fails to comply with any of the terms and conditions of this Licence.
3. Termination of this Licence does not affect either party's accrued rights and obligations as at the date of termination.
4. Upon termination of this Licence for whatever reason, licensee shall cease to make any use of the accessed Copyright.
5. All provisions of this Licence, which are necessary for the interpretation or enforcement of a party's rights or obligations, shall survive termination of this Licence and shall continue in full force and effect.
6. Notwithstanding License termination, confidentiality clauses related to any content, document or part of it as stated in the document itself will remain in force for a period of 5 years after license issue date or the period stated in the document whichever is the longer.

4. USE

1. Licensee shall not breach or denigrate the integrity of the Copyright Notice and in particular shall not:
 - i. remove this Copyright Notice on a Document or any of its reproduction in any form in which those may be achieved;
 - ii. change or remove the title of a Document;
 - iii. use all or any part of a Document as part of a specification or standard not emanating from the Licensor without the prior written consent of the Licensor; or
 - iv. do or permit others to do any act or omission in relation to a Document which is contrary to the rights and obligations as stated in the present license and agreed with the Licensor

5. COPYRIGHT NOTICES

1. All Works shall bear a clear notice asserting the Licensor's Copyright. The notice shall use the wording employed by the Licensor in its own copyright notice unless the Licensor otherwise instructs licensees.

6. WARRANTY

1. The Licensor warrants the licensee that the present licence is issued on the basis of full Copyright ownership or re-licensing agreements granting the Licensor full licensing and enforcement power.
2. For the avoidance of doubt the licensee should be aware that although the Copyright in the documents is given under warranty this warranty does not extend to the content of any document which may contain references or specifications or technologies that are covered by patents (also of third parties) or that refer to other standards. AXMEDIS is not responsible and does not guarantee that the information contained in the document is fully proprietary of AXMEDIS consortium and/or partners.
3. Licensee hereby undertakes to the Licensor that he will, without prejudice to any other right of action which the Licensor may have, at all times keep the Licensor fully and effectively indemnified against all and any liability (which liability shall include, without limitation, all losses, costs, claims, expenses, demands, actions, damages, legal and other professional fees and expenses on a full indemnity basis) which the Licensor may suffer or incur as a result of, or by reason of, any breach or non-fulfillment of any of his obligations in respect of this License.

7. INFRINGEMENT

1. Licensee undertakes to notify promptly the Licensor of any threatened or actual infringement of the Copyright which comes to licensee notice and shall, at the Licensor's request and expense, do all such things as are reasonably necessary to defend and enforce the Licensor's rights in the Copyright.

8. GOVERNING LAW AND JURISDICTION

1. This Licence shall be subject to, and construed and interpreted in accordance with Italian law.
2. The parties irrevocably submit to the exclusive jurisdiction of the Italian Courts.

Please note that:

- You can become affiliated with AXMEDIS. This will give you the access to a huge amount of knowledge, information and source code related to the AXMEDIS Framework. If you are interested please contact P. Nesi at nesi@dsi.unifi.it. Once affiliated with AXMEDIS you will have the possibility of using the AXMEDIS specification and technology for your business.
- You can contribute to the improvement of AXMEDIS documents and specification by sending the contribution to P. Nesi at nesi@dsi.unifi.it
- You can attend AXMEDIS meetings that are open to public, for additional information see WWW.axmedis.org or contact P. Nesi at nesi@dsi.unifi.it

Table of Content

| | | |
|----------|--|-----------|
| 1 | EXECUTIVE SUMMARY AND REPORT SCOPE | 6 |
| 1.1 | THIS DOCUMENT CONCERNS | 8 |
| 1.2 | CONCEPTUAL ARCHITECTURE..... | 10 |
| 1.3 | LIST OF MODULES OR EXECUTABLE TOOLS SPECIFIED IN THIS DOCUMENT | 11 |
| 1.4 | LIST OF FORMATS SPECIFIED IN THIS DOCUMENT..... | 11 |
| 1.5 | LIST OF DATABASES SPECIFIED IN THIS DOCUMENT | 12 |
| 1.6 | LIST OF PROTOCOLS SPECIFIED IN THIS DOCUMENT | 12 |
| 1.7 | LIST OF EXTERNAL TECHNOLOGIES INCLUDED IN THE 4HOME DEMONSTRATOR | 13 |
| 2 | GENERAL USE CASES AND SCENARIOS | 14 |
| 3 | SPECIFICATION OF THE MODULES TO BE PRODUCED (BBC,ALL) | 15 |
| 3.1 | CONTENT FACTORY (TI) | 15 |
| 3.2 | DISTRIBUTION TO THE INTERNET AND MOBILE ENVIRONMENT (TI) | 16 |
| 3.2.1 | Description..... | 16 |
| 3.2.2 | OMA DRM v2 server architecture..... | 17 |
| 3.2.3 | Mobile scenario implementation..... | 18 |
| 3.2.4 | Execution of the content adaptation rule in the AXCP Rule engine | 19 |
| 3.2.5 | Deployment configuration | 20 |
| 3.2.6 | Nokia N91 technical specifications..... | 21 |
| 3.3 | DISTRIBUTION TO THE NON MOBILE ENVIRONMENT (BROADCAST AND PEER TO PEER), (BBC, ETRI, PKU)... | 25 |
| 3.4 | HOME DOMAIN ENVIRONMENT (BBC, PKU)..... | 50 |
| 3.4.1 | Module or Tools for Establishment of Home Domain..... | 51 |
| 3.4.2 | General Description of the Execution Process..... | 51 |
| 3.4.3 | Module Design in terms of Classes..... | 52 |
| 3.4.4 | User Interface Description | 53 |
| 3.5 | TOOL MANAGEMENT FOR THE PROTECTION OF RECORDED CONTENT IN THE BROADCAST ENVIRONMENT (ETRI) 53 | |
| 3.5.1 | General Description of Tool Management..... | 53 |
| 3.5.2 | System architecture..... | 54 |
| 3.5.2.1 | DRM Protection Tool Server (simplified Tool Server)..... | 54 |
| 3.5.2.2 | Protection Processor in AXMEDIS Client..... | 55 |
| 3.5.3 | Tool management protocol | 56 |
| 3.5.3.1 | Registering..... | 56 |
| 3.5.3.2 | Downloading | 57 |
| 3.5.3.3 | Playing..... | 58 |
| 3.5.4 | User interface description | 58 |
| 3.5.5 | Technical and Installation information | 59 |
| 3.5.6 | Module or Executable Tool Profile..... | 59 |
| 3.6 | SPECIFICATION OF THE AXMEDIS IP ONTOLOGY FOR IP ENTITY REPRESENTATION – INCLUDING ASSOCIATED DEMONSTRATION COMPONENTS. (SDAE)..... | 60 |
| 3.6.1 | IP Ontology..... | 60 |
| 3.6.2 | IP Ontology deployment..... | 60 |
| 3.6.3 | API..... | 60 |
| 3.6.4 | Module or Executable Tool for AxIPOntologyAPI | 62 |
| 3.6.5 | General Description of the Module..... | 63 |
| 3.6.6 | Module Design in terms of Classes..... | 63 |
| 3.6.7 | Technical and Installation information | 64 |
| 3.6.8 | Draft User Manual | 64 |
| 3.6.9 | Examples of usage | 64 |
| 3.6.10 | General OWL classes descriptions: class AxIPOntologyModel..... | 65 |
| 3.6.11 | Instances management: class AxIPOntologyEx | 66 |
| 3.6.12 | Ontology access..... | 68 |
| 4 | FORMAL DESCRIPTION OF FORMAT AXIPONTOLOGY..... | 71 |
| 5 | FORMAL DESCRIPTION OF FORMAT BBC METADATA API..... | 74 |

| | | |
|------------|---|-----------|
| 6 | BIBLIOGRAPHY | 83 |
| 7 | GLOSSARY | 84 |
| TVA | TV-ANYTIME..... | 84 |
| 8 | ANNEX 1 NON MOBILE DEMONSTRATOR FACT SHEET | 84 |
| 9 | ANNEX 2 DEMONSTRATOR FACT SHEET FOR MOBILE DISTRIBUTION..... | 90 |

1 Executive Summary and Report Scope

This document contains the specification of the AXMEDIS-4HOME demonstrator.

The AXMEDIS-4HOME demonstrator is made up of a number of components from the AXMEDIS framework (AX-FW) as identified in the previous deliverable (DE12.1.2.1), plus new extensions and interfaces that will become part of the AXMEDIS framework. In addition, supplementary technologies, not included as part of the AXMEDIS framework but used within the demonstrator to achieve the Use Cases described in DE12.1.2.1, the ‘AXMEDIS-4HOME Use Case and Requirements’ document are required.

The full AXMEDIS specification document has been decomposed in the following parts:

| DE number | Deliverable title | responsible |
|--------------|--|-------------|
| DE3.1.2.2.1 | Specification of General Aspects of AXMEDIS framework, first update of DE3.1.2 part A AXMEDIS-DE3-1-2-2-1-Spec-of-AX-Gen-Asp-of-AXMEDIS-framework-upA-v1-0.doc | DSI |
| DE3.1.2.2.2 | Specification of AXMEDIS Command Manager, first update of DE3.1.2 part B AXMEDIS- DE3-1-2-2-2-Spec-of-AX-Cmd-Man-upB-v1-0.doc | DSI |
| DE3.1.2.2.3 | Specification of AXMEDIS Object Manager and Protection Processor, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-3-Spec-of-AXOM-and-ProtProc-upB-v1-0.doc | DSI |
| DE3.1.2.2.4 | Specification of AXMEDIS Editors and Viewers, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-4-Spec-of-AX-Editors-and-Viewers-upB-v1-0.doc | DSI |
| DE3.1.2.2.5 | Specification of External AXMEDIS Editors/Viewers and Players, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-5-Spec-of-External-Editors-Viewers-Players-upB-v1-0.doc | EPFL |
| DE3.1.2.2.6 | Specification of AXMEDIS Content Processing, first update of DE3.1.2 part C AXMEDIS-DE3-1-2-2-6-Spec-of-AX-Content-Processing-upC-v1-0.doc | DSI |
| DE3.1.2.2.7 | Specification of AXMEDIS External Processing Algorithms AXMEDIS-DE3-1-2-2-7-Spec-of-AX-External-Processing-Algorithms-v1-0.doc | FHGIGD |
| DE3.1.2.2.8 | Specification of AXMEDIS CMS Crawling Capabilities, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-8-Spec-of-AX-CMS-Crawling-Capab-v1-0.doc | DSI |
| DE3.1.2.2.9 | Specification of AXMEDIS database and query support, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-9-Spec-of-AX-database-and-query-support-v1-0.doc | EXITECH |
| DE3.1.2.2.10 | Specification of AXMEDIS P2P tools, AXEPTTool and AXMEDIA, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-10-Spec-of-AXEPTTool-and-AXMEDIA-tools-v1-0.doc | DSI |
| DE3.1.2.2.11 | Specification of AXMEDIS Programme and Publication tools, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-11-Spec-of-AX-Progr-and-Pub-tool-v1-0.doc | UNIVLEEDS |
| DE3.1.2.2.12 | Specification of AXMEDIS Workflow Tools, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-12-Spec-of-AX-Workflow-Tools-v1-0.doc | IRC |
| DE3.1.2.2.13 | Specification of AXMEDIS Certifier and Supervisor and networks of AXCS, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-13-Spec-of-AXCS-and-networks-v1-0.doc | DSI |
| DE3.1.2.2.14 | Specification of AXMEDIS Protection Support, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-14-Spec-of-AX-Protection-Support-v1-0.doc | FUPF |
| DE3.1.2.2.15 | Specification of AXMEDIS accounting and reporting, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-15-Spec-of-AX-Accounting-and-Reporting-v1-0.doc | EXITECH |

1.1 This Document Concerns

The components selected from the AXFW listed above to be included within the AXMEDIS-4HOME demonstrator are listed below.

| DE number | Deliverable title | Responsible |
|--|--|--------------------|
| <u>DE3.1.2.2.1</u> | Specification of General Aspects of AXMEDIS framework, first update of DE3.1.2 part A AXMEDIS-DE3-1-2-2-1-Spec-of-AX-Gen-Asp-of-AXMEDIS-framework-upA-v1-0.doc | DSI |
| <u>DE3.1.2.2.14</u> <u>Protection Support</u> | Specification of AXMEDIS Protection Support, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-14-Spec-of-AX-Protection-Support-v1-0.doc | UPF |
| <u>DE3.1.2.2.10</u> | Specification of AXMEDIS P2P tools, AXEPTool and AXMEDIA, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-10-Spec-of-AXEPTool-and-AXMEDIA-tools-v1-0.doc | CRS4/DSI |
| DE3.1.2.2.3 | Specification of AXMEDIS Object Manager and Protection Processor, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-3-Spec-of-AXOM-and-ProtProc-upB-v1-0.doc | DSI |
| <u>DE3.1.2.2.4</u> | Specification of AXMEDIS Editors and Viewers, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-4-Spec-of-AX-Editors-and-Viewers-upB-v1-0.doc | DSI |
| DE3.2.1C | Framework and tools Specification (Content Production) | DSI |
| DE3.1.2.2.12 | Specification of AXMEDIS Workflow Tools, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-12-Spec-of-AX-Workflow-Tools-v1-0.doc | IRC |
| DE3.1.2.2.2 | Specification of AXMEDIS Command Manager, first update of DE3.1.2 part B AXMEDIS- DE3-1-2-2-2-Spec-of-AX-Cmd-Man-upB-v1-0.doc | DSI |
| DE3.1.2.2.5 | Specification of External AXMEDIS Editors/Viewers and Players, first update of DE3.1.2 part B AXMEDIS-DE3-1-2-2-5-Spec-of-External-Editors-Viewers-Players-upB-v1-0.doc | EPFL |
| DE3.1.2.2.6 | Specification of AXMEDIS Content Processing, first update of DE3.1.2 part C AXMEDIS-DE3-1-2-2-6-Spec-of-AX-Content-Processing-upC-v1-0.doc | DSI |
| DE3.1.2.2.7 | Specification of AXMEDIS External Processing Algorithms AXMEDIS-DE3-1-2-2-7-Spec-of-AX-External-Processing-Algorithms-v1-0.doc | FHGIGD |
| DE3.1.2.2.8 | Specification of AXMEDIS CMS Crawling Capabilities, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-8-Spec-of-AX-CMS-Crawling-Capab-v1-0.doc | DSI |
| DE3.1.2.2.9 | Specification of AXMEDIS database and query support, first update of part of DE3.1.2 AXMEDIS-DE3-1-2-2-9-Spec-of-AX-database-and-query-support-v1-0.doc | EXITECH |
| DE3.1.2.2.11 | Specification of AXMEDIS Programme and Publication tools, first update of part of DE3.1.2 | UNIVLEEDS |

| | | |
|---------------------|---|----------------|
| | AXMEDIS-DE3-1-2-2-11-Spec-of-AX-Progr-and-Pub-tool-v1-0.doc | |
| DE3.1.2.2.13 | Specification of AXMEDIS Certifier and Supervisor and networks of AXCS, first update of part of DE3.1.2 | DSI |
| | AXMEDIS-DE3-1-2-2-13-Spec-of-AXCS-and-networks-v1-0.doc | |
| DE3.1.2.2.15 | Specification of AXMEDIS accounting and reporting, first update of part of DE3.1.2 | EXTTECH |
| | AXMEDIS-DE3-1-2-2-15-Spec-of-AX-Accounting-and-Reporting-v1-0.doc | |

In addition to the deployment of the completely specified items, there are a number of changes, extensions and interfaces to be developed and defined. These are summarized in the following Sections 1.3 to Section 1.6 Section 1.7 lists a number of external technologies to be deployed as part of the demonstration that shows the utility of the AXMEDIS Framework but will remain outside.

General non mobile distribution and Home Domain

- Broadcast formatter (out of scope)
- Broadcast client incorporating
 - Streaming interface
 - Free to Air Broadcast Functionality on the client.
 - Access interface to enhancements (Using AXMEDIA)
- PMS – Domain Home
- PMS – Client
- Domain Manager
- Domain Registration Manager
- AXMEDIA client and network support including publication tool
- DRM Viewer

Downloadable Tools

- DRM Protection Tool Server, (Required)
- IMP Tool packager (External Technology)
- DRM Protection Tool and Metadata storage (Re-use AXFW)
- ID and PW interface (Uses AXFW support)
- DRM Protection Tool Manager (To be added to the AXMEDIS Client)

Ontology

Modules developed in other WPs:

- PMS Server,
- PMS Client,
- Protection Processor,
- DRM Editor and Viewer, (noted as an editor plug in)
- Contract analyser,
- other AXMEDIS Tools.

Modules to be developed in this WP:

- Creation Model Ontology (CMO),
- CMO API (Application Programming Interface).

Content to OMA and non-mobile gateway

- the AXMEDIS Editor and Viewer – to author new content
- the AXMEDIS Content Processing (AXCP) – to adapt content to different formats
- the AXEPTool – to connect in peer-to-peer the AXMEDIS-4HOME Content Factory to all other AXMEDIS Factories
- the Programme and Publication Engine – to distribute protected content over multiple channels

1.2 Conceptual Architecture

The 4HOME demonstrator is conceptually shown below. It consists of a common AXMEDIS Content Factory feeding out into a mobile and non mobile distribution branches, enabling consumption of content from the AXMEDIS factory in both home user domain and mobile user domain environments.

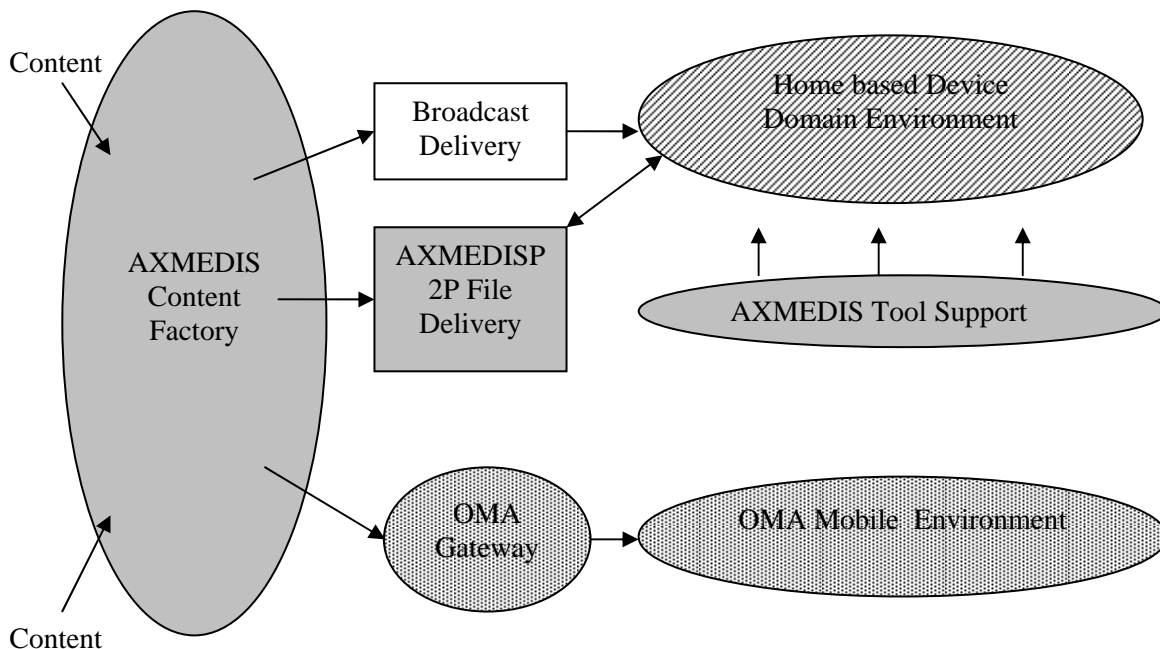


Figure 1 Schematic of Demonstrator Architecture

The arrangement of the components and the way in which they interact with the external technologies are included in the subsequent sections specifying the functional aspects of the demonstrator.

An overview of connectivity between the partners taking part in the 4HOME demonstrator architecture is given in the figure below.

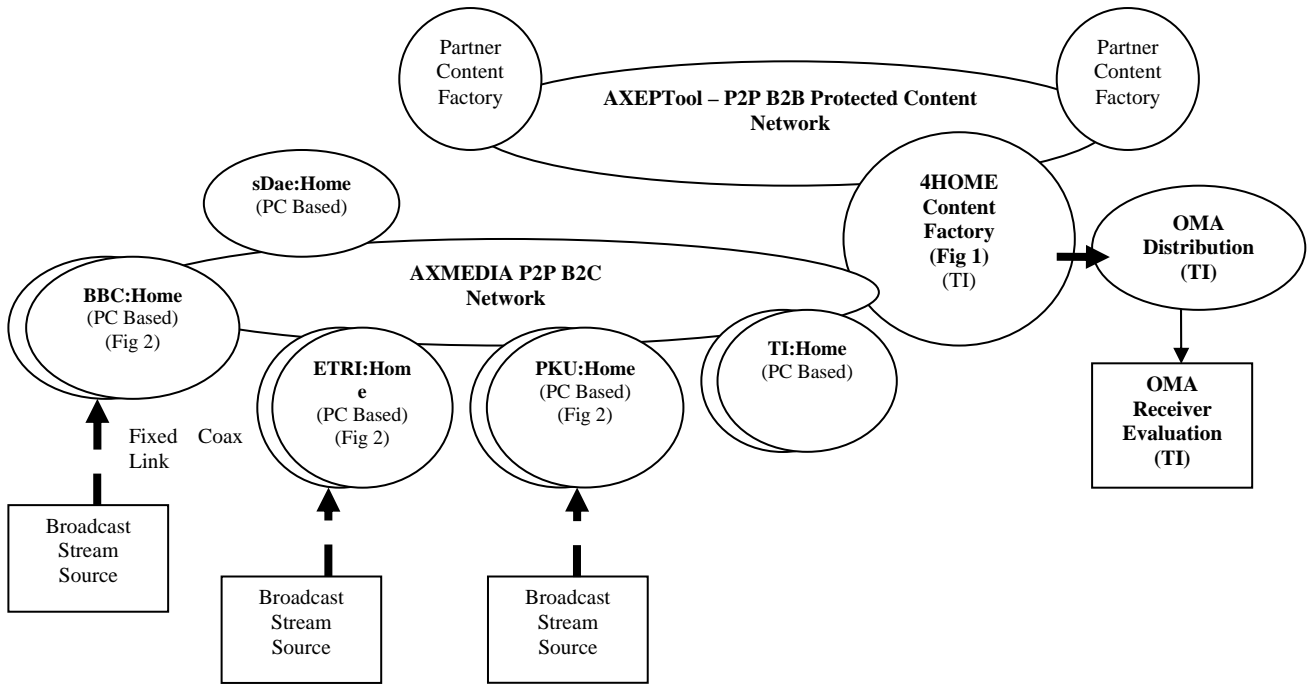


Figure 2 Connectivity of Partners in 4HOME Demonstrator set up

1.3 List of Modules or Executable Tools Specified in this document

| Module/tool Name | Module/Tool Description and purpose, state also in which other AXMEDIS area is used | Standards exploited if any |
|---------------------------------------|---|--|
| FTA Broadcast functionality on client | Record from broadcast stream into AXMEDIS object, To parse and associate metadata, including AXMEDIS ID To identify and retrieve enhancements for trusted clients and bind new object to domain | MPEG-2 and AMD-1 TV-Anytime metadata AXMEDIS presentation formats (Smil) |
| DRM Protection Tool Server | Package Protection Tool Download Protection Tool to the AXMEDIS client | |
| Ontology API Implementation | An implementation of the Ontology API running on ... (Contract Analyser, Protection processor) | |

1.4 List of Formats Specified in this document

| Format Name | Format Description and purpose, state also in which other modules is used | Standards exploited if any |
|--------------------|--|-----------------------------------|
| FTA Signalling | Simple Free to Air Signalling/'broadcast flag' | DVB-GBS |
| Ontology API | To interact with Ontology based licence tool | |
| Protection Tool | Format for downloadable Protection Tool | |

1.5 List of Databases Specified in this document

| Database Name | database Description and purpose, state also in which other AXMEDIS area is using | Standards exploited if any |
|----------------------|--|-----------------------------------|
| Protection Tool DB | Stores and manages registered DRM Protection Tools | |

1.6 List of Protocols Specified in this document

A protocol is a communication modality among distinct processes that can be located or not on different computers.

| Protocol Name | protocol Description and purpose, state also in which other modules is used | Who is the master and who is the slave | Standards exploited if any |
|---------------------------------------|--|---|---|
| Carriage of Licence Over MPEG-2 | To convey the rights and condition or use of the broadcast content over MPEG 2 | Master: Broadcast head end Slave: Broadcast client | MPEG-2 AMD1 and DVB GBS Carriage of Metadata over DVB Streams |
| Download DRM Protection Tool | To convey the DRM Tool to the Broadcast client | Master: Protection Tool Server Slave: AXMEDIS client | TCP/IP |
| Access Free to Air enhancements | Using AXMEDIS ID and enhancements ID to retrieve enhancements to Broadcast captured content. | Master – Client Slave – content server/axmedia | AXMEDIA |
| AXMEDIS broadcast streaming interface | To parse incoming broadcast stream from a DVB-T receiver card | Master: Incoming stream Slave: Parsing client | MPEG-2 |

1.7 List of External technologies included in the 4HOME Demonstrator

| Technology Name | Purpose and Roles within Project. | Standards exploited if any |
|-----------------------------------|--|---|
| TV-Anytime metadata | This is a metadata standard for the description of programme 'attractors'. It would be an alternative to the Dublin Core that may support a richer service proposition over broadcast and P2P/web service. | TV-Anytime |
| Proposed FTA broadcast Signalling | For the most simple Free to Air services the provision of 'Do not redistribute', 'do not scramble' and 'ignore revocation' are required | Not yet standardised |
| MPEG-2 Broadcast Transport. | Current industrial standard for the carriage of broadcast resources and associated data | MPEG 2 |
| MPEG-2 Coding | The format for encoding broadcast resources (Audio/video) | MPEG 2 |
| Programme Metadata Server | To provide programme metadata to the demonstrator | BBC API |
| Broadcast formatter/streamer | To provide MPEG 2 formatted broadcast streams | DVB ASI through DVB-T Modulator. |
| OWL | Web Ontology Language . Used in section 3.7 | W3C Recommendation 10 February 2004 |
| DRM Protection Tool Packaging | To package DRM Protection Tool downloadable to AXMEDIS client | |
| OMA DRM v2 Rights Issuer | DRM server based on OMA DRM v2 specifications. | OMA DRM v2 |
| Root Certification Authority | Root CA issuing certificates for the OMA DRM Rights Issuers, DRM Agents, and OCSP responders. | OMA DRM v2 with CMLA extensions; X.509 certificates |
| OCSP responder | Online verification of certificates, used by the OMA DRM Rights Issuer to verify certificates. | OMA DRM v2; RFC 2560 |
| SMSC | A Short Message Service Center (SMSC) is a network element in the mobile telephone network which delivers SMS messages. It is used by the DRM server to send links and ROAP triggers to the end user's mobile phone. | SMPP v.5 |
| Nokia N91 phone | Client for the mobile scenario | OMA and 3GPP specifications |

2 General Use Cases and scenarios

The general use cases and scenarios that result in this demonstration architecture are reported in deliverable DE12.1.2.1.

The home domain arrangement will enable a usability testing trial to gain feedback on the utility (usefulness) and ease of use of the system as perceived by end users in conjunction with the newly implemented home domain concept.

This will involve observing and tracking how users use the technology and identify parts which do not work well and parts that are well aligned with the user expectation. The tests will be structured to obtain the most meaningful results and try to take into account the short and long term response as the user becomes more expert in the system under test.

The test will be carried out on between 12-20 volunteers drawn from within the BBC, schools and possibly other outside volunteers. They may take place within a user testing laboratory within the BBC, or alternatively at the home, work desk or school of the volunteer, depending on the nature of the trial design.

3 Specification of the modules to be produced (BBC,All)

As this document is the basis of a take-up demonstrator, this specification covers an overall architecture and method of deployment of the tools of the AXMEDIS framework to build the demonstrator. It is therefore not limited to the specification of modules, except where new configurations may require further work by the 4HOME consortium. In particular this is the case for the provision of the home domain architecture, the integration of the streaming interface, third party protection tools and the integration of the OMA gateway.

3.1 Content Factory (TI)

The AXMEDIS Content Factory is the area of the AXMEDIS Architecture (fig. 1) where content is imported, formatted and made available for distribution. In particular, it supports the following operations:

- collecting content from legacy CMSs;
- producing the content,
- programming and scheduling the production process,
- processing metadata,
- composing and formatting content,
- collecting content information from content usage,
- producing licenses to harmonize the production with workflow applications in the factory and among geographically distributed factories.

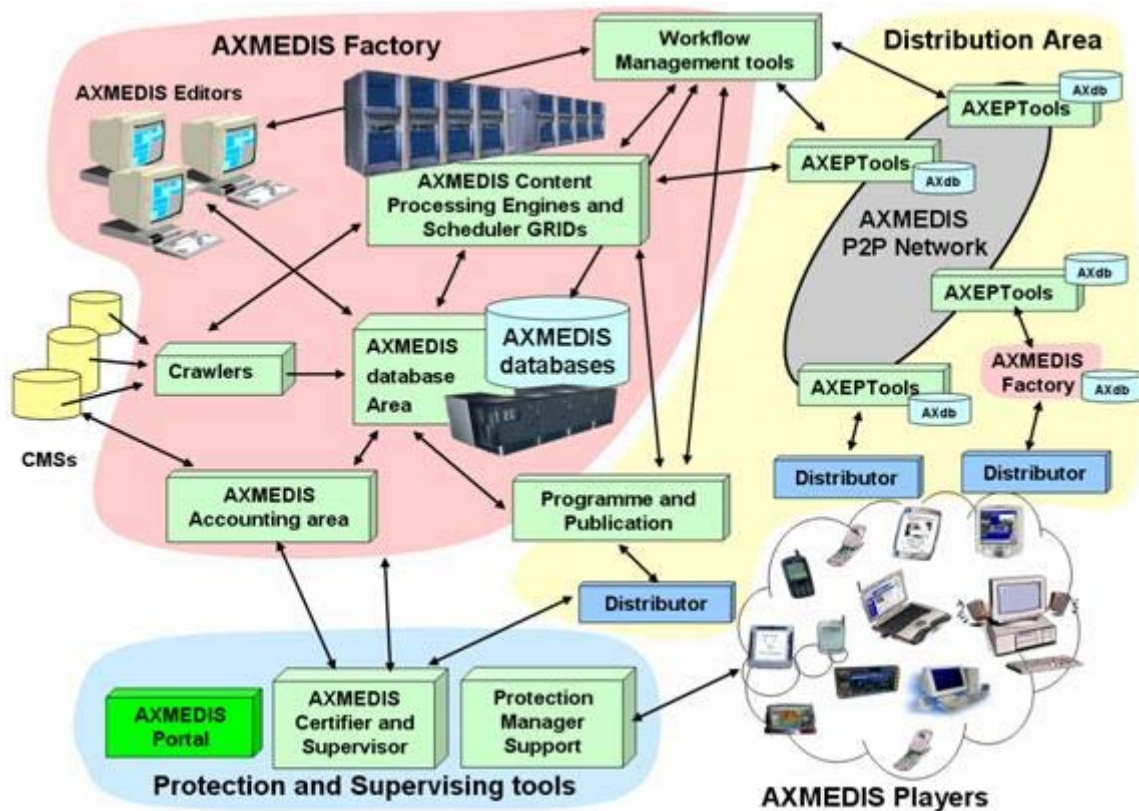


Figure 1: AXMEDIS Architecture

The AXMEDIS Content Factory will be needed in AXMEDIS-4HOME for content preparation and content publication, as described in DE 12.1.2.1 “AXMEDIS-4HOME Use Case and Requirement Analysis”. One instance of the AXMEDIS Content Factory will be set up and maintained operational by TI Labs. In addition some aspects of the AXMEDIS content factory will be replicated by each AXMEDIS-4Home partner involved in content production to support the overall demonstrator.

The components that will be used are:

- the AXMEDIS Editor and Viewer – to author new content
- the AXMEDIS Content Processing (AXCP) – to adapt content to different formats
- the AXEPTool – to connect in peer-to-peer the AXMEDIS-4HOME Content Factory to all other AXMEDIS Factories

Other components of the AXMEDIS Content Factory will be needed only by those partners who operate a distribution channel:

- the Programme and Publication Engine – to distribute protected content over multiple channels
- The AXMEDIS Database – to store content and associated metadata. The database also includes produced licenses for the objects, history of performed actions on content, potentially available rights for each digital resource, models of contracts.
- The AXMEDIS Accounting tools - to collect administrative information about content usage (both business and consumer).

All the above mentioned tools will be used as they are, without modification.

All tools can be downloaded from the AXMEDIS Web site:

- browse the following URL:
http://www.axmedis.org/attivita/documenti/download.php?area_id=1&attivita_id=27&l_s=struttura&gruppo=427&order_by=data&asc_desc=desc
- locate the latest version of
 - axeptool-p2p-setup-v3-2-8.exe
 - axeptool-user-manual-and-installation-guide-v1-2.pdf
 - axmedis-major-tools-full-version-author-grid-v2-7-2.exe
- download and install them

Alternatively, the AXMEDIS Tools can be obtained using the AXEPTool via peer-to-peer communication:

1. first, install the AXEPTool: http://www.axmedis.org/documenti/view_documenti.php?doc_id=2960
2. configure and run the tool as explained in [axeptool-user-manual-and-installation-guide-v1-2.pdf](#)
3. Then, locate axmedis-major-tools-full-version-author-grid-v2-7-2.exe in the catalogue of the axeptool;
4. click on the file name to start the download.

Other tools will be used in AXMEDIS-4HOME, which are not part of the AXMEDIS Factory. These are:

- The Protection Manager Support
- The AXMEDIS Certifier and supervisor
- The AXMEDIS Portal
- The AXMEDIS Player (PC version)

Initially all content factories will share the central PMS Server; subsequently, each partner acting as distributor will have its own instance of the PMS.

3.2 Distribution to the Internet and mobile environment (TI)

3.2.1 Description

The use case scenario for mobile distribution is described in deliverable DE12.1.2.1. In brief, the demonstration shows how a service provider can distribute premium content over fixed and mobile networks in an integrated way. The end user can buy certain Premium content on the service provider portal and then download and play the content (in different formats) on his mobile phone, his PC or both. The goal is to show that the same content (with different quality) may be used on the two clients seamlessly, enabling a business model where the customer can buy the content once, and use it independently of the delivery network (and client device/DRM technology).

In this context the AXMEDIS platform will be used as a unified solution for the distribution of Premium Content over multiple channels: the Internet and the mobile network. It is assumed that the client device will be a PC and a mobile phone respectively. The PC will use the AXMEDIS technology while the mobile phone will use the standard OMA/3GPP technology.

On the Internet the following configuration will be used for content purchase and usage:

- PC Client equipped with the AXMEDIS Player and the AXOM
- AXMEDIS specific Content Format (based on MPEG-21)
- MPEG-21 licenses
- PMS acting as DRM server
- AXMEDIS certificates provided by AXCS

On the mobile network the corresponding configuration will be the following:

- Mobile phone equipped with a native multimedia player and an OMA DRM v2 Agent
- Content format: OMA DCF v2 (based on ISO Base Media File Format)
- OMA DRM REL v2 licenses
- OMA DRM Rights Issuer provided by TI Multimedia as DRM server
- OMA and CMLA compliant certificates provided by TI Multimedia root CA

In order to achieve the above stated goal, the AXMEDIS Framework (including the AXMEDIS Player) will be used without modifications on the Internet, and in connection with an OMA DRM v2 server provided by Telecom Italia for distribution on the mobile network towards OMA DRM v2 mobile devices. The format of content uploaded in the AXMEDIS Content Factory will be adapted automatically using the AXMEDIS tools (AXCP) for mobile distribution. Licenses will be translated as well from their native language (MPEG-21 REL) to the ODRL subset used by OMA DRM REL v2, using the AXCP tool.

The OMA DRM 2.0 Server (which is pre-existing know-how of TI), will be configured to operate as an OMA Gateway in order to be able to receive content from the AXMEDIS platform.

The AXMEDIS components used in the demo will not be modified.

3.2.2 OMA DRM v2 server architecture

The OMA DRM Server is external technology provided by TI Multimedia in order to implement the mobile distribution scenario.

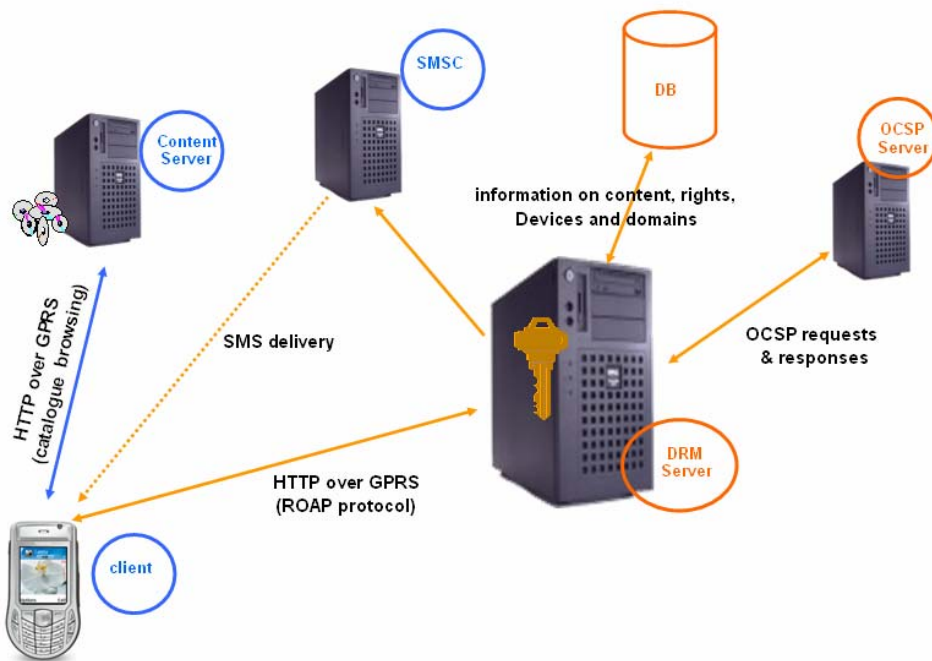


Figure 2: OMA DRM v2 server architecture

The architecture components are:

- Content Server (CS)
 - The web portal where Premium content can be obtained;
- OCSP: online certificate status protocol
 - The OCSP server provides certificate validation and DRM time synchronization;
- DB: relational database containing information and data pertaining to:
 - content and associated rights
 - registered devices and domains (keys, certificates, etc.)
- DRM Server or RI Server
 - Implements the OMA DRM v2 specification; executes the ROAP protocol; deliver Rights Objects (licenses) to OMA clients (via SMSC).
- SMSC: network components sending messages via wap-push from the DRM server to the mobile device (links and ROAP triggers)

3.2.3 Mobile scenario implementation

The implementation of the Internet and mobile scenario assumes that the following operation are performed:

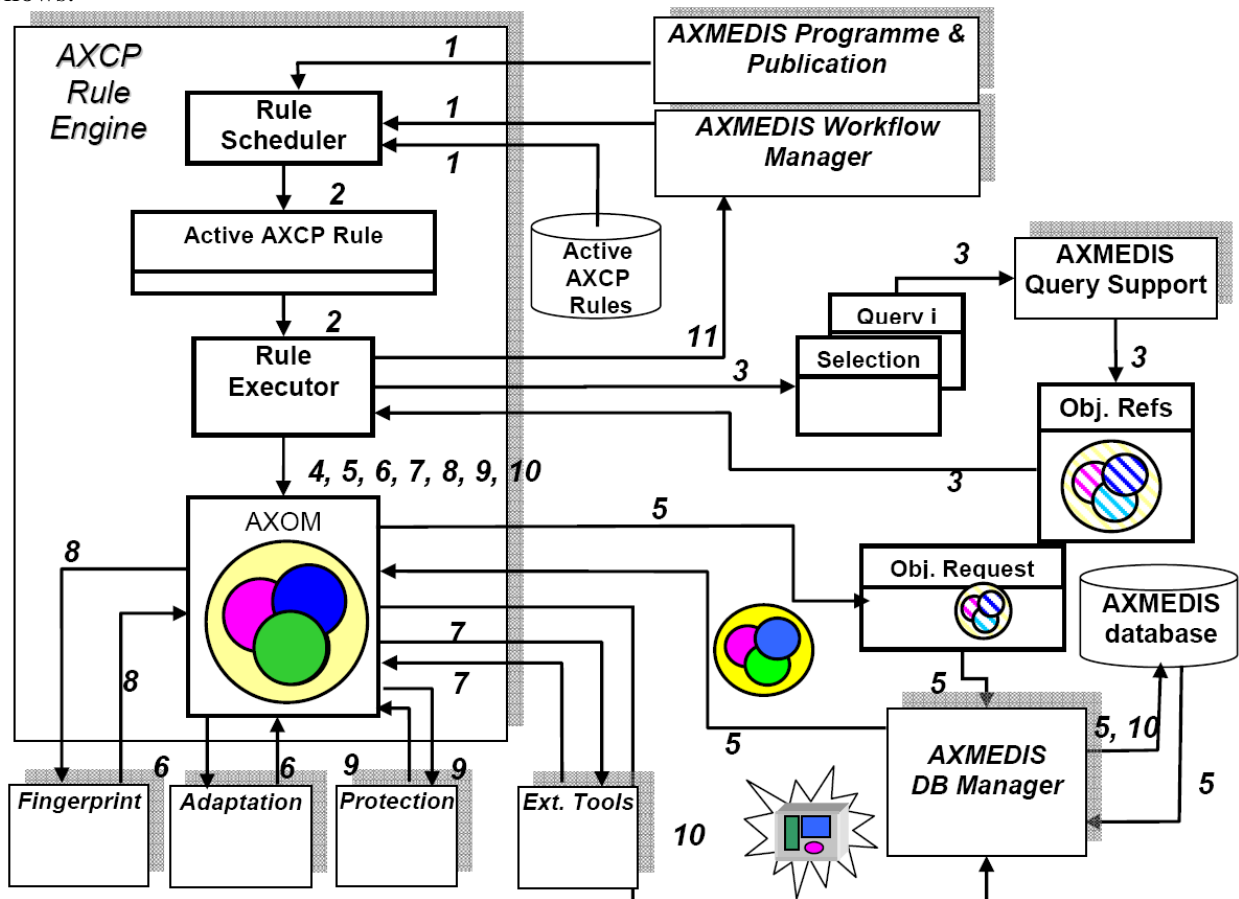
- content is stored in an AXMEDIS B2B distributed database;
- content is protected and licensed to the service provider by means of an AXMEDIS “mother license”;
- the service provider defines two rules on the AXCP in order to
 - adapt content to OMA/3GPP formats;
 - convert licenses from MPEG-21 to OMA DRM REL;
- a check is performed periodically (using the *Rule Scheduler*) to verify if any new content is available on the AXMEDIS B2B database and ready to be published by the service provider (this check may also be performed using the *Programme and Publication* tool by a human operator)
- if such content exists, the adaptation rule is fired and the result of adaptation is posted in cleartext to the OMA DRM server and made available to all service provider’s customers on the mobile network; the content is further processed by the OMA DRM server, which generates a random

encryption key, packages the content in a OMA DRM DCF container, encrypts the payload and stores the key in the DRM server database.

- at the same time, the same content is made available in the original AXMEDIS format on a Web portal to all service provider’s customers on the Internet;
- when a customer buys such content on the service provider portal, a final license is generated for him/her in AXMEDIS format for usage over the Internet;
- subsequently, the rule to convert the license from AXMEDIS to OMA DRM REL is fired and the new license is sent to the OMA DRM server; the license is further processed by the OMA DRM server before delivery of the Rights Object to the mobile device, in order to provide the necessary protection information in the RO, as required by the OMA DRM v2 specification;
- when the end user plays the protected content on the PC, the PMS will provide the necessary protection information as normally expected in AXMEDIS
- when the end user plays the same protected content on the mobile device, the OMA DRM Rights Issuer will provide the necessary protection information as normally expected in OMA DRM v2.

3.2.4 Execution of the content adaptation rule in the AXCP Rule engine

In the following the execution of the content adaptation rule is described in detail using as a template the use case “Execution of a rule” given in DE3.1.2.2.6 – “Specification of AXMEDIS Content Production”. Content adaptation is needed in order to adapt AXMEDIS content to OMA/3GPP formats supported in the mobile environment. The full list of supported formats is provided in the mobile phone specification which follows.



1. **Start process.** The AXCP Rule Engine receives a rule firing request from the AXMEDIS Programme & Publication tool, or from the internal scheduler. Optionally, also the Workflow Manager can generate the request.
2. **Rule execution request.** The scheduler sends the rule execution request to the rule executor with the corresponding rule (*Running rule*).

3. **AXMEDIS Objects selection request.** For each selection and/or query specified in the rule, the rule executor sends queries to the AXMEDIS Query Support to obtain references to AXMEDIS objects that match the request. The query selects objects that are licensed to the service provider for further distribution and have characteristics that make them suitable for mobile distribution (e.g. a limited duration, a simple structure).
4. **AXOM usage.** An object request with the relative object reference is sent to the AXOM to retrieve the object.
5. **Physical Objects request.** The physical object is requested to the AXMEDIS Database by means of its reference.
6. **Adaptation request.** This request is performed via AXOM in order to convert the content format into a 3GPP/OMA compliant format. The following processing may be applied:
 - Perform adaptation algorithm (change resolution, change dimension, time or spatial best fitting, etc...)
 - Apply spatial and temporal constraints specified in the rule (i.e. graphical layout, temporisation, transitions effects, etc...)
 - Perform synchronisation algorithm (audio and text audio and images, video and text, etc...)
 - Convert the whole formatted object into a specific output format (i.e. MPEG or 3GPP).
7. **External tools calling.** The result of adaptation is posted (in cleartext) to the OMA DRM server. In this way some formatting operation are delegated and performed in the OMA DRM server environment. This operation exports the object from the AXMEDIS environment and imports it under the OMA DRM server control.
8. **Fingerprint request.** This operation is not required.
9. **Protection request.** This operation is not required.
10. **Storing AXMEDIS object.** This operation is not required.
11. **End process notification.** The end of the formatting process is notified to the AXMEDIS Workflow Manager.

3.2.5 Deployment configuration

Mobile client

Two Nokia N91 phones equipped with DRM Agent certificates provided by TI Multimedia will be made available as client devices for experimentation and demonstrations. The Nokia N91 mobile phone that will be used is a commercial phone which is currently available on the market. A summary of the N91 technical specification is provided below.

Internet client

A standard PC running WinXP and the AXMEDIS Player will be used.

Content type

Content delivered to end users will be of type:

- Still pictures
- digital music
- digital video

Content format

The following content formats will be used:

- still pictures:
 - compression: JPEG
 - resolution: 640 x 480
- digital music
 - compression: AAC
- digital video
 - 3GPP H.263 or MPEG-4

Mobile network

The cellular network infrastructure and the SIM cards needed in order to perform the demonstration will be provided by Telecom Italia (TIM) in Italy (roaming will not be supported).

AXMEDIS Content Factory server

A server located in Turin and configured as AXMEDIS Content Factory is available online since April 2007. This server will be maintained operational at full time and will host all the AXMEDIS tools that are needed to implement the Internet and mobile scenario.

The server hosting the AXMEDIS Content Factory for the Internet and mobile scenario is the following:

- Hostname: host218-96.pool8290.interbusiness.it
- IP address: 82.90.96.218
- PC model: HP xw4100 2.7 GHz dual Pentium

The server is reachable at <https://82.90.96.218/showLogon.do> with proper credentials.

OMA DRM v2 Right Issuer

The OMA DRM v2 Right Issuer is a DRM server based on OMA DRM v2 specifications. It is external technology provided by TI Multimedia, and is available online at full time at the following address:

<http://atlantis.tilab.com:8080/ROAP/ROAPWEB.SendTrigger?type=%3ctriggerType>

Root Certification Authority

A root CA issuing certificates for the OMA DRM Rights Issuers, DRM Agents, and OCSP responders is installed on the same server. This is external technology provided by TI Multimedia.

OCSP responder

A service that provides online verification of certificates, used by the OMA DRM Rights Issuer to verify certificates, is also installed on the same server.

The server hosting the above mentioned services is the following:

- hostname: atlantis.tilab.com
- IP address: 163.162.91.85
- PC model: HP PROLIANT DL 380 G4

SMSC

A Short Message Service Center (SMSC) is a network element in the mobile telephone network which delivers SMS messages. It is used by the DRM server to send links and ROAP triggers to the end user's mobile phone.

3.2.6 Nokia N91 technical specifications

A summary of the Nokia N91 mobile phone technical specification is provided here for the reader convenience. The full technical specification is available at <http://forum.nokia.com/devices/N91>

General

Operating frequency and Data transfer*

- GSM/GPRS/EDGE 900/1800/1900MHz + WCDMA 2100 **
- Data transfer: WCDMA download up to speed 384 kbps, GPRS/EGPRS (class B/multislot class 10) download up to 236.8 kbps

* Actual achieved speeds may vary depending on network support

** WCDMA frequency not available in the US

Dimensions

- Weight: 164 g
- Length: 113.1 mm
- Width (max): 55.2 mm
- Thickness (max): 22 mm

Display and User Interface

- Color display (176 x 208, 262k color)
- Automatic brightness control for display and keypad
- Music keys: Play/Pause, Forward, Back, Stop. Switch easily between music player and last used application with dedicated key.
- Keys: Two configurable soft keys, Send & End, 5-way configurable navigation key, keys for S60 (Menu, Edit, Clear), ITU-T (number/text) keypad, Volume (up & down), Power, Device lock switch
- Operating system: [S60 software on Symbian OS](#)

Memory

- Up to 4GB of internal dynamic memory for music content (up to 3000 songs*), multimedia messages, ringing tones, images, video clips, and applications
- Additional 30MB of storage for contacts, calendar notes and events, text messages.

* Capacity based on 3:45 minutes per song and 48 kbps eAAC+ encoding. Capacity with 128 kbps AAC encoding is up to 1000 songs

Digital Rights Management

- For protection/digital rights management for music files: Open Mobile Alliance (OMA) DRM 2.0 and Windows Media (WM) DRM 10 for portable devices

Power Management

| | |
|-----------------|----------------------------|
| Standby time: | GSM up to 190 hours* |
| | WCDMA up to 170 hours* |
| Talk time: | GSM up to 3 - 4 hours* |
| | WCDMA up to 2.5 - 3 hours* |
| Music Playback: | up to 10 hours* |

Imaging

Camera

- 2 Megapixel
- Image viewers: JPEG, EXIF, GIF, BMP, PNG
- Still Image resolutions: 1600 x 1200 (default), 640 x 480
- Still Images file format: JPEG

- Advanced camera modes: still, burst, video and options for night, brightness adjustment, image quality, and self-timer.
- 20x digital zoom

Video

Video Formats

- Video playback: 3GPP H.263, MPEG4, RealVideo
- Image viewers: JPEG, EXIF, GIF, BMP, PNG
- Download and play multimedia files (video and music)
- Stream media files from compatible media portals
- Full-screen video playback on the phone to view downloaded, streamed or recorded video clips in larger size

Music

Music Player

- Intelligently designed music player - optimized for music listening
- Dedicated playback keys and volume control
- Enhanced on the go playlist management - edit playlists, add new music
- Meta data for tracks can be viewed and modified directly on your N91
- Enhanced music features: mixer, 5-band equalizer, loudness, music DRC (Dynamic Range Compression)
- See what is playing when the Nokia N91 is in the idle state
- Answer calls while listening to music and music pauses and then resumes again when the call is ended
- Smart integration with music services (easy access to music shop)
- Music user profile or sim-less operation allows for uninterrupted listening
- Transfer any multimedia content from PC to your Nokia N91
- Download unprotected** new music over the air to your Nokia N91

Audio formats (decoding)

- MP3, AAC, AAC+, eAAC+, Real, WAV, WMA, M4A, AMR-WB, True Tones, AMR-NB

Hi-Fi Quality Sound

- Use the Nokia Music Headset HS-28 with remote control or use your own high quality headphone and connect directly to the 3.5mm audio jack on the Nokia N91

- Connect the Nokia N91 to external speakers via the CA-63U stereo audio jack

Connectivity

- 3G
 - Near broadband speeds for faster and more responsive downloads.
- WLAN 802.11b/g
 - WLAN for hot spot connectivity at your favorite café or meeting place.
- USB
 - USB 2.0 full speed interface with mass storage profile (USB mini)
- Nokia PC Suite
 - Connectivity with USB or Bluetooth wireless technology
- Synchronization
 - Local synchronization of contacts, calendar and music files to a compatible PC using compatible connection
 - Synch contacts and calendar wirelessly over Bluetooth
- Bluetooth
 - Integrated Bluetooth wireless technology v.1.2
 - Send and receive images, video clips, graphics, and business cards via Bluetooth wireless technology
 - Profiles with Bluetooth connectivity: Basic Printing Profile (BPP)

Browsing

- XHTML browser
- Improved Web compatibility with support for HTML 4.01, including support for elements such as image maps, background images, and frames
- Support for a subset of JavaScript 1.5, which includes the most commonly used functions found on the Internet
- File upload over HTTP using standard HTML forms
- Small screen rendering option including a faster page-up/page-down scrolling style
- Full-screen mode, download progress bar, and adaptive history list
- 3GPP video streaming
- OMA DRM 2.0 including forward lock for content protection, combined delivery and separate delivery
- Wallet: convenient online use and storage of your numbers and passwords

Java Applications

- Java™ MIDP 2.0, CLDC 1.1 (Connected Limited Device Configuration (J2ME)), JTWI 1.0
- Over-the-air download of Java-based applications and games

3.3 Distribution to the non mobile environment (Broadcast and Peer to Peer), (BBC, ETRI, PKU)

This functionality is really a combination of functions already offered by the AXMEDIS framework, although some of the functionality will be incorporated into extensions of the current modules.

The Diagram below shows the overall architecture schematic for the non mobile environment. It builds on the common Content Factory used by the 4HOME team for the distribution of content to the mobile and non mobile branches.

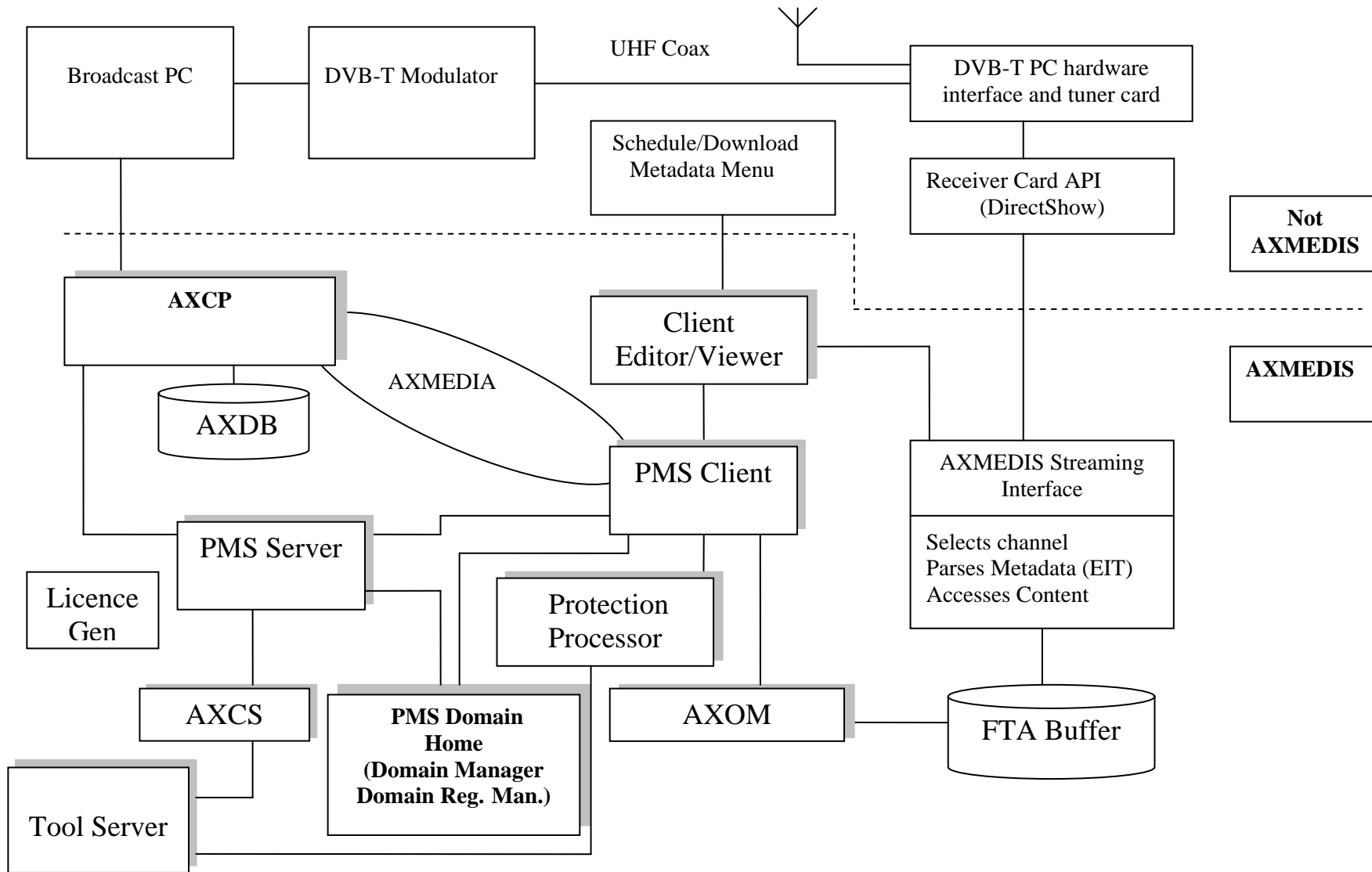


Figure 3 Overall Schematic of non-mobile branch

This branch is intended to harmonise the user experience between incoming broadcast streams and the availability of content on demand over P2P whilst exercising the functionality offered by the AXMEDIS client including redistribution control and control limited to domains. The policy to be enforced on incoming broadcast content may depend upon the availability of a downloadable tool.

The branch will comprise the following.

(i) AXMEDIS broadcast streaming interface

For receiving streaming, the AXMEDIS broadcast streaming interface will permit;

- Parse MPEG TS from say DVB-T card eg, Hauppauge or WinTV card (can be USB connected)
- Extract Single TS from the Multiplex
- Extract a resource and related Info carried as metadata – this may include the related P2P AXMEDIS object ID. The resource within the incoming transport stream will be identified through the service location and time, possibly also the DVB event Id can be used.

Current DVB-T cards are available that are supported by the Linux DVB API. The Linux DVB API is implemented in modern distributions of Linux, and there may be a number of utilities which offer programmatic access using Java at a high level to enable the identification and extraction of metadata and resources.

Windows streaming DVB-T cards use the Broadcast Driver Architecture (BDA) and may be less easy to build an application over due to its low level and specialist MS Windows programming requirements. Known applications that operate over these cards include Winamp and a selected few Open Source projects using GPL. Some of these are listed in the following table.

| Project Title | Description | Licence /Comments |
|---------------------|---|---|
| DVB Decoder | This programme allows you to analyse the tables and packets within a DVB Transport Stream either from a recorded file or live (dependant on hardware). | GNU General Public License (GPL) OS: 32-bit MS Windows (NT/2000/XP) C#, C++ |
| DVB Central in Java | A set of java libraries and applications to capture and save and distribute a DVB stream on a Linux box with a DVB-T card. | Common Development and Distribution License, Linux, Java |
| MeuhMeuhTV | Documented in French Les futures fonctionnalités de MeuhMeuhTV : <ul style="list-style-type: none"> • Support des cartes disposant des drivers BDA (notamment de la TNT). • Prise en charge des cartes analogiques stéréo. • Support de l'EPG pour la TNT. | OS: 32-bit MS Windows (NT/2000/XP) License: GNU General Public License (GPL) |

| | | |
|------------------|---|---|
| | <ul style="list-style-type: none"> • Skinnage qui permet de changer l'aspect de MeuhMeuhTV (une dizaine de skins sont déjà disponibles). • Enregistrement des émissions analogiques et numériques. • Possibilité de choisir les codecs (son et vidéo) pour l'enregistrement. • Lecture des fichiers AVI et MPEG. • MeuhmeuhTV devient multi-utilisateur. | |
| DirectShow .NET | <p>NET c# directshow lib, capture and playback using c# and directshow</p> <p>Includes</p> <p>Samples\BDA\DTVviewer - use BDA to display Digital TV into a Windows Form.</p> <p>Samples\VMR9\VMR9Allocator2 - an alternative to the DirectShow VMR9Allocator program.</p> <p>Samples\Misc\Toolkit – A collection of useful utilities.</p> | <p>OS:32-bit MS Windows (NT/2000/XP), Microsoft Windows Server 2003</p> <p>License: GNU Library or Lesser General Public License (LGPL)</p> <p>C#</p> |
| BDA Developments | <p>A group of user interface projects based around Microsoft's new BDA architecture for digital TV. We will be primarily focused on DVB-T support in the UK using the Nova-T card but hope the applications will have wider application.</p> <p>Not really started.</p> | <p>OS:32-bit MS Windows (95/98), All 32-bit MS Windows (95/98/NT/2000/XP)</p> <p>License: GNU General Public License (GPL)</p> <p>C++ Visual basic.</p> |
| MediaPortal | <p>Includes ‘mytv’ section for recording.</p> <p>Supported by many cards.</p> <p>A comprehensive TV and Media player on PC.</p> <p>MediaPortal opens the portal to all</p> | <p>Operating System: 32-bit MS Windows (NT/2000/XP), Win2K, WinXP</p> <p>License: GNU General Public License (GPL)</p> <p>C#</p> |

| | | |
|---------------|--|---|
| | your media. Listen, record and organize music, movies, radio, streams, pictures and even pause TV! Use internet sources to enrich your media with albumart or songnames. Enjoy your mediacenter or HTPC like never before! | |
| MythTv Player | MythTv Player enables playback of MythTV recordings from Windows. It connects directly to the MythTv backend and does not require any external codecs. | Operating System: All 32-bit MS Windows (95/98/NT/2000/XP) License: GNU General Public License (GPL) |

Given the need to demonstrate a number of AXMEDIS Tools from the AXMEDIS Framework working in concert on a home user based environment it may be that a choice of Windows PC is most convenient and appropriate if code such as that available from DirectShow .NET can be utilized. In recent weeks the Microsoft support for the DirectShow interface has also been increased opening the possibility of writing to the Directshow interface, including the BDA interface in C++. This latter approach will also be considered as it will provide more portable code on the application side and would be the more favourable option if possible.

The AXMEDIS object ID can be conveyed from a metadata field ('Other Identifier') or a table of AXMEDIS object ID mapping from DVB event ID made available. The latter approach will give the benefit of reducing the broadcast complexity and lessens the technological dependence between broadcast signal and AXMEDIS platform, allowing for the possibility of multiple enhanced content service providers working from an independent broadcast stream.

(ii) FTA Broadcast AXMEDIS object creation. (Achieved by the AXMEDIS Object Manager AXOM)

The client side AXOM encapsulates the functionality for building the AXMEDIS object, including license/domain key with 'broadcast enhancements'

The following processes are to be scripted and enabled on the client side using the AXMEDIS scripting language functionality to control the client side AXOM. So far there are a number of operations to be enabled.

Step 1: Determine enhancements from;

Client Usage history (AXMEDIS already has this)

Content Id conveyed within the broadcast metadata of the broadcast stream and retrieved from Streaming Interface, or derived from mapping

This requires an algorithm to look up from the DVB Event ID and the Usage History to determine the enhancement content to be used or accessed from the Content Factory. This must be subsequently accessed securely from the Content Factory and collected for presentation by the client.

Step 2: Enact Enhancement download protocol (Over AXMEDIA) for;

Promotions (video if possible)

Added metadata

New AXMEDIS object ID and licence?

Some enhancement content might already be on the user client.

- Step 3: Store as an AXMEDIS browsable item
 - Transcode video segment to [H.264, 1.2Mbit/s], Audio??
 - Associate a Licence
 - Bind to Domain

The AXMEDIS client may need to utilize a FTA buffer to capture the incoming content whilst making the necessary access to the Content Factory support.

The packaging of the free to air as a user-personalized package may involve the AXMEDIS smil functionality and associated player. The template for the smile script will be available to the client??

The binding to the Home domain is achieved through the home domain arrangement managed by the PMS client and associated PMS support modules described in the section below.

The operation on the client of creating the enhanced FTA package as an AXMEDIS object will utilize much of the current functionality of the AXMEDIS client software and in the event may not be an identifiable module but a set of actions performed by a number of existing AXMEDIS Tools, in particular be controlled and accessed through the AXMEDIS Editor Software Architecture which includes internal and external viewers, the AXMEDIS Object Manager and Command Manager. (See AXMEDIS DE3.1.2a-axfw-spec-(general and model)-part-a-v2-1, and DE3.1.2.2.4 – Specification of AXMEDIS Editors and Viewers, first update of DE3.1.2 part B. The scripting tools are documented in the addition to DE5-0-1-1 AXMEDIS Content Processing GRID: Script Language User Manuals

The following schematic shows the conceptual operation implemented on the AXMEDIS client to create the Enhanced object and depends heavily on the scripted authoring capabilities of the AXMEDIS client.

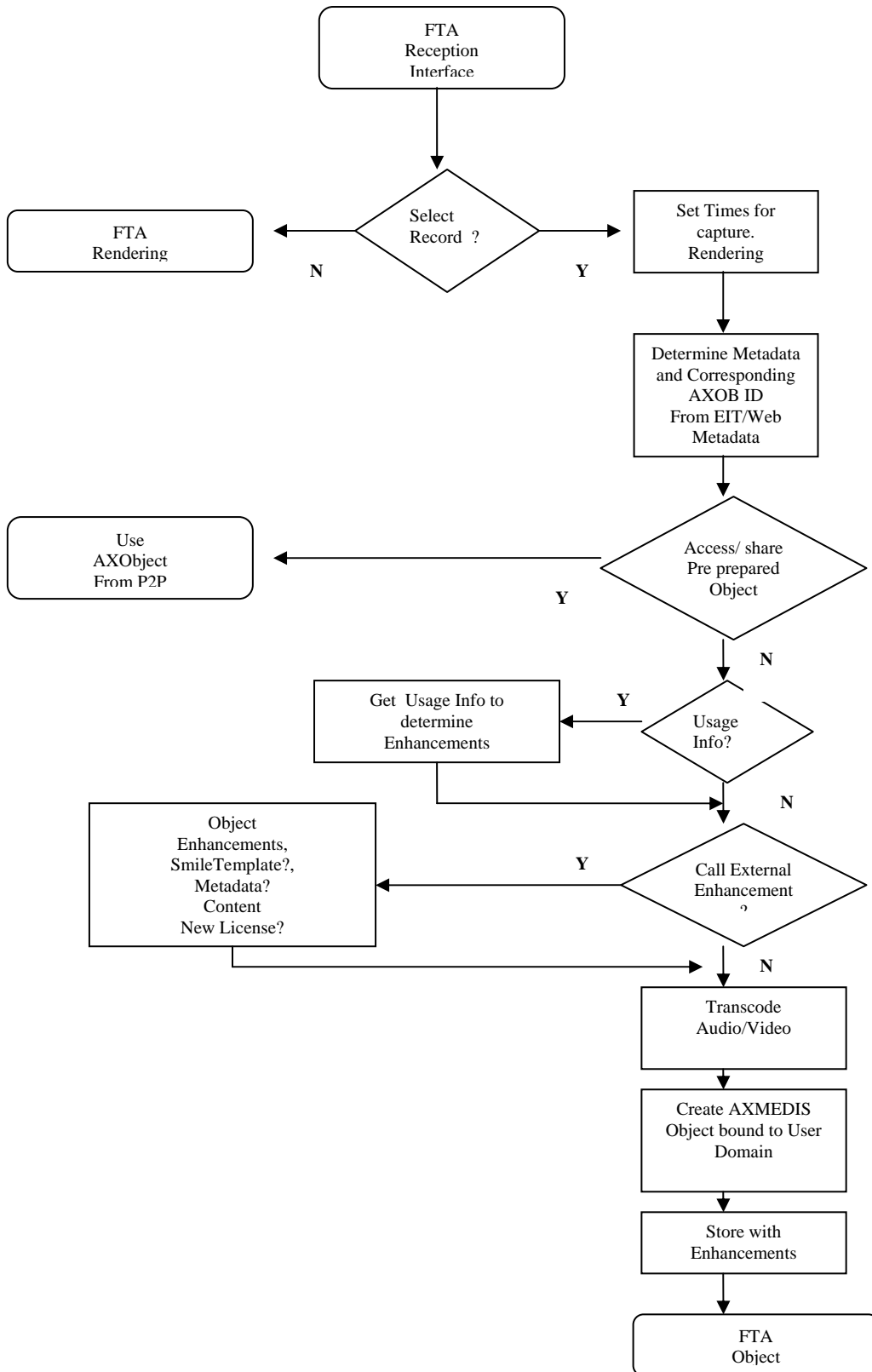


Figure 4 Schematic of scripting Operations to Create the Free to Air AXMEDIS object on the client.

Analysis of scripting tools in context of the above diagram

Set times for Capture/rendering

Calls to the DirectShow DVB interface.
Calls to the AXMEDIS renderer to send resources to the disc between times (or events)
MPEG-TS – then convert to MP4 H.264
Create Resource

Determine Metadata and corresponding AXOB ID

Calls to the DirectShow DVB interface.
Lookup the DVB Event_ID to AXOB ID
EIT P/F This may be Broadcast, web server or local lookup

Determine Usage History/Profile

Use JSAXUserProfile

Download Supporting Content from AXEPTool

AXP2P Manager::
DownloadedObjectListResponse
DownloadStatusResponse
PublicationStatusResponse
PublicObjectListResponse

Calls to Metadata Service

HTTPConnection
WebserviceConnection
Then use the CGI style calls to BBC interface

Transcode Audio/Video

Use of Adaptation Plug in (ToMP4)
AxResource methods
AXMetadata
AxInfo

Create and Bind AXMEDIS Object

Calls to PMS/Domain Home
License Object calls
PMSClient
(Needs Domain Binding module)

Store

I/OFunctions::
File Functions
DIR Functions
Process Functions

Note that whilst in principle the flow of Figure 4 above will be used to achieve the Use Case outlined in DE12.1.2, the AXMEDIS framework may also be able to support the downloading of scripts to enable more flexibility to the processing order if required.

(iii) P2P access to content from Content Factory.

This will use the AXMEDIA tool, driven automatically by the AXMEDIS Client. The AXMEDIA content request can use the AXMEDIS content ID retrieved from the broadcast service, or downloaded from the Web service of downloadable content.

The AXMEDIA Tool is specified in AXMEDIS-DE3-1-2-2-10-Spec-of-AXEPTool-and-AXMEDIA-tools-v1-0.doc

The AXMEDIS client should also permit live viewing of the input selected from the broadcast streaming interface, ie the Streaming interface should be integrated with the AXMEDIS Players for rendering on the PC in real time, though not in a way that encourages storage without domain binding

(iv) P2P access to enhancements to content for ‘adaptation’ and domain binding.

Content promotions or audio/video enhancements will be sourced from Content Factory over P2P as with all content.

Enhancements are identified by the AXMEDIS client and Content Factory using both the AXMEDIS ID derived from the broadcast metadata, DVB Event_ID and the User History available. The implementation of the packaging of enhanced FTA resources should be automatic and stimulated on record.

(v) Content control enforcement within a user domain.

The PMS Client will be used in conjunction with the PMS Domain Home and associated (implied) administrator to bind the content to the Domain.

The PMS Modules are specified in DE3.1.2.2.14 Specification of AXMEDIS Protection Support, first update of part of DE3.1.2 and DE4.5.1.2 Content Protection and Supervision, 1st update

This will permit (as described in Section **Errore. L'origine riferimento non è stata trovata.**);

- Setting up a user home domain – Using AXMEDIS PMS client
- Registering a client device in the home domain – This may need an extension to the Protection Management Support Tools
- Adding and removing devices from the AXMEDIS domain.

The above are described in Section **Errore. L'origine riferimento non è stata trovata.** Under these conditions, the following conditions can be enforced on the enhanced-FTA AXMEDIS object.

- Binding content, as per license, to devices registered within the domain.

Transfer between home devices – PMS will implement license conditions and prevent general distribution . The local license given to the newly formed AXMEDIS object will specify this behaviour. Ensure the license can specify this to the PMS. Section **Errore. L'origine riferimento non è stata trovata.** below describes the arrangement of the 4HOME domain.

(vi) Client Player/Viewer

This module provides the access to the home library through the AXMEDIS content interface, using metadata associated with the content to provide the title etc. This will utilise the AXMEDIS Player and will show Content Rights status by indicating which items are bound to domain and which are not.

It will include a ‘Push’ option to other domain devices facility.

As the client Player/Viewer is the main integrated user interface, it will provide access and control of other modules, eg,

- Selection of the tuning for the incoming DVB Stream
- Selection of recording – and enhancing of content from DVB FTA
- Selection of the protected Content to be downloaded over P2P
- Viewing of the Rights associated with content

This functionality will be available through the AXMEDIS Editor Software Architecture which includes internal and external viewers, the AXMEDIS Object Manager and Command Manager. (See AXMEDIS DE3.1.2a-axfw-spec-(general and model)-part-a-v2-1, and DE3.1.2.2.4 – Specification of AXMEDIS Editors and Viewers, first update of DE3.1.2 part B

(vii) The Web service – a menu of available downloadable content

This is a web based service providing a list of downloadable content that can be accessed over the AXMEDIS P2P service.

It can be provided using the TV-Anytime programme metadata standard. It can be accessed either directly through a simple API made available from the BBC backstage or repurposed for the demonstrator with an AXMEDIS interface (this may be necessary in order to add the AXMEDIS object ID into the received data)

The application to access the interface and pass the data to the AXMEDIS client user interface can be built in Java.

(viii) The Web Service client

This module will render the content website, extract the AXMEDIS P2P object ID from the user selection, (ie parse the metadata fragment from the HTTP Get command) and initiate an AXMEDIA download action.

The API is typically designed to be respond to queries from clients using HTTP ‘Get’ and using a query method in the URL, delimited with a “?”. The name value pairs that follow are delimited with “=” and different parameter name value pairs are separated by “&”.

Note that the BBC is currently updating this API namespace from “bbc” to “broadcast” to enable more open implementations. The upgrade is not yet complete at the time of writing.

The method calls are categorized under five headings as listed below.

channel

- bbc.channel.getInfo
- bbc.channel.getLocations
- bbc.channel.list

genre

- bbc.genre.getInfo
- bbc.genre.getMembers
- bbc.genre.list

group

- `bbc.group.getInfo`
- `bbc.group.getMembers`
- `bbc.group.list`

programme

- `bbc.programme.getInfo`
- `bbc.programme.getLocations`
- `bbc.programme.search`

schedule

- `bbc.schedule.getProgrammes`

The following is a short list of error codes which may be thrown by the API.

| Code | Description |
|------|--------------------------------------|
| 1 | programme_id "xxx" not found |
| 2 | channel_id "xxx" not found |
| 3 | genre_id "xxx" not found |
| 4 | group_id "xxx" not found |
| 105 | Sorry, service currently unavailable |
| 111 | format "xxx" not found |
| 112 | Method "xxx" not found |
| 113 | detail "xxx" not found |
| 150 | Required argument "xxx" not found |
| 151 | date malformed |

| Channel bbc.channel.getInfo | |
|--|---|
| Method | method=bbc.channel.getInfo |
| Description | A method to access TV-Anytime service information for a specific service. |
| Input parameters | channel_id (e.g., BBCOne format (must be set to tvanytime) |
| Output parameters | Service Description TV-Anytime Schema. |
| Request Sample Message | ?method=bbc.channel.getInfo&channel_id=BBCOne&format=tvanytime |
| Response Sample Message | <rsp stat="ok"> <TVAMain xml:lang="en"> |

| | |
|--|--|
| | <pre> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ServiceInformationTable> <ServiceInformation serviceId="BBCOne"> <Name>BBC One</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.1044</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> </ServiceInformation> </ServiceInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |
| | |

| Channel | |
|---------------------------------|---|
| bbc.channel.getLocations | |
| Method | method=bbc.channel.getLocations |
| Description | A method to access TV-Anytime service information for a specific service |
| Input parameters | channel_id=BBCOne format=tvanytime |
| Output parameters | TV-Anytime XML response or error response |
| Request Sample Message | ?method=bbc.channel.getLocations&channel_id=BBCOne&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ServiceInformationTable> <ServiceInformation serviceId="BBCOne"> <Name>BBC One</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.1044</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> </ServiceInformation> </ServiceInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |

| Channel | |
|-------------------------|---|
| bbc.channel.list | |
| Method | method=bbc.channel.list&format=tvanytime |
| Description | To access all service information available in TV-Anytime format |
| Input parameters | Format=tvanytime |
| Output parameters | TV-Anytime Service information |
| Request Sample Message | ?method=bbc.channel.list&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <TVAMain xml:lang="en"> <ProgramDescription> -<ServiceInformationTable> <ServiceInformation serviceId="BBCOne"> <Name>BBC One</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.1044</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> </ServiceInformation> <ServiceInformation serviceId="BBCTwo"> <Name>BBC Two</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.1084</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> </ServiceInformation> <ServiceInformation serviceId="BBCNewsTF"> <Name>BBC News 24</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.113f</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> <ServiceGenre href="urn:tva:metadata:cs:ContentCS:2005:3.1.1"> <Name>News</Name> </ServiceGenre> </ServiceInformation> <ServiceInformation serviceId="BBCThree"> <Name>BBC Three</Name> <Owner>BBC</Owner> <ServiceURL>dvb://233a.1004.10ff</ServiceURL> <ServiceGenre href="urn:tva:metadata:cs:MediaTypeCS:2005:7.1.3"> <Name>Audio and video</Name> </ServiceGenre> <ServiceGenre href="urn:tva:metadata:cs:IntendedAudienceCS:2005:4.2.2.2"> <Name>age 18-24</Name> </pre> |

| | |
|--|--|
| | <pre> </ServiceGenre> <ServiceGenre href="urn:tva:metadata:cs:IntendedAudienceCS:2005:4.2.2.3"> <Name>age 25-34</Name> </ServiceGenre> </ServiceInformation> </ServiceInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |
|--|--|

| Genre bbc.genre.getInfo | |
|--|--|
| Method | method=bbc.genre.getInfo&genre_id=comedy&format=tvanytime |
| Description | Method to give genre identifiers and names. |
| Input parameters | genre_id (required) Identifier for genre format (simple) Response format |
| Output parameters | Genre definition, name and ID |
| Request Sample Message | <pre> rsp stat="ok"> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.3"> <Name>Astonishing</Name> </Genre> </rsp> </pre> |
| Response Sample Message | <pre> <rsp stat="ok"> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.3"> <Name>Astonishing</Name> </Genre> </rsp> </pre> |

| Genre bbc.genre.getMembers | |
|---|--|
| Method | method=bbc.genre.getMembers |
| Description | To access programme information for programmes with the requested genre_id |
| Input parameters | genre_id=urn%3Atva%3Ametadata%3Acs%3AAtmosphereCS%3A2005%3A8.1 format=tvanytime |
| Output parameters | XML TV-Anytime data |
| Request Sample Message | ?method=bbc.genre.getMembers&genre_id=urn%3Atva%3Ametadata%3Acs%3AAtmosphereCS%3A2005%3A8.1&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ProgramInformationTable> <ProgramInformation programId="crid://bbc.co.uk/285155432"> </pre> |

| |
|--|
| <pre> <BasicDescription> <Title>Little Britain</Title> <Synopsis length="short"> 6/6. Matt Lucas and David Walliams take a comic look at life in Britain. Lou and Andy go to the seaside and Dafydd has been to a rugby match. Some strong language. [AD,S] </Synopsis> <Genre href="urn:tva:metadata:cs:IntentionCS:2005:1.1"> <Name>ENTERTAIN</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.5.7"> <Name>Comedy</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.5"> <Name>AMUSEMENT/ENTERTAINMENT</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.5.7.4"> <Name>Satire</Name> </Genre> <Genre href="urn:tva:metadata:cs:IntentionCS:2005:1.1.1"> <Name>Pure entertainment</Name> </Genre> <Genre href="urn:tva:metadata:cs:FormatCS:2005:2.4.4"> <Name>Standup comedian(s)</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.42"> <Name>Satirical</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.32"> <Name>Irreverent</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.1"> <Name>Alternative</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.27"> <Name>Humorous</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.46"> <Name>Silly</Name> </Genre> <CaptionLanguage closed="true">EN-UK</CaptionLanguage> <RelatedMaterial> <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2002:10"> <Name>For more information</Name> </HowRelated> <MediaLocator> <mpeg7:MediaUri>http://www.bbc.co.uk/littlebritain/</mpeg7:MediaUri> </MediaLocator> </RelatedMaterial> </BasicDescription> <AVAttributes> <AudioAttributes> <NumOfChannels>2</NumOfChannels> </AudioAttributes> <AudioAttributes> <NumOfChannels>1</NumOfChannels> </pre> |
|--|

| | |
|--|---|
| | <pre> <AudioLanguage purpose="urn:tva:metadata:cs:AudioPurposeCS:2005:1">EN- UK</AudioLanguage> </AudioAttributes> <VideoAttributes> <AspectRatio>16:9</AspectRatio> </VideoAttributes> </AVAttributes> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/LittleBritain"/> </ProgramInformation> </ProgramInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |
|--|---|

| Genre bbc.genre.getList | |
|------------------------------------|--|
| Method | method=bbc.genre.list |
| Description | To access a list of genre id |
| Input parameters | active (true) Flag to specify only active genres, set false to show all available genres format (simple) Response format |
| Output parameters | List of genre HREF and names |
| Request Sample Message | ?method=bbc.genre.list&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.1"> <Name>Alternative</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.2"> <Name>Analytical</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.3"> <Name>Astonishing</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.7"> <Name>Chilling</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.10"> <Name>Confrontational</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.11"> <Name>Contemporary</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.12"> <Name>Crazy</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.13"> <Name>Cutting edge</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.14"> <Name>Eclectic</Name> </pre> |

| | |
|--|---|
| | <pre> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.16"> <Name>Exciting</Name> </Genre> </rsp> </pre> |
|--|---|

| Group bbc.group.getInfo | |
|-----------------------------------|---|
| Method | method=bbc.group.getInfo |
| Description | Access TV-Anytime Group information tables |
| Input parameters | group_id format=tvanytime |
| Output parameters | Group information tables or error codes |
| Request Sample Message | ?method=bbc.group.getInfo&group_id=crid%3A%2F%2Fbbc.co.uk%2FBBCGroups&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <GroupInformationTable> <GroupInformation groupId="crid://bbc.co.uk/BBCGroups" ordered="true"> <GroupType xsi:type="ProgramGroupType" value="otherCollection"/> <BasicDescription> <Title>BBC groups</Title> <Synopsis>All BBC groups</Synopsis> </BasicDescription> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/Groups"/> </GroupInformation> </GroupInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |

| Group bbc.group.getMembers | |
|--------------------------------------|---|
| Method | method=bbc.group.getMembers |
| Description | Method to access the content referencing table containing the group members associated with a group,. |
| Input parameters | group_id (BBC Groups) Identifier for group, leave blank to retrieve members of the 'BBC Groups' highest level group detail (simple) Level of information in response: simple, schedule format (simple) Response format |
| Output parameters | TV-Anytime Content Referencing table or error codes. |

| | |
|-------------------------|---|
| Request Sample Message | ?method=bbc.group.getMembers&group_id=crid%3A%2F%2Fbbc.co.uk%2FHighlights&format=tvanytime |
| Response Sample Message | <pre><rsp stat="ok"> <ContentReferencingTable version="1" xsi:schemaLocation="http://www.tv- anytime.org/2002/02/ContentReferencing I:/tv- anytime/schema/content_refv1.2spec(validateslocally).xsd"> <Result CRID="crid://bbc.co.uk/Highlights" status="resolved" complete="false" acquire="all"> <CRIDResult> <Crid>crid://bbc.co.uk/SpaceRace</Crid> <Crid>crid://bbc.co.uk/Spooks</Crid> <Crid>crid://bbc.co.uk/WakingtheDead</Crid> <Crid>crid://bbc.co.uk/JackDeeLiveatTheApollo</Crid> <Crid>crid://bbc.co.uk/MurderBlues</Crid> </CRIDResult> </Result> </ContentReferencingTable> </rsp></pre> |

| Group bbc.group.list | |
|---------------------------------|---|
| Method | method=bbc.group.list |
| Description | To access a list of all the available 'Programme Groups' |
| Input parameters | format=tvanytime |
| Output parameters | TV-Anytime group information table |
| Request Sample Message | ?method=bbc.group.list&format=tvanytime |
| Response Sample Message | <pre><rsp stat="ok"> <TVAMain xml:lang="en"> <ProgramDescription> <GroupInformationTable> <GroupInformation groupId="crid://bbc.co.uk/Groups" ordered="true"> <GroupType xsi:type="ProgramGroupTypeType" value="otherCollection"/> <BasicDescription> <Title>Groups</Title> <Synopsis>All groups</Synopsis> </BasicDescription> </GroupInformation> <GroupInformation groupId="crid://bbc.co.uk/BBCGroups" ordered="true"> <GroupType xsi:type="ProgramGroupTypeType" value="otherCollection"/> <BasicDescription> <Title>BBC groups</Title> <Synopsis>All BBC groups</Synopsis> </BasicDescription> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/Groups"/> </GroupInformation> <GroupInformation groupId="crid://bbc.co.uk/Highlights" ordered="true"> <GroupType xsi:type="ProgramGroupTypeType" value="otherCollection"/> <BasicDescription></pre> |

| | |
|--|--|
| | <pre> <Title>Highlights</Title> <Synopsis>What's new and interesting</Synopsis> </BasicDescription> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/BBCGroups"/> </GroupInformation> <GroupInformation groupId="crid://bbc.co.uk/DoctorWho" ordered="true"> <GroupType xsi:type="ProgramGroupTypeType" value="series"/> <BasicDescription> <Title>Doctor Who</Title> <Synopsis>Doctor Who</Synopsis> </BasicDescription> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/Favourites"/> </GroupInformation> <GroupInformation groupId="crid://bbc.co.uk/Favourites" ordered="true"> <GroupType xsi:type="ProgramGroupTypeType" value="otherCollection"/> -<BasicDescription> <Title>Favourites</Title> <Synopsis>Old and new firm favourites</Synopsis> </BasicDescription> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/BBCGroups"/> </GroupInformation> </GroupInformationTable> </ProgramDescription> </TVAMain> </rsp> </pre> |
|--|--|

| Programme bbc.programme.getInfo | |
|---|--|
| Method | method=bbc.programme.getInfo |
| Description | Provides access to bbc programme information as TV-Anytime Programme Information table |
| Input parameters | programme_id=crid%3A%2F%2Fbbc.co.uk%2F272901906 format=tvanytime |
| Output parameters | Programme Information Table, referenced by programme CRID |
| Request Sample Message | ?method=bbc.programme.getInfo&programme_id=crid%3A%2F%2Fbbc.co.uk%2F272901906&format=tvanytime |
| Response Sample Message | <pre> <rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ProgramInformationTable> <ProgramInformation programId="crid://bbc.co.uk/272901906"> <BasicDescription> <Title>EastEnders</Title> <Synopsis length="short"> Stacey and Tanya come to blows at Peggy's birthday party, forcing Max to take drastic steps. </pre> |

| | |
|--|---|
| | <p>Billy worries that he won't be able to provide for his family. Phil is pushed too far by Ian.</p> <pre> [AD,S] </Synopsis> <PromotionalInformation>q</PromotionalInformation> <Keyword type="main">soap</Keyword> <Keyword type="main">London</Keyword> <Genre href="urn:tva:metadata:cs:IntentionCS:2005:1.1"> <Name>ENTERTAIN</Name> </Genre> <Genre href="urn:tva:metadata:cs:FormatCS:2005:2.2"> <Name>REPRESENTATION/PLAY</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.4.2"> <Name>Soap</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.4.15"> <Name>Contemporary drama</Name> </Genre> <Genre href="urn:tva:metadata:cs:AtmosphereCS:2005:8.20"> <Name>Gripping</Name> </Genre> <CaptionLanguage closed="true">EN-UK</CaptionLanguage> <RelatedMaterial> <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2002:6"> <Name>Recommendation</Name> </HowRelated> <MediaLocator> <mpeg7:MediaUri>crid://bbc.co.uk/285189256</mpeg7:MediaUri> </MediaLocator> </RelatedMaterial> <RelatedMaterial> <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2002:10"> <Name>For more information</Name> </HowRelated> <MediaLocator> <mpeg7:MediaUri>http://www.bbc.co.uk/eastenders</mpeg7:MediaUri> </MediaLocator> </RelatedMaterial> </BasicDescription> <AVAttributes> <AudioAttributes> <NumOfChannels>2</NumOfChannels> </AudioAttributes> <AudioAttributes> <NumOfChannels>1</NumOfChannels> <AudioLanguage purpose="urn:tva:metadata:cs:AudioPurposeCS:2005:1">EN- UK</AudioLanguage> </AudioAttributes> <VideoAttributes> <AspectRatio>16:9</AspectRatio> </VideoAttributes> </AVAttributes> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/EastEnders"/> </ProgramInformation> </ProgramInformationTable> </pre> |
|--|---|

| | |
|--|--|
| | <pre></ProgramDescription> </TVAMain> </rsp></pre> |
|--|--|

| Programme bbc.programme.getLocations | |
|--|---|
| Method | method=bbc.programme.getLocations |
| Description | Accesses the TV-Anytime content referencing table for a programme identified through the CRID. |
| Input parameters | <pre>programme_id (required) Programme identifier type file, ip-stream, dvb codec Preferred type, e.g. WMV, MPEG-2, FLV, QT etc. format (simple) Response format</pre> |
| Output parameters | TV-Anytime content referencing table for the programme, containing locations of the programme. |
| Request Sample Message | ?method=bbc.programme.getLocations&programme_id=crid%3A%2F%2Fbbc.co.uk%2F272901906&format=tvanytime |
| Response Sample Message | <pre><rsp stat="ok"> <ContentReferencingTable version="1" xsi:schemaLocation="http://www.tv- anytime.org/2002/02/ContentReferencing I:/tv- anytime/schema/content_refv1.2spec(validateslocally).xsd"> <Result CRID="crid://bbc.co.uk/272901906" status="resolved" complete="true" acquire="any"> <LocationsResult> <Locator instanceMetadataId="imi:bbc.co.uk/272901906"> dvb://233a.1004.1044;2712@2007-03-23T19:59:15Z/PT00H30M </Locator> <Locator instanceMetadataId="imi:bbc.co.uk/285157046"> dvb://233a.1004.10ff;26b6@2007-03-23T22:03:00Z/PT00H30M </Locator> </LocationsResult> </Result> </ContentReferencingTable> </rsp></pre> |

| Programme bbc.programme.search | |
|--|---|
| Method | method=bbc.programme.search |
| Description | Access Programme Information Table, Programme Location Table through search on a number of parameters. |
| Input parameters | <pre>title Title (or part thereof, minimum 3 characters) start (xxxx-xx-xxT00:00:00+00:00) Start time of window in which to retrieve programmes, defaults to now end (xxxx-xx-xx(+1)T00:00:00+00:00) End time of window in which to retrieve programmes, defaults to 23:59:59 today</pre> |

| | |
|-------------------------|--|
| | <p>keyword Free-text search term (minimum 3 characters)</p> <p>limit (100) Maximum number of programmes to return. Maximum value is 500</p> <p>detail (simple) Level of information in response: simple, schedule</p> <p>sort (title) The order in which the programmes are returned. Possible values are: title (default) - alphabetical by title title-reverse - reverse alphabetical by title</p> <p>format (simple) Response format</p> |
| Output parameters | Programme Information Table, Programme Location Table |
| Request Sample Message | ?method=bbc.programme.search&title=eastenders&limit=100&keyword=soap&start=2007-03-23T12:13:00Z&end=2007-03-23T23:59:59Z&format=tvanytime&detail=schedule |
| Response Sample Message | <pre><rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ProgramInformationTable> <ProgramInformation programId="crid://bbc.co.uk/272901906"> <BasicDescription> <Title>EastEnders</Title> <Synopsis length="short"> Stacey and Tanya come to blows at Peggy's birthday party, forcing Max to take drastic steps. Billy worries that he won't be able to provide for his family. Phil is pushed too far by Ian. [AD,S] </Synopsis> <PromotionalInformation>q</PromotionalInformation> <Keyword type="main">soap</Keyword> <Keyword type="main">London</Keyword> <Genre href="urn:tva:metadata:cs:IntentionCS:2005:1.1"> <Name>ENTERTAIN</Name> </Genre> <Genre href="urn:tva:metadata:cs:FormatCS:2005:2.2"> <Name>REPRESENTATION/PLAY</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.4.1"> <Name>General light drama</Name> </Genre> <Genre href="urn:tva:metadata:cs:FormatCS:2005:2.2.1"> <Name>Fictional portrayal of life</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.4.2.1"> <Name>Soap opera</Name> </Genre> <CaptionLanguage closed="true">EN-UK</CaptionLanguage> <RelatedMaterial> <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2002:6"></pre> |

| |
|--|
| <pre> <Name>Recommendation</Name> </HowRelated> <MediaLocator> <mpeg7:MediaUri>crid://bbc.co.uk/285189256</mpeg7:MediaUri> </MediaLocator> </RelatedMaterial> <RelatedMaterial> <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2005:10"> <Name>For more information</Name> </HowRelated> <MediaLocator> <mpeg7:MediaUri>http://www.bbc.co.uk/eastenders/</mpeg7:MediaUri> </MediaLocator> </RelatedMaterial> </BasicDescription> <AVAttributes> <AudioAttributes> <NumOfChannels>2</NumOfChannels> </AudioAttributes> <AudioAttributes> <NumOfChannels>1</NumOfChannels> <AudioLanguage purpose="urn:tva:metadata:cs:AudioPurposeCS:2005:1">EN- UK</AudioLanguage> </AudioAttributes> <VideoAttributes> <AspectRatio>16:9</AspectRatio> </VideoAttributes> </AVAttributes> <MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/EastEnders"/> </ProgramInformation> </ProgramInformationTable> <ProgramLocationTable> <Schedule serviceIDRef="BBCOne"> <ScheduleEvent> <Program crid="crid://bbc.co.uk/272901906"/> <ProgramURL> dvb://233a.1004.1044;2712@2007-03-23T19:59:15Z/PT00H30M </ProgramURL> <InstanceMetadataId>imi:bbc.co.uk/272901906</InstanceMetadataId> <PublishedStartTime>2007-03-23T20:00:00Z</PublishedStartTime> <PublishedDuration>PT00H30M00S</PublishedDuration> </ScheduleEvent> </Schedule> <Schedule serviceIDRef="BBCThree"> <ScheduleEvent> <Program crid="crid://bbc.co.uk/272901906"/> <ProgramURL> dvb://233a.1004.10ff;26b6@2007-03-23T22:03:00Z/PT00H30M </ProgramURL> <InstanceMetadataId>imi:bbc.co.uk/285157046</InstanceMetadataId> <PublishedStartTime>2007-03-23T22:00:00Z</PublishedStartTime> <PublishedDuration>PT00H30M00S</PublishedDuration> </ScheduleEvent> </Schedule> </ProgramLocationTable> </pre> |
|--|

| | |
|--|--|
| | <pre></ProgramDescription> </TVAMain> </rsp></pre> |
|--|--|

| Schedule bbc.schedule.getProgrammes | |
|---|--|
| Method | method=bbc.schedule.getProgrammes |
| Description | Access to Programme Information Tables and Programme Location Tables corresponding to a service schedule over the requested time window |
| Input parameters | <p>start (xxxx-xx-xxTxx:xx:00Z) GMT Start time of window in which to retrieve programmes, defaults to current time and date</p> <p>end (xxxx-xx-xxT23:59:00Z) GMT End time of window in which to retrieve programmes, defaults to 23:59:59 today</p> <p>channel_id Comma-separated list of channels</p> <p>limit (100) Maximum number of programmes to return per channel (e.g. use '2' for 'now and next'), maximum value is 500</p> <p>detail (simple) Level of information in response: simple, schedule</p> <p>format (simple) Response format</p> |
| Output parameters | |
| Request Sample Message | ?method=bbc.schedule.getProgrammes&channel_id=BBCOne,BCCBBC&start=2007-03-23T12:52:00Z&end=2007-03-23T23:59:59Z&limit=100&format=tvanytime |
| Response Sample Message | <pre><rsp stat="ok"> <TVAMain xml:lang="en"> <!-- This data is strictly for non-commercial use only. See http://backstage.bbc.co.uk/archives/2005/01/terms_of_use.html for details of the terms and conditions. --> <ProgramDescription> <ProgramInformationTable> <ProgramInformation programId="crid://bbc.co.uk/272901828"> <BasicDescription> <Title>A Question of Sport</Title> <Synopsis length="short"> Sports quiz hosted by Sue Barker and regular captains Ally McCoist and Matt Dawson, with guests Kelly Sotherton, Ken Doherty, Junior Witter and Graeme Le Saux. [S] </Synopsis> <Keyword type="main">quiz</Keyword> <Keyword type="main">sport</Keyword> <Genre href="urn:tva:metadata:cs:IntentionCS:2005:1.2"> <Name>INFORM</Name> </Genre> <Genre href="urn:tva:metadata:cs:FormatCS:2005:2.1.5"> <Name>Discussion/Interview/Debate/Talkshow</Name> </Genre> <Genre href="urn:tva:metadata:cs:ContentCS:2005:3.2"> <Name>SPORTS</Name></pre> |


```

</Genre>
<CaptionLanguage closed="true">EN-UK</CaptionLanguage>
<RelatedMaterial>
<HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2005:14">
<Name>Support</Name>
</HowRelated>
<MediaLocator>
<mpeg7:MediaUri>mailto:question.of.sport@bbc.co.uk</mpeg7:MediaUri>
</MediaLocator>
</BasicDescription>
<AVAttributes>
<AudioAttributes>
<NumOfChannels>2</NumOfChannels>
</AudioAttributes>
<VideoAttributes>
<AspectRatio>16:9</AspectRatio>
</VideoAttributes>
</AVAttributes>
<MemberOf xsi:type="MemberOfType" crid="crid://bbc.co.uk/__SERAQuestionofSport"/>
</ProgramInformation>
..
</ProgramInformation>
</ProgramInformationTable>
<ProgramLocationTable>
<Schedule serviceIDRef="BCCBBC">
<ScheduleEvent>
<Program crid="crid://bbc.co.uk/306128763"/>
<ProgramURL>
dvb://233a.1004.123f;277b@2007-03-23T12:30:40Z/PT00H27M
</ProgramURL>
<PublishedStartTime>2007-03-23T12:30:00Z</PublishedStartTime>
<PublishedDuration>PT00H30M00S</PublishedDuration>
</ScheduleEvent>
<ScheduleEvent>
<Program crid="crid://bbc.co.uk/306128759"/>
<ProgramURL>
dvb://233a.1004.123f;2777@2007-03-23T12:59:32Z/PT00H19M
</ProgramURL>
<PublishedStartTime>2007-03-23T13:00:00Z</PublishedStartTime>
<PublishedDuration>PT00H20M00S</PublishedDuration>
</ScheduleEvent>
<ScheduleEvent>
<Program crid="crid://bbc.co.uk/306128757"/>
<ProgramURL>
dvb://233a.1004.123f;2775@2007-03-23T13:20:26Z/PT00H08M
</ProgramURL>
<PublishedStartTime>2007-03-23T13:20:00Z</PublishedStartTime>
<PublishedDuration>PT00H10M00S</PublishedDuration>
</ScheduleEvent>
...
</Schedule>
</ProgramLocationTable>
</ProgramDescription>
</TVAMain>

```

</rsp>

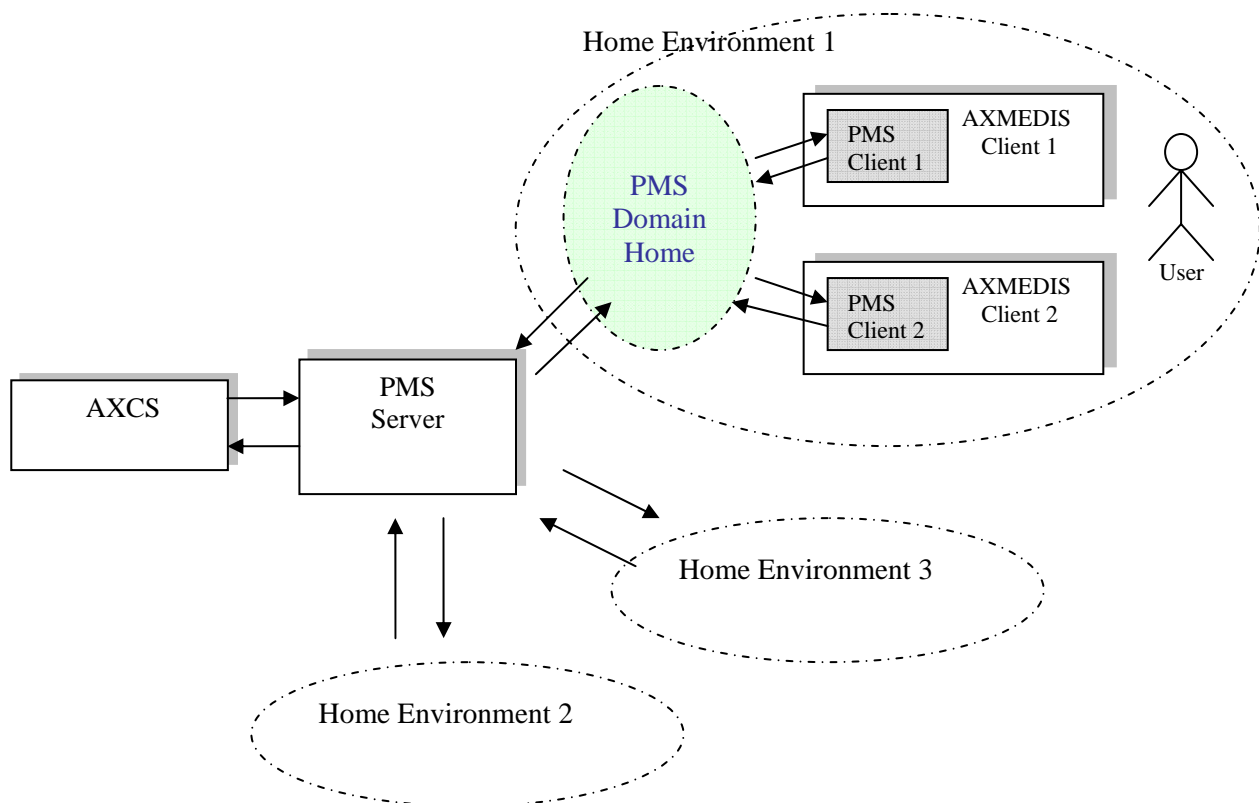
3.4 Home domain environment (BBC, PKU)

It is our intention to use the AXFW for this functionality. It will depend on work undertaken by other workpackages.

Note: The current implementation of Domains does not include the registration and management of devices, as it was not specified nor requested in the original specifications of the AXMEDIS project. This will be developed for the 4HOME demonstrator.

According to the top-level demonstrator diagram, there are will be 5 home domain environments or more. In each of the home environment, there can be as many devices as possible. All the home environments can access content via both the DVB-T distribution and the P2P distribution at the B2C level.

The Diagram below, based on “DE 12.1.2.1 AXMEDIS-4HOME Use Case and Requirement Analysis” shows the overall architecture schematic for the Home domain management.



In the figure above, an end user in each home environment, as the domain administrator, is responsible of managing the corresponding domain, including creation and deletion of the domain., and any member in the home can add or delete a device to the domain. In each of home domain environment, the PMS Domain Home Web service must be installed on any of the device of the domain. On the content factory side, the

PMS Server should be installed to provide the full set of DRM protection functionalities and plays the role of the interface to the AXCS,

3.4.1 Module or Tools for Establishment of Home Domain

- 1) PMS Domain Home:
Provision of the domain management needed for a set of PMS Clients in a single home environment. The Domain functionality inside The PMS Domain Home was designed only for allowing content consumption (not creation) inside a Domain that it should be installed on a set-top box or a PC with DVB-T card in the home environment.

The functionalities of device registration should be implemented in this module.

- 2) PMS Client:
This module provides protection functionality to the AXMEDIS users inside AXMEDIS clients, including players, editors, etc.. It can work in offline and taking decisions about what can be done or not, and can also work as an interface to PMS Server. In both cases PMS Client uses different modules to perform decisions or just to store useful information that will be needed in offline scenario. It currently provides the functionalities for license and PAR generation, authorisation support and is the interface to the PMS Server and AXCS. The PMS Client should be installed in PCs with AXMEDIS client tools

- 3) PMS Server:
This module provides protection functionality to the AXMEDIS users inside the AXMEDIS system. It is a front end for the different modules inside it. It provides the functionalities corresponding to the license manager, license generator, authorisation support modules and is the interface between PMS Client and AXCV. The PMS server provides the whole functionality regarding security and licensing at a global level. The PMS server should be installed on the content factory side.

- 4) Domain Manager (console application)
The Domain Manager Application is a console application that only manages domains. With this application users can create, delete or rename domains.

The functionalities of device registration should be implemented in this module.

Additional work to these AXMEDIS tools to enable the addition and removal of devices will be done in conjunction with the workpackages developing these tools.

3.4.2 General Description of the Execution Process

Establishment of home domain

The domain administrator creates a domain through the Domain Manager, a console application installed on the same device as the PMS Domain Home. And the PMS Domain Home Web service could be installed on the same device as the Domain Manager or on a different one. In any case, the Domain Manager and the PMS Domain Home should share the same domain database.

As part of the development work yet to be done, the PMS server could be deployed to play the role of allocating domain ID or domain key at the global level.

Management of home domain

1) When the domain administrator needs to add a device to an established domain, the PMS client on the device requests to the PMS Home for authentication. After the authentication process, a certificate containing the domain key, issued by the PMS Domain Home, is returned to the requesting device

This functionality is expected to be implemented.

2) When end user deletes a device from the domain, the PMS Domain Home deletes the information of the device.

The PMS Domain Home needs to refresh to the domain key to keep the domain still protected. Else, the PMS client installed on the leaving device can delete the domain key safely. This is to be resolved during implementation.

3) Any member wishing to join the domain should firstly register to the Domain Home from the device that is ready registered with the Domain Home.

4) After authentication process, the reference of the user to the domain is created by the PMS client on the device, and this information should be shared by the domain database. After registration, the user can play content with any device in the domain.

The device, from which the user registered with the Domain Home, should have been registered with the Domain Home in the first instance.

Note, in the process described above, All of the modules or tools should have a certificate that should be issued by the AXMEDIS CA.

Content usage in the home domain

1) For requesting content with a domain license, the user should send the user ID and the domain ID, the PMS Domain Home and the PMS server verify the information accordingly.

2) Any user verified as registered in the domain can have access to any content with a license for the domain, unless the user was blocked for some other reason, and this situation is detected on the AXCS during the verification phase.

Content usage between the home domains

Content redistribution between home domains is strictly not allowed, Only the reference to the Content Factory can be transferred between domains..

3.4.3 Module Design in terms of Classes

Refer to “DE3.1.2.2.14 Specification of AXMEDIS Protection Support”

New functionalities for device registration to be added into the existing module of “Domain Home”:

| DoDeviceRegistration | |
|-----------------------------|--|
| Method | DoDeviceRegistration |
| Description | Domain manager requests device registration |
| Input parameters | std::string DEVICE_ID: Device identifier std::string AXUID: User identifier std::string AXDOM: Domain identifier |
| Output parameters | std::int result |

| DoDeviceUnregistration | |
|-------------------------------|------------------------|
| Method | DoDeviceUnregistration |

| | |
|-------------------|--|
| Description | Domain manager requests device unregistration |
| Input parameters | std::string DEVICE_ID: Device identifier std::string AXUID: User identifier std::string AXDOM: Domain identifier |
| Output parameters | std::int result |

| isDeviceAlreadyRegistered | |
|----------------------------------|--|
| Method | isDeviceAlreadyRegistered |
| Description | Check if device is already registered in the domain |
| Input parameters | std::string DEVICE_ID: Device identifier std::string AXDOM: Domain identifier |
| Output parameters | std::int result |

| RetrieveRegisteredDevices | |
|----------------------------------|---|
| Method | retrieveRegisteredDevices |
| Description | Retrieve the list of registered devices in a domain |
| Input parameters | None |
| Output parameters | std::vector<std::string> result |

3.4.4 User Interface Description

AXMEDIS provides the console application of Domain Manager.

3.5 Tool Management for the protection of recorded content in the Broadcast Environment (ETRI)

The main goal of this functionality is to provide DRM Protection Tool to the AXMEDIS client.

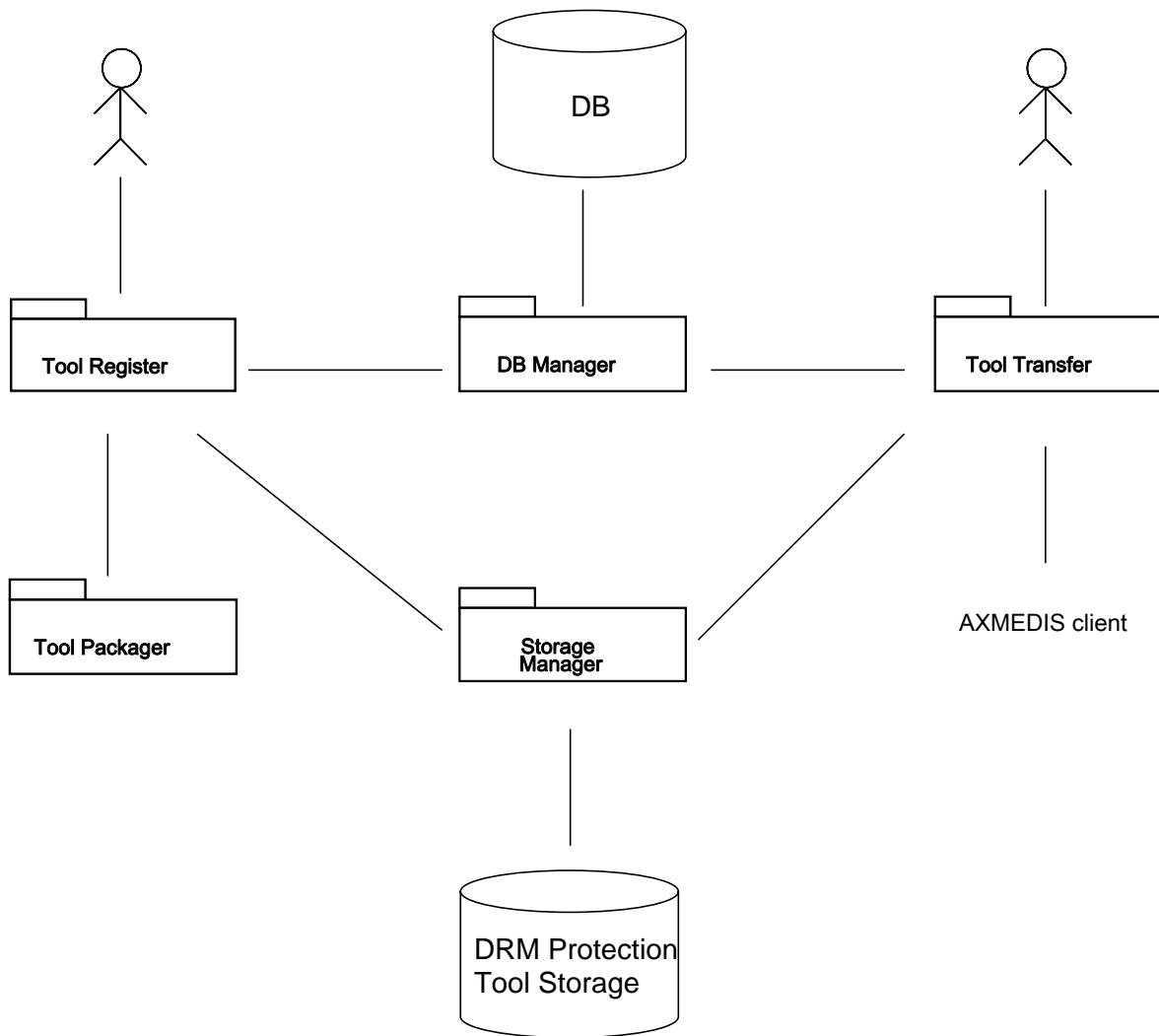
3.5.1 General Description of Tool Management

As the digital contents market becomes larger, the characteristic problem of illegal copying of the digital content is raising its head. It originates from ease copy and identical quality. There are DRM techniques being developed for copyright protection from above environment. However the DRM techniques at the present time are available only on exclusive terminals, causing problems with contents abridgment and reducing benefit to the consuming market, therefore bringing about the general vicious circle of contents market.

To solve this problem, we define the DRM protection tool and its message interface.

Tool Management System consists of DRM Protection Tool Server and AXMEDIS Client. DRM Protection Tool Server packages the DRM Tool and enables an AXMEDIS Client or a Broadcast client to download it. The DRM Protection Tool Server issues DRM Protection Tool ID in order for the AXMEDIS Client to distinguish the DRM Protection Tool. AXMEDIS Client can register or download DRM Protection Tools through the message protocol provided by the Protection Tool Server.

In the figure below, it shows the part of DRM Protection Tool Server and AXMEDIS Client, which are registering and downloading of DRM Protection Tool.



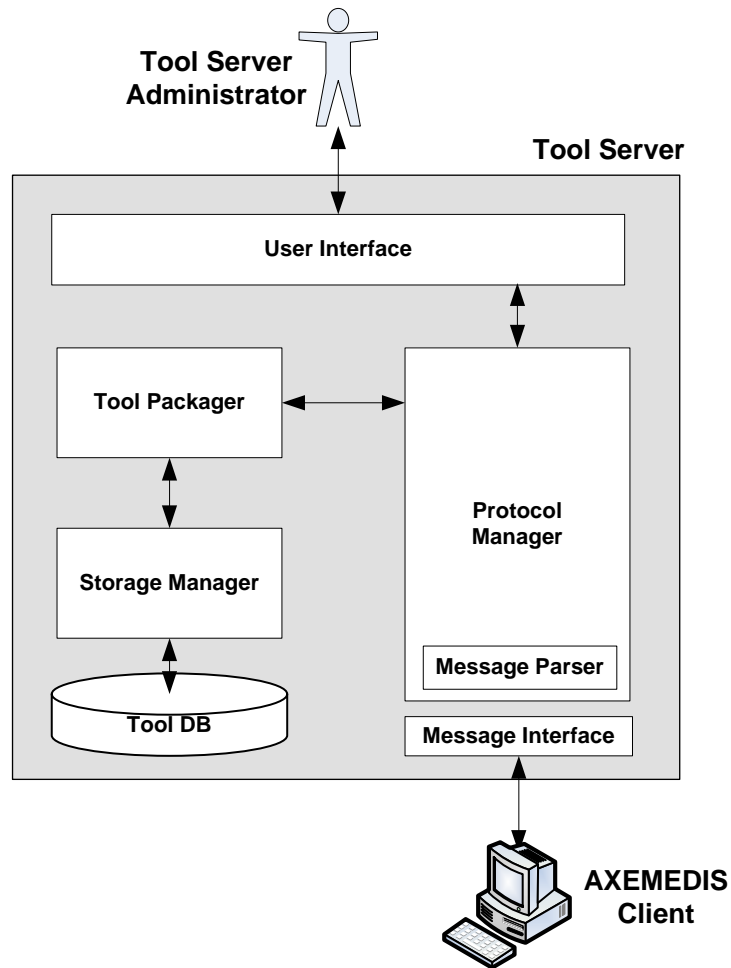
3.5.2 System architecture

3.5.2.1 DRM Protection Tool Server (simplified Tool Server)

DRM Protection Tool Server consists of Tool DB, Storage Manager, Tool Packager and Protocol Manager. The role of each module is described below.

- Tool DB: is storage module for tool and tool related information. Tool related information includes tool ID, tool meta information and tool binary data and so on.
- Storage Manager: is module for management of Tool DB as a middleware with Tool Packer.
- Tool Packager: is module for generation of tool message using tool related information which is stored from Tool DB.
- Protocol Manager: is module for management of End-User Device or tool related protocol with user. The tool related protocol includes Tool Register and Tool Download.

The architecture of the DRM Protection Tool Server is described below.

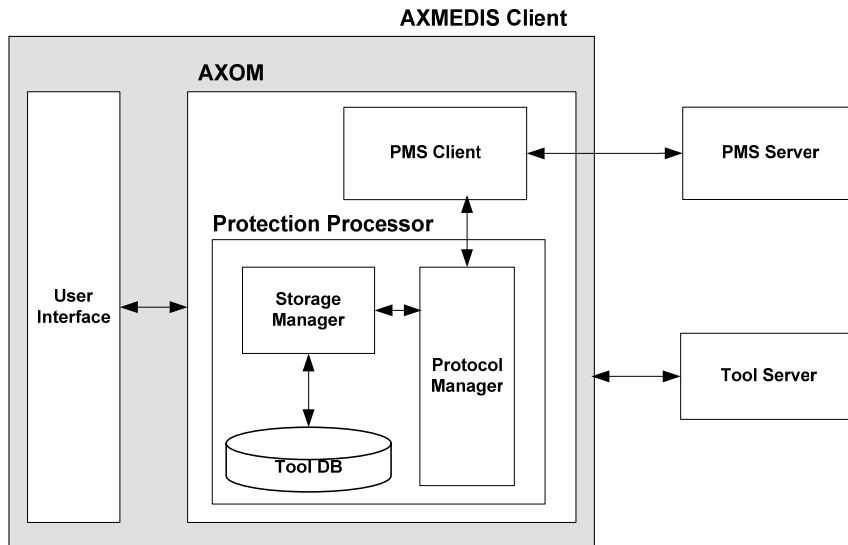


3.5.2.2 Protection Processor in AXMEDIS Client

AXMEDIS Client is composed AXOM, PMS Client and Protection Processor. And Protection Processor consists of Tool DB, Storage Manager and Protocol Manager.

The role of each module in the Protection Processor is described below.

- **Tool DB:** is storage module for tool and tool related information. Tool related information includes tool ID, tool meta information and tool binary data and so on.
- **Storage Manager:** is module for management of Tool DB as a middleware with Tool Packer.
- **Protocol Manager:** is module for management of End-User Device or tool related protocol with user. The tool related protocol includes Tool Query and Tool Download.



3.5.3 Tool management protocol

3.5.3.1 Registering

User registers a DRM Protection Tool to through the provided GUI.

1. DRM Protection Tool Server:
 - a. User logs in with ID and PW
 - b. Click “Register DRM Protection Tool” button in the menu
 - c. DRM Protection Tool Server displays form interface to receive detail information of the DRM Protection Tool.
 - d. User provides information of the DRM Protection Tool. This information includes following items:
 - Tool Name: This indicates tool name assigned by a user [optional]
 - Developer: This indicates the developer of the tool [optional]
 - Date of development: This indicates date of development [optional]
 - Reference URL: Home page of the developer [optional]
 - Functionality Description: brief information indicating what this tool does [optional]
 - Target Device Information: This indicates target platform that the DRM Protection Tool may operate on. Device Information includes type of operating system, memory, CPU, etc [mandatory]
 - Package Type: This indicates the type of package of binary tool body (Zip, TAR, etc...) [mandatory]
 - Binary Tool Agent: This includes a master tool taking care of message interface with Protection Processor in the AXMEDIS client [mandatory]
 - Binary Tool Group: This includes a library for encryption or watermarking. [optional]
 - e. User clicks “OK” button
 - f. DRM Protection Tool Server validates DRM Protection Tool information. If there is error, display warning message
 - g. DRM Protection Tool Server display the result of registration
 - h. User confirms it

3.5.3.2 Downloading

Downloading DRM Protection Tool (by user)

User may download DRM Protection Tool from the DRM Protection Tool Server.

- a. User logs in with ID and PW
- b. Click “Download DRM Tool” button in the menu
- c. DRM Tool Server displays the list of registered DRM Tools
 User may sort the list by several key factor
- d. User selects one and click “download” button
- e. DRM Tool will be downloaded to the user’s PC

Downloading DRM Protection Tool (by AXMEDIS client)

The protocol between DRM Protection Tool Server and AXMEDIS client is for downloading DRM Protection Tool.

1. AXMEDIS client:
 - a. Mutually Authenticates with DRM Protection Tool Server. The URL of the DRM Protection Tool Server is contained in the DRM information
 - b. sends a GetTools message to the DRM Protection Tool Server, containing:
 - The requested DRM Protection Tool IDs
 - Message signature
2. DRM Protection Tool Server:
 - a. Receives the GetTools message
 - b. Verifies the signature
 - c. Prepares DRM Protection Tools and packages them.
 - d. Delivers GetToolsResponse message to the AXMEDIS client
3. AXMEDIS client:
 - a. Receives the GetToolsResponse message;
 - b. Sends Ack message to DRM Protection Tool Server to notify the receipt (or otherwise) of the DRM Protection Tools
 - c. Extract Tool Information and Tool binary

As described above, AXMEDIS Client and Tool Server exchange XML messages to request and response tools. Followings are the structure and the description of GetTools and GetToolsResponse messages.

GetTools

| | |
|--------------------|--|
| Diagram | |
| Description | This message is sent by AXMEDIS Client to Tool Server to request necessary tools. This message conveys the list of the requested tools. Receiving this message, Tool Server sends response back to AXMEDIS Client with GetToolsResponse message. |
| Source | <pre><xs:element name="GetTools"> <xs:complexType> <xs:sequence> <xs:element ref="ipmpinfo:IPMPToolID" maxOccurs="unbounded" /> </xs:sequence> </xs:complexType> </xs:element></pre> |

| | |
|--|---|
| | <pre> </xs:sequence> </xs:complexType> </xs:element> </pre> |
|--|---|

GetToolsResponse

| | |
|--------------------|--|
| Diagram | |
| Description | <p>This message is for embodiment of raw DRM Protection Tools into structured and self-explained DRM Protection Tools. When the Tool Server receives the tool request messages from the AXMEDIS Client, the Tool Server create a Tool package with Tool XML Profiles and Tool binaries for transmission. Below schema structure shows packaged DRM Protection Tools, which includes XML profile and Binaries (DLL) of the tools.</p> |
| Source | <pre> <xs:element name="GetToolsResponse"> <xs:complexType> <xs:sequence> <xs:element ref="pp:ToolPackage" maxOccurs="unbounded" /> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="ToolPackage"> <xs:complexType> <xs:sequence> <xs:element ref="pp:Tool" /> <xs:element name="ToolBinary" type="xs:Base64Binary" /> </xs:sequence> </xs:complexType> </xs:element> </pre> |

3.5.3.3 Playing

1. Player:
 - a. Initializes AXOM
 - b. Sends Tool ID list to Protection Processor through PMS Client
2. Protection Processor
 - a. Searches Tools specified in the Tool ID list
 - A. If exist, verifies the Tool.
 - B. If not exist, downloads and verifies the Tool

Every time the user tries to play the content.

3. AXOM
 - a. Requests authorization to PMS Client
 - b. Loads Tools
 - c. Plays content

3.5.4 User interface description

- Description of the User Interface, if any:
 - Visual Shape and design of the main frame
 - Menu with major and minor items and related associated functionalities.
 - Contextual menu
 - Main functionalities provided from the user interface
 - Visual Shape and design of the major dialog boxes.

- Usage of tool bars, scrollbars, and any gadget or widget, etc.
- Description of main activities of the users in terms of Use Cases, see the other deliverable
- For usability aspects please consult ACIT partner

3.5.5 Technical and Installation information

- Installation capability, it has to be installable in a very easy manner
- Manual support for technical and user point of views

| | |
|---|---|
| References to other major components needed | |
| Problems not solved | • |
| Configuration and execution context | |

3.5.6 Module or Executable Tool Profile

| Tool Profile | | |
|---|--|---|
| DRM Protection Tool Server | | |
| Responsible Name | Sangwoo Ahn | |
| Responsible Partner | ETRI | |
| Status (proposed/approved) | Proposed | |
| Implemented/not implemented | Not implemented | |
| Status of the implementation | Functional description | |
| Executable or Library/module (Support) | Executable | |
| Single Thread or Multithread | Single Thread | |
| Language of Development | Asp for web UI. MS-SQL for DB. C++ for Packager | |
| Platforms supported | Microsoft Windows | |
| Interfaces API with other tools, named as | Name of the communicating tools References to other major components needed | Communication model and format (protected or not, etc.) |
| User Authentication | AXCS | |
| DRM Protection Certificate | Tool AXCS | |
| DRM Protection Downloading | Tool AXMEDIS client | |
| Formats Used | Shared with | format name or reference to a section |
| DRM Protection Packaging Format | Tool | |
| Protocol Used | Shared with | Protocol name or reference to a section |
| Used Database name | | |
| DRM Protection metadata DB | Tool | |
| User Interface | Development model, language, etc. | Library used for the development, platform, etc. |
| Log in form | ASP | |

| | | |
|---------------------|---------------------------------|--|
| DRM Protection Tool | ASP | |
| Registering form | | |
| DRM Protection Tool | ASP | |
| Downloading form | | |
| | | |
| Used Libraries | Name of the library and version | License status: GPL. LGPL. PEK, proprietary, authorized or not |
| | | |

Related AXMEDIS Tool/Module:

AXCS: This tool is capable of issuing certificate for the Tool, Device, User, etc. Therefore AXCS can be used to issue a certificate for the DRM Protection Tool.

Protection Processor: This module is to support operation of DRM Protection Tool in the AXMEDIS client.

3.6 Specification of the AXMEDIS IP Ontology for IP Entity representation – including associated demonstration components. (sDae)

3.6.1 IP Ontology

The reference and normative model for the Intellectual Property (IP) Objects Value Chain is expressed as a machine readable Ontology. It includes all the classes (kinds of entities), their main attributes and their relations in a compact, technology neutral and self contained expression.

OWL (Web Ontology Language) has been chosen to express the Ontology, following the state of the art trends and conformance with the W3C standards. For those new to OWL, more information can be found at the following reference <http://www.w3.org/TR/owl-features/>.

3.6.2 IP Ontology deployment

Although there are several approaches (methodologies) for developing Ontologies, we propose the simplest permitting incremental development with milestones for checking consistency and to obtain feedback to the Ontology development process. The milestones are as follows:

- **Formalise the DMP Creation Model (CM):** Representation of the basic structure of IP Entity Relationships as well as the initial steps in the value chain.
- **Fine tuning of the Creation Model:** As the CM is this the basis of the paradigm, its formalisation should be simple, consistent and well tested, thus at the end of this second milestone, a solid API should be available.
- **Incorporation of REL rights:** Since the AXMEDIS framework will be providing REL based licenses the set of existing standard REL terms should be included in the model.
- **Development a Distribution model:** Incorporation of concepts for B2B and B2C distribution.

The Ontology will match Axmedis concepts as tightly as possible. For example, it should be able to describe the following entities: Axmedis Object Creators, Axmedis Object Distributors, Axmedis Final Users, Axmedis B2B Users, Collecting Societies, Tool Producers etc.

3.6.3 API

The Ontology is the only valid description of the model, but in order to have an easier access to the semantics contained in it, a high-level API needs to be provided. Given that Java is the *de facto* standard when programming ontology applications it has been chosen as the language for programming the API. See section 3.6.4 for more details

3.6.4 Module or Executable Tool for AxIPOntologyAPI

| Module/Tool Profile | | |
|--|--|---|
| AxIPOntologyAPI | | |
| Responsible Name | Marc Gauvin | |
| Responsible Partner | sDAE | |
| Status (proposed/approved) | | |
| Implemented/not implemented | Not implemented | |
| Status of the implementation | Initial | |
| Executable or Library/module (Support) | AxIPOntologyAPI.jar | |
| Single Thread or Multithread | Single thread | |
| Language of Development | Java | |
| Platforms supported | Any with a Java Platform | |
| Reference to the AXFW location of the source code demonstrator | https://cvs.axmedis.org/repos/ Not yet decided | |
| Reference to the AXFW location of the demonstrator executable tool for internal download | https://cvs. Not yet available | |
| Reference to the AXFW location of the demonstrator executable tool for public download | Not yet available | |
| Address for accessing to WebServices if any, add accession information (user and Passwd) if any | Does not apply | |
| Test cases (present/absent) | Absent | |
| Test cases location | http:// not yet | |
| Usage of the AXMEDIS configuration manager (yes/no) | No | |
| Usage of the AXMEDIS Error Manager (yes/no) | No | |
| Major Problems not solved | -- -- | |
| Major pending requirements | -- -- | |
| | | |
| Interfaces API with other tools, named as | Name of the communicating tools References to other major components needed | Communication model and format (protected or not, etc.) |
| | | |
| | | |
| | | |
| Formats Used | Shared with | format name or reference to a section |
| OWL Web Ontology Language | | |
| | | |

| | | |
|--------------------|-----------------------------------|--|
| | | |
| | | |
| Protocol Used | Shared with | Protocol name or reference to a section |
| | | |
| | | |
| | | |
| Used Database name | | |
| | | |
| | | |
| | | |
| User Interface | Development model, language, etc. | Library used for the development, platform, etc. |
| | | |
| | | |
| | | |
| Used Libraries | Name of the library and version | License status: GPL. LGPL. PEK, proprietary, authorized or not |
| Jena | Jena version 2.4 | LGPL |
| Pellet | Pellet version 1.3 | MIT |
| | | |
| | | |
| | | |

3.6.5 General Description of the Module

AxIPOntologyAPI is a Java API to access the IP Ontology. This Ontology is specified in section 3.7 of this document. The API is a set of functions that facilitate the programmatic access to the ontology.

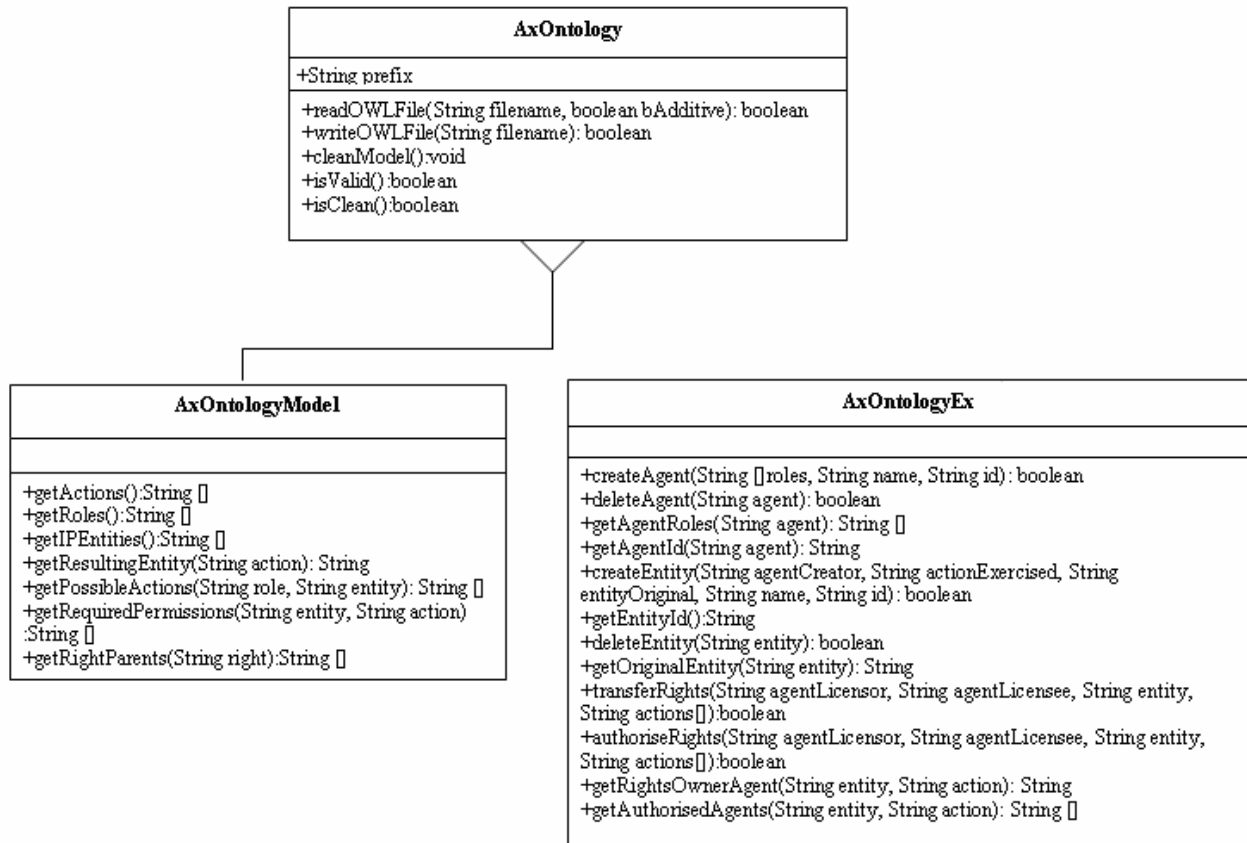
The API hides the details of access interface (Jena, Pellet), making it transparent through a set of simple function calls. Therefore, the general style of the API deals with *Strings*, rather than with *OntClass* (see Jena library) classes or similar. This is the most general form of the functions, and although the functions may receive as parameters strings that do not correspond to any class or individual, validity will always be checked.

3.6.6 Module Design in terms of Classes

AxIPOntologyAPI comes under the form of a single .jar file, as it is simple enough as to be contained in a single package. It will require however, a few other .jar files from its dependant libraries.

Three classes are defined:

- AxIPOntology: Contains the general methods for loading, storing and validating an ontology.
- AxIPOntologyModel: Contains the general methods for querying general relations.
- AxIPOntologyEx: Contains the methods to deal with individuals. It will be constructed with an AxIPOntology object.



3.6.7 Technical and Installation information

The API is based in other intermediate libraries to gain access to the OWL file. These libraries are the most widely used, although there are others that provide similar functionalities in a similar fashion:

- **Jena.** Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, SPARQL and includes a rule-based inference engine. Jena is open source.
- **Pellet.** Pellet is an open-source OWL-DL reasoner written in Java, compatible with Jena. Pellet is also open source software, available under the terms of the MIT License.

| | |
|---|--------------|
| References to other major components needed | Jena, Pellet |
| Problems not solved | • |
| Configuration and execution context | |

3.6.8 Draft User Manual

TO BE DONE

3.6.9 Examples of usage

An example of use can be in an scenario where an Axmedis Object is created making use of the AxIPontology, as described in the document DE12.1.2.2, section 3.2.1. There, Bob has created a resource, and tries to convert into an Axmedis Object. The resource is based in a previous song of Alice (who gave

Bob the appropriate rights), called “I Love You”, and receives the name “I Love You (remix)”. It will be possible to make actions like this in a couple of lines of code:

```
AxIPOntologyEx m_onto=new AxIPOntologyEx();
boolean bOk=m_onto.createEntity("Bob", "MakeAdaptation", "I Love You", "I Love You
(remix)", "id.xxxxx");
if (bOk)
    ...
```

3.6.10 General OWL classes descriptions: class AxIPOntologyModel

This set of functions, do not deal with individuals (instances) but show the general classes and their relations in the Ontology. In this set of functions, the arguments are always strings that must represent a valid class. The utility of these methods in a practical application is perhaps not apparent, but they help to clarify the concepts.

| AxIPOntologyModel::getActions | |
|--------------------------------------|---|
| Method | <code>String []getActions();</code> |
| Description | Returns the set of actions defined in the Ontology. |
| Input parameters | None |
| Output parameters | The set of actions defined in the Ontology. |

| AxIPOntologyModel::getRoles | |
|------------------------------------|---|
| Method | <code>String []getRoles();</code> |
| Description | Returns the set of roles defined in the Ontology. |
| Input parameters | None |
| Output parameters | The set of roles defined in the Ontology |

| AxIPOntologyModel::getIPEntities | |
|---|--|
| Method | <code>String []getIPEntities();</code> |
| Description | Returns the set of IPEntities defined in the Ontology. |
| Input parameters | None |
| Output parameters | The set of IPEntities defined in the Ontology. |

| AxIPOntologyModel::getResultingEntity | |
|--|---|
| Method | <code>String getResultingEntity(String action);</code> |
| Description | Returns the kind of IPEntity generated as a result of the exercise of the given <i>action</i> . |
| Input parameters | Name of a class derived from Action. |
| Output parameters | Name of a class derived from IPEntity. |

| AxIPOntologyModel::getPossibleActions | |
|--|--|
| Method | <code>String []getPossibleActions(String role, String entity);</code> |
| Description | Returns the set of actions that a <i>role</i> is able to perform over an <i>entity</i> . |
| Input parameters | Name of a class derived from IPEntity |
| Output parameters | String array with a set of names of classes derived from Role. |

| AxIPOntologyModel::getRequiredPermissions | |
|--|---|
| Method | <code>String []getRequiredPermissions(String entity, String action)</code> |
| Description | Returns the set of roles who must give rights to perform the <i>action</i> over <i>entity</i> . |

| | |
|-------------------|---|
| Input parameters | Entity Name of an entity-derived class Action Name of an Action-derived class. |
| Output parameters | Set of role classes name. |

| AxIPOntologyModel::getRightParents | |
|---|--|
| Method | <code>String []getRightParents(String right)</code> |
| Description | If rights are to be classified hierarchically, this functions returns the set of parents of the given <i>right</i> . |
| Input parameters | Right Name of the Action whose ancestor is required to be known. |
| Output parameters | Set of ancestor rights. |

3.6.11 Instances management: class AxIPOntologyEx

The Ontology defines the classes, attributes and relations, but it is not a mere semantic description of the model. It is also able to deal with instances of the classes, (also called individuals), bringing also practical applications.

In this set of functions, the arguments are always strings that must represent a valid individual name.

| AxIPOntologyEx::createAgent | |
|------------------------------------|--|
| Method | <code>boolean createAgent(String []roles, String name, String id);</code> |
| Description | Creates an agent whose name is <i>name</i> and who plays the given <i>roles</i> . Id is an identifier. It is likely to be the ISPI (International Standard Party Information Code). Returns true on success. |
| Input parameters | Roles. Set of roles that the agent will have. Name name of the agent. Id. Id of the agent, reference for an external database. |
| Output parameters | True on success. |

| AxIPOntologyEx::deleteAgent | |
|------------------------------------|--|
| Method | <code>boolean deleteAgent(String agent);</code> |
| Description | Deletes the agent <i>agent</i> from the ontology. Returns true on success. |
| Input parameters | Agent Name of the agent. |
| Output parameters | True on success. |

| AxIPOntologyEx::getAgentRoles | |
|--------------------------------------|--|
| Method | <code>String []getAgentRoles(String agent);</code> |
| Description | Returns the set of roles that an agent plays. (An agent is an individual who plays one or more roles). |
| Input parameters | Agent Name of the agent. |
| Output parameters | Set of roles that the agent has. |

| AxIPOntologyEx::getAgentId | |
|-----------------------------------|---|
| Method | <code>String getAgentId(String agent);</code> |
| Description | Gets the agent id, like ISPI. |
| Input parameters | Agent Name of the agent |
| Output parameters | ID of the agent.. |

| AxIPOntologyEx::createEntity | |
|-------------------------------------|---|
| Method | <code>boolean createEntity(String agentCreator, String actionExercised, String entityOriginal, String name, String</code> |

| | |
|-------------------|---|
| | <code>id);</code> |
| Description | Creating an entity is the exercise of an action whose result is another entity. This function creates an entity named <i>name</i> , result of exercising action <i>actionExercised</i> over entity <i>entityOriginal</i> . It has to be studied if an entity may have at the same time two or more origins, in such a case, <i>entityOriginal</i> would obviously be an array of strings. It will be checked whether <i>agentCreator</i> has the required rights to exercise the right. Id is an identification code, like the ISWC (International Standard Musical Work Code)). Returns true on success. |
| Input parameters | AgentCreator Name of the creator agent. actionExercised Name of the exercised action entityOriginal Name of the original IPEntity name name Name of the future IPEntity id ID of the future IPEntity |
| Output parameters | True on success. |

AxIPOntologyEx::getEntityId

| | |
|-------------------|------------------------------------|
| Method | <code>String getEntityId();</code> |
| Description | Gets the entity id |
| Input parameters | None |
| Output parameters | External ID of the IPEntity |

AxIPOntologyEx::deleteEntity

| | |
|-------------------|---|
| Method | <code>boolean deleteEntity(String entity);</code> |
| Description | Deletes the entity <i>entity</i> from the ontology. |
| Input parameters | Entity Name of the entity to be deleted |
| Output parameters | True on success. |

AxIPOntologyEx::getOriginalEntity

| | |
|-------------------|--|
| Method | <code>String getOriginalEntity(String entity);</code> |
| Description | Returns the IPEntity class from which <i>entity</i> depends |
| Input parameters | Entity Name of the entity whose origin is wanted to be known |
| Output parameters | IPEntity origin. |

AxIPOntologyEx::transferRights

| | |
|-------------------|--|
| Method | <code>boolean transferRights(String agentLicensor, String agentLicensee, String entity, String actions[]);</code> |
| Description | The agent <i>agentLicensor</i> transfer to the agent <i>agentLicensee</i> the make and transfer <i>actions</i> over <i>entity</i> . After this transfer, agentLicensor can no longer exercise the rights. Returns true on success. |
| Input parameters | agentLicensor Agent who transfers the rights agentLicensee Agent who receives the rights entity Entity's name being transferred actions Set of Actions that are being transferred |
| Output parameters | True on success. |

AxIPOntologyEx::authoriseRights

| | |
|-------------|--|
| Method | <code>boolean authoriseRights(String agentLicensor, String agentLicensee, String entity, String actions[]);</code> |
| Description | Returns a set of agents who should give their authorisation for <i>action</i> to be performed over <i>entity</i> . |

AxIPOntologyEx::getRightsOwnerAgent

| | |
|--------|--|
| Method | <code>String getRightsOwnerAgent(String entity, String action);</code> |
|--------|--|

| | |
|-------------------|--|
| Description | Returns the agent who can use and transfer rights to make <i>action</i> over <i>entity</i> . |
| Input parameters | Entity Entity's name Action Actions's name |
| Output parameters | A name of an agent (or null). |

| AxIPOntologyEx::getAuthorisedAgents | |
|--|--|
| Method | <code>String []getAuthorisedAgents(String entity, String action);</code> |
| Description | Returns the set of agents who can use the right <i>action</i> over <i>entity</i> . |
| Input parameters | Entity Entity's name Action Name of the action |
| Output parameters | A set of agents or null. |

3.6.12 Ontology access.

Ontology individuals can be read from and written to an OWL file. This class also includes validation functionalities.

Validation should always be performed after loading an ontology file. OWL files with individuals may have been created from another platform not respectful to the rules, and prior to any operation, it should be checked that no inconsistencies are present.

| AxIPOntology::setModel | |
|----------------------------------|--|
| Method | <code>void setModel(String url);</code> |
| Description | Stablishes the ontology model. It has to be compatible with the originally designed.. |
| Input parameters | Url Points to an ontology definition, either a namefile or a remote URL. |
| Output parameters | None |
| AxIPOntology::readOWLFile | |
| Method | <code>boolean readOWLFile(String filename, boolean bAdditive);</code> |
| Description | Reads an OWL file with instances from the file <i>filename</i> . |
| Input parameters | Filename Name of the file If <i>bAdditive</i> is true, any previous loaded individuals will not be erased, and the read file will add their individuals to the previously existent. |
| Output parameters | True on success |

| AxIPOntology::writeOWLFile | |
|-----------------------------------|--|
| Method | <code>boolean writeOWLFile(String filename);</code> |
| Description | Stores the ontology in the file <i>filename</i> . Returns true on success. |
| Input parameters | |
| Output parameters | True on success |

| AxIPOntology::cleanModel | |
|---------------------------------|---|
| Method | <code>void cleanModel();</code> |
| Description | Deletes all the instances in the model. |
| Input parameters | None |
| Output parameters | None |

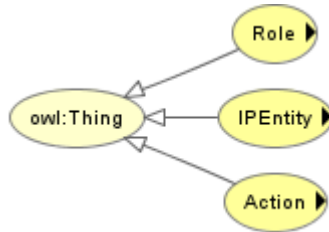
| AxIPOntology::isValid | |
|------------------------------|---|
| Method | <code>boolean isValid();</code> |
| Description | Returns true if the ontology is valid. A valid ontology is consistent and includes no major logical contradictions. |
| Input parameters | None |

| | |
|------------------------------|---|
| Output parameters | True if valid |
| AxIPOntology::isClean | |
| Method | <code>boolean isClean();</code> |
| Description | Returns true if the ontology is clean. A clean ontology is a valid ontology which throws no warnings. Warnings are issued by minor logical contradictions, i.e., if it is defined a class to whom the logic grants no possible existence of individuals. All clean ontologies are valid ontologies, but not all the valid ontologies are clean. |
| Input parameters | None |
| Output parameters | True if clean. |

4 Formal description of format AxIPOntology

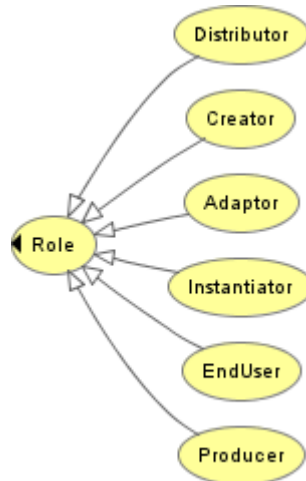
IP individuals subject to be represented in the Ontology, adhere to the AxIPOntology. This ontology is defined, in its first draft, by the OWL file shown at the end of the section.

The main classes are three (Action, IPEntity, Role), as can be seen in the next figure.

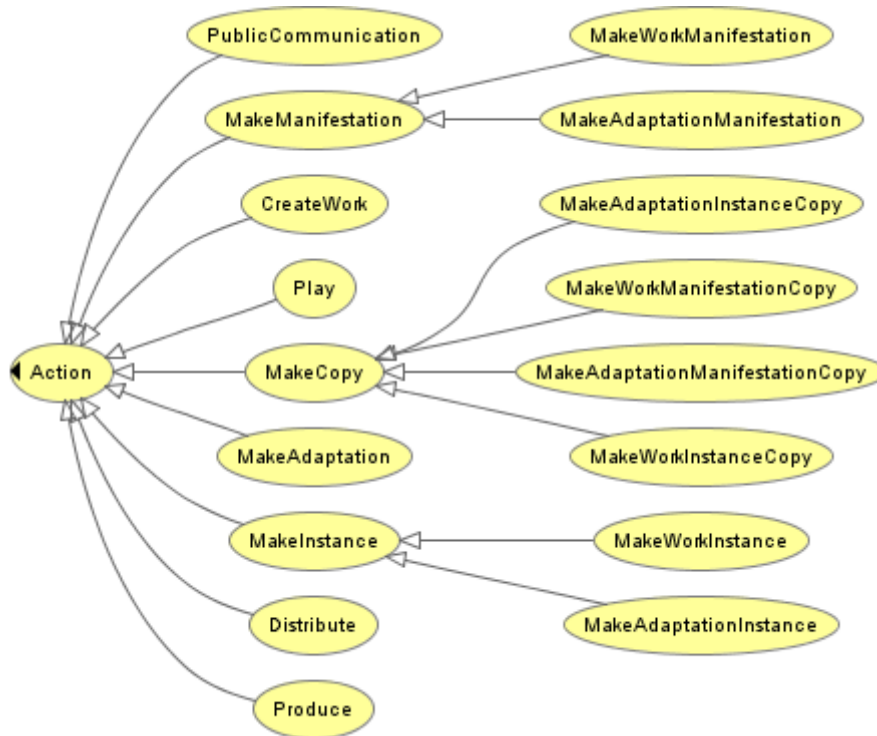


- The class Role defines a pre-determined set of actions and corresponding conditions attributed to a user. Their subclasses can be seen in the next figure:

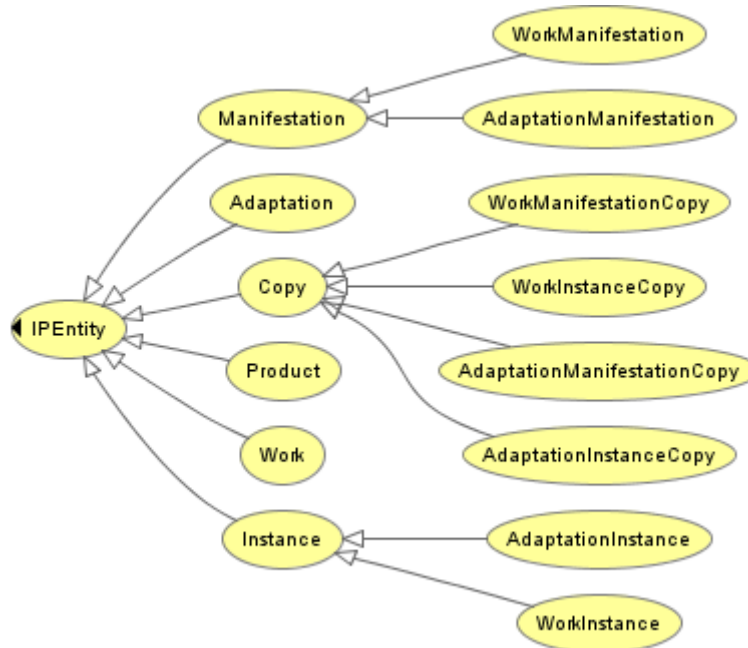
•



- The class Action represents the actions that can be done over the objects, as shown below.



- Finally, the class IPEntity, generically representing forms of different types of IP is shown along with its descendents:



The main relations are shown in the next table, here for every relation there is a domain in which it is applicable and a range of possible values. Thus, there will be classes having the relation “ResultsIn”, these classes will have their domain in TransformingAction classes, and their range will be the class IPEntity. Thus and in this example, every TransformingAction results in the production of a particular individual of the range IPEntity.

| Relation | Domain | Range |
|---------------------------|--------------------|----------|
| ResultsIn | TransformingAction | IPEntity |
| ComesFrom | IPEntity | IPEntity |
| RequiresAuthorisationFrom | Action | Role |

| | | |
|-------------|--------|----------|
| CanExercise | Role | Action |
| CanApply | Action | IPEntity |

Although the above list provides sufficient information regarding the relationships between classes, the list may be expanded as required.

5 Formal description of format BBC Metadata API

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:tns="http://www.broadcast.org/TVA_Interface/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="BBC_Interface"
targetNamespace="http://www.broadcast.org/TVA_Interface/"
xmlns:tva="urn:tva:metadata:2005" xmlns:mpeg7="urn:mpeg:mpeg7:schema:2005">
  <wsdl:types>
    <xsd:schema
targetNamespace="http://www.broadcast.org/TVA_Interface/"
xmlns:tns="http://www.broadcast.org/TVA_Interface/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:tva="urn:tva:metadata:2002"
xmlns:mpeg7="urn:mpeg:mpeg7:schema:2005" xmlns:CR="http://www.tv-
anytime.org/2002/02/ContentReferencing">
      <xsd:import namespace="urn:tva:metadata:2002"
schemaLocation="I:\tv-anytime\schema13\tva_metadata_v13.xsd"/>
      <xsd:import namespace="http://www.tv-
anytime.org/2002/02/ContentReferencing" schemaLocation="I:\tv-
anytime\schema\content_refv1.2spec(validateslocally).xsd"/>
      <!-- Channel Schema Definitions -->
<complexType name="channelInfoResponseType">
  <sequence>
<element name="TVAMain">
  <complexType>
  <sequence>
<element name="ProgramDescription">
  <complexType>
  <sequence>
<element name="ServiceInformationTable" type="tva:ServiceInformationTableType"/>
  </sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="channelLocationsResponseType">
  <sequence>
<element name="TVAMain">
  <complexType>
  <sequence>
<element name="ProgramDescription">
  <complexType>
  <sequence>
<element name="ServiceInformationTable" type="tva:ServiceInformationTableType"/>
  </sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>

```

```

</complexType>
<complexType name="channelListResponseType">
<sequence>
<element name="TVAMain">
<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="ServiceInformationTable" type="tva:ServiceInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<!-- Genre Schema Definitions -->
<complexType name="genreInfoResponseType">
<!-- This should be just the genre tag, needs looking up -->
<sequence>
<element name="TVAMain">
<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="genreMembersResponseType">
<sequence>
<element name="TVAMain">
<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="genreListResponseType">
<!-- Should be just repeated Genre tags -->
<sequence>
<element name="TVAMain">

```

```

<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<!-- Group Schema Definitions -->
<complexType name="groupInfoResponseType">
<sequence>
<element name="TVAMain">
<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="GroupInformationTable" type="tva:GroupInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="groupMembersResponseType">
<!-- Should this be constrained to crid only (not locations) -->
<sequence>
<element name="ContentReferencingTable" type="CR:ContentReferencingTableType"/>
</sequence>
</complexType>
<complexType name="groupListResponseType">
<sequence>
<element name="TVAMain">
<complexType>
<sequence>
<element name="ProgramDescription">
<complexType>
<sequence>
<element name="GroupInformationTable" type="tva:GroupInformationTableType"/>
</sequence>
</complexType>
</element>
</sequence>
<attribute name="lang" type="xsd:string"/>
</complexType>
</element>
</sequence>
</complexType>
<!-- Programme Schema Definitions -->
<complexType name="programmeInfoResponseType">
<sequence>

```

```

<element name="TVAMain">
  <complexType>
    <sequence>
      <element name="ProgramDescription">
        <complexType>
          <sequence>
            <element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
    <attribute name="lang" type="xsd:string"/>
  </complexType>
</element>
</sequence>
</complexType>
<complexType name="programLocationsResponseType">
  <!-- Should this be constrained to locations only (not crid) -->
  <sequence>
    <element name="ContentReferencingTable" type="CR:ContentReferencingTableType"/>
  </sequence>
</complexType>
<complexType name="programmeSearchResponseType">
  <sequence>
    <element name="TVAMain">
      <complexType>
        <sequence>
          <element name="ProgramDescription">
            <complexType>
              <sequence>
                <element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
                <element name="ProgramLocationTable" type="tva:ProgramLocationTableType"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
        <attribute name="lang" type="xsd:string"/>
      </complexType>
    </element>
  </sequence>
</complexType>
<complexType name="scheduleProgrammesResponseType">
  <sequence>
    <element name="TVAMain">
      <complexType>
        <sequence>
          <element name="ProgramDescription">
            <complexType>
              <sequence>
                <element name="ProgramInformationTable" type="tva:ProgramInformationTableType"/>
                <element name="ProgramLocationTable" type="tva:ProgramLocationTableType"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
        <attribute name="lang" type="xsd:string"/>
      </complexType>
    </element>
  </sequence>
</complexType>
</xsd:schema>

```

```

</wsdl:types>
<!-- Channel Message Definitions -->
<wsdl:message name="bbc.channel.getInfoRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="channelID" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.channel.getInfoResponse">
  <wsdl:part name="body" type="tns:channelInfoResponseType"/>
</wsdl:message>
<wsdl:message name="bbc.channel.getLocationsRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="channelID" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.channel.getLocationsResponse">
  <wsdl:part name="body" type="tns:channelLocationsResponseType"/>
</wsdl:message>
<wsdl:message name="bbc.channel.getListRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.channel.getListResponse">
  <wsdl:part name="body" type="tns:channelListResponseType"/>
</wsdl:message>
<!-- Programme Message Definitions -->
<wsdl:message name="bbc.genre.getInfoRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="genreID" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.genre.getInfoResponse">
  <wsdl:part name="body" type="tns:genreInfoResponseType"/>
</wsdl:message>
<wsdl:message name="bbc.genre.getMembersRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="genreID" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.genre.getMembersResponse">
  <wsdl:part name="body" type="tns:genreInfoResponseType"/>
</wsdl:message>
<wsdl:message name="bbc.genre.getListRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="active" type="xsd:bool"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.genre.getListResponse">
  <wsdl:part name="body" type="tns:genreListResponseType"/>
</wsdl:message>
<!-- Group Message Definitions -->
<wsdl:message name="bbc.group.getInfoRequest">
  <wsdl:part name="method" type="xsd:string"/>
  <wsdl:part name="groupID" type="xsd:string"/>
  <wsdl:part name="format" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="bbc.group.getInfoResponse">
  <wsdl:part name="body" type="tns:groupInfoResponseType"/>
</wsdl:message>
<wsdl:message name="bbc.group.getMembersRequest">
  <wsdl:part name="method" type="xsd:string"/>

```

```

        <wsdl:part name="groupID" type="xsd:string"/>
        <wsdl:part name="detail" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.group.getMembersResponse">
        <wsdl:part name="body" type="tns:groupMembersResponseType"/>
    </wsdl:message>
    <wsdl:message name="bbc.group.getListRequest">
        <wsdl:part name="method" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.group.getListResponse">
        <wsdl:part name="body" type="tns:groupListResponseType"/>
    </wsdl:message>
    <!-- Programme Message Definitions -->
    <wsdl:message name="bbc.programme.getInfoRequest">
        <wsdl:part name="method" type="xsd:string"/>
        <wsdl:part name="programID" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.programme.getInfoResponse">
        <wsdl:part name="body" type="tns:programmeInfoResponseType"/>
    </wsdl:message>
    <wsdl:message name="bbc.programme.getLocationsRequest">
        <wsdl:part name="method" type="xsd:string"/>
        <wsdl:part name="programID" type="xsd:string"/>
        <wsdl:part name="type" type="xsd:string"/>
        <wsdl:part name="codec" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.programme.getLocationsResponse">
        <wsdl:part name="body" type="tns:programmeLocationsResponseType"/>
    </wsdl:message>
    <wsdl:message name="bbc.programme.searchRequest">
        <wsdl:part name="method" type="xsd:string"/>
        <wsdl:part name="tile" type="xsd:string"/>
        <wsdl:part name="start" type="xsd:string"/>
        <wsdl:part name="end" type="xsd:string"/>
        <wsdl:part name="keyword" type="xsd:string"/>
        <wsdl:part name="limit" type="xsd:string"/>
        <wsdl:part name="detail" type="xsd:string"/>
        <wsdl:part name="sort" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.programme.searchResponse">
        <wsdl:part name="body" type="tns:programmeSearchResponseType"/>
    </wsdl:message>
    <!-- Schedule Message Definitions -->
    <wsdl:message name="bbc.schedule.getProgrammesRequest">
        <wsdl:part name="method" type="xsd:string"/>
        <wsdl:part name="start" type="xsd:string"/>
        <wsdl:part name="end" type="xsd:string"/>
        <wsdl:part name="channelID" type="xsd:string"
xsd:maxoccurs="unbounded"/>
        <wsdl:part name="limit" type="xsd:string"/>
        <wsdl:part name="detail" type="xsd:string"/>
        <wsdl:part name="format" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="bbc.schedule.getProgrammesResponse">
        <wsdl:part name="body" type="tns:scheduleProgrammesResponseType"/>
    </wsdl:message>

```

```

<wsdl:message name="NewOperationRequest">
  <wsdl:part name="NewOperationRequest" type="xsd:string"/>
</wsdl:message>
<!-- End of Message definitions -->
<wsdl:portType name="BBC_Interface">
  <wsdl:operation name="channelGetInfo">
    <wsdl:input message="tns:bbc.channel.getInfoRequest"/>
    <wsdl:output message="tns:bbc.channel.getInfoResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="channelGetLocations">
    <wsdl:input message="tns:bbc.channel.getLocationRequest"/>
    <wsdl:output message="tns:bbc.channel.getLocationResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="channelGetList">
    <wsdl:input message="tns:bbc.channel.getListRequest"/>
    <wsdl:output message="tns:bbc.channel.getListResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="genreGetInfo">
    <wsdl:input message="tns:bbc.genre.getInfoRequest"/>
    <wsdl:output message="tns:bbc.genre.getInfoResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="genreGetMembers">
    <wsdl:input message="tns:bbc.genre.getMembersRequest"/>
    <wsdl:output message="tns:bbc.genre.getMembersResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="genreGetList">
    <wsdl:input message="tns:bbc.genre.getListRequest"/>
    <wsdl:output message="tns:bbc.genre.getListResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="groupGetInfo">
    <wsdl:input message="tns:bbc.group.getInfoRequest"/>
    <wsdl:output message="tns:bbc.group.getInfoResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="groupGetMembers">
    <wsdl:input message="tns:bbc.group.getMembersRequest"/>
    <wsdl:output message="tns:bbc.group.getMembersResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="groupGetList">
    <wsdl:input message="tns:bbc.group.getListRequest"/>
    <wsdl:output message="tns:bbc.group.getListResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="programmeGetInfo">
    <wsdl:input message="tns:bbc.programme.getInfoRequest"/>
    <wsdl:output message="tns:bbc.programme.getInfoResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>
  <wsdl:operation name="programmeGetLocations">
    <wsdl:input message="tns:bbc.programme.getLocationsRequest"/>
    <wsdl:output
message="tns:bbc.programme.getLocationsResponse"/>
    <wsdl:fault name="nmtoken" message="qname"/>
  </wsdl:operation>

```



```

<wsdl:operation name="programmeSearch">
  <wsdl:input message="tns:bbc.programme.searchRequest"/>
  <wsdl:output message="tns:bbc.programme.searchResponse"/>
  <wsdl:fault name="nmtoken" message="qname"/>
</wsdl:operation>
<wsdl:operation name="scheduleGetProgrammes">
  <wsdl:input message="tns:bbc.schedule.getProgrammesRequest"/>
  <wsdl:output
message="tns:bbc.schedule.getProgrammesResponse"/>
  <wsdl:fault name="nmtoken" message="qname"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="BBC_InterfaceHTTP" type="tns:BBC_Interface">
  <http:binding verb="GET"/>
  <wsdl:operation name="channelGetInfo">
    <http:operation location="/bbc.channel.getInfo.html"/>
    <wsdl:input>
      <http:urlEncoded/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="text/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="channelGetLocations">
    <http:operation location="/bbc.channel.getLocations.html"/>
    <wsdl:input>
      <http:urlEncoded/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="text/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="channelGetList">
    <http:operation location="/bbc.channel.list.html"/>
    <wsdl:input>
      <http:urlEncoded/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="text/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="genreGetInfo">
    <http:operation location="/bbc.genre.getInfo.html"/>
    <wsdl:input>
      <http:urlEncoded/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="text/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="genreGetMembers">
    <http:operation location="/bbc.genre.getMembers.html"/>
    <wsdl:input>
      <http:urlEncoded/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="text/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="genreGetList">
    <http:operation location="/bbc.genre.list"/>

```

```

        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="groupGetInfo">
        <http:operation location="/bbc.group.getInfo.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="groupGetMembers">
        <http:operation location="/bbc.group.getMembers.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="groupGetList">
        <http:operation location="/bbc.group.list.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="programmeGetInfo">
        <http:operation location="/bbc.programme.getInfo.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="programmeGetLocations">
        <http:operation location="/bbc.programme.getLocations.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="programmeSearch">
        <http:operation location="/bbc.programme.search.html"/>
        <wsdl:input>
            <http:urlEncoded/>
        </wsdl:input>
        <wsdl:output>
            <mime:content type="text/xml"/>
        </wsdl:output>
    </wsdl:operation>

```

```

        <wsdl:operation name="scheduleGetProgrammes">
            <http:operation location="/bbc.schedule.getProgrammes.html"/>
            <wsdl:input>
                <http:urlEncoded/>
            </wsdl:input>
            <wsdl:output>
                <mime:content type="text/xml"/>
            </wsdl:output>
        </wsdl:operation>
    </wsdl:binding>
    <wsdl:service name="BBC_Interface">
        <wsdl:port binding="tns:BBC_InterfaceHTTP" name="BBC_InterfaceHTTP">
            <http:address
location="http://www0.rdthdo.bbc.co.uk/services/api"/>
        </wsdl:port>
    </wsdl:service>
</wsdl:definitions>

```

6 Bibliography

1. DE 12.1.2.1 “AXMEDIS-4HOME Use Case and Requirement Analysis”
2. TV-Anytime Metadata ETSI TS 102 822-3-2 v1.3.1 (2006-01); Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 2: System aspects in a uni-directional environment
3. DVB-CPT Blue Book:DVB CPCM Usage State Information (USI) DVB Doc A094 <http://www.dvb.org/technology/dvb-cpcm/a094.DVB-CPCM.pdf>
4. MPEG –REL: ISO/IEC 21000-5 Information technology – Multimedia framework (MPEG-21) – Part 5: Rights Expression Language
5. DMP: Digital Media Project; Approved Document No. 2 – Technical Reference: Architecture; 2005/04
6. Open Mobile Alliance: "DRM Specification 2.0"
7. Open Mobile Alliance: "DRM Content Format 2.0"
8. Open Mobile Alliance: "OMA DRM V2.0"
9. Open Mobile Alliance: "OMA DRM Rights Expression Language V2.0"
10. 3GPP TS 26.235 “Packet switched conversational multimedia applications; Default codecs”
11. 3GPP TS 26.110 “Codec for circuit switched multimedia telephony service; General description”
12. 3GPP TS 26.140 “Multimedia Messaging Service (MMS); Media formats and codecs”
13. 3GPP TS 26.234 "Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs"
14. Open Mobile Alliance: “Enabler Release Definition for DRM V2.0”.

15. Open Mobile Alliance: Normative support file to DRM v2, filename “OMA-DRM-ROAP-V2_0.xsd”

16. Open Mobile Alliance: support file to DRM v2, filename “OMA-DRM-TRIGGER-V2_0.xsd”

7 Glossary

| | |
|-----|--|
| TVA | TV-Anytime |
| CMO | Creation Model Ontology |
| OWL | Web Ontology Language |
| DVB | The Digital Video Broadcasting Project |
| OMA | Open Mobile Alliance |

8 ANNEX 1 Non Mobile Demonstrator Fact Sheet

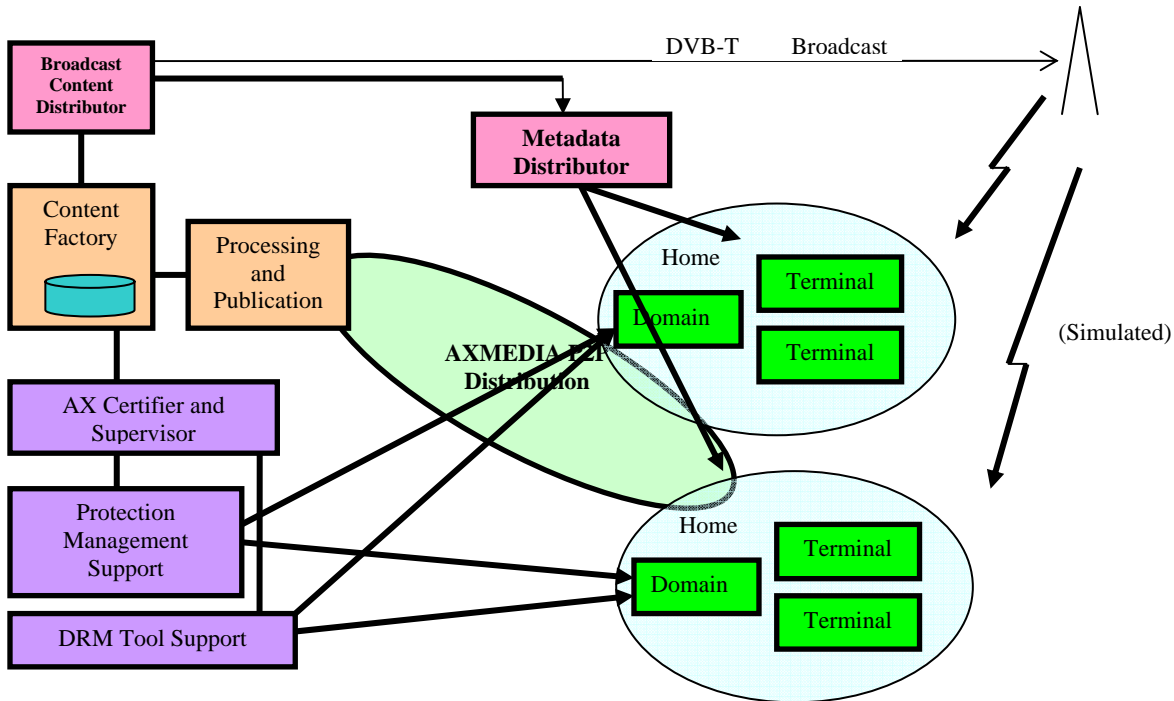
4HOME Demonstrator fact sheet: Content Distribution via Broadcasting and Domain Management

- **Main purpose:**

This aspect of the AXMEDIS 4HOME demonstrator shows how the AXMEDIS framework can be deployed to support the distribution of content over internet and broadcast channels to a rights-managed platform within the home.

The 4HOME demonstrator uses the rights management aspects of the AXMEDIS Framework on both source and client side to enable the rightful use of content distributed over the Internet, or on content derived from Free to Air broadcast content in the home and supplemented with other services over the Internet.

The 4HOME demonstrator uses downloadable tools on the home PC to give the Service Provider flexibility. The demonstrator incorporates License authoring and validation based upon the rules governing the Intellectual Property Entities within the composite content, such as ‘Work’ and ‘Manifestation’.



- **Review of the architecture integration with AXMEDIS:**

- **Acquisition / providing of content, where and how**

- Content is uploaded to the central 4HOME AXMEDIS content factory by participating partners. This is then adapted to form licensed, registered AXMEDIS objects for distribution over IP or mobile. Similar content is also made available over DVB-T transport, unprotected.

- **Production of content, where and how:**

- Content is produced using the AXMEDIS Editor at each 4HOME partner site and then stored (and shared) in the distributed database provided by the AXEPTool. It can be registered with the 4HOME public content factory for distribution over the AXMEDIA P2P.
 - number of content items produced per day - 1
 - number of content items produced at the same time - 1

- **Processing content, where and how**

- number of content items processed per day -1
 - number of content items processed at the same time -1

- **Protecting content, where and how**

- Content is protected and registered within the central content factory.

- Content packaged for the user into enhanced presentations on the user client are bound to the user's home domain or given an AXMEDIS license from the content factory and registration facility.
- **Mother licenses are produced, where and how**
 - Mother licenses are produced using the AXMEDIS Editor and associated to the AXMEDIS Object.
 - The AXMEDIS Ontology will be used to check that the creation of licenses is allowed to the corresponding AXMEDIS user and AXMEDIS Object.
- **Final licenses are produced, where and how**
 - The final licenses will be produced in the central content factory for the demonstrator. The AXMEDIS Editor should be also used for this purpose and the AXMEDIS Ontology should check if the license creation is permitted.
- **Registration of user and devices, where and how**
 - The 4HOME demonstrator uses the AXMEDIS home domain concept to license content to any device within a user home domain. Both users and devices are registered in the AXMEDIS PMS Domain Home and AXCS.
- **Distributing content, where and how**
 - Content is made available over IP in AXMEDIS format , selectable by the end user from an appropriate user interface. In addition, new AXMEDIS objects can be made on the home client from a combination of the Free to Air broadcast content and further enhancements delivered over the protected IP channel.
 - The Free to Air content can be selected for capture and enhancement through the AXMEDIS viewer
- **Description of the effective installation**
 - **Servers**
 - For the broadcast transport stream integration, the delivery of the transport stream to the end user client is through pre assembled MPEG streams served from a PC with DVB Asynchronous Streaming Interface (ASI interface) through a DVB-T modulator and fed by coax cable to the receiver set top box client.
 - The distribution of AXMEDIS content over the internet to the home users is achieved over the AXMEDIS AXMEDIA P2P tool for P2P distribution of protected content to the home.
 - **Portals**
 - The AXMEDIS client can retrieve programme description metadata from a metadata server using a simple protocol and delivering metadata in the TV-Anytime format.
 - **Distribution infrastructure needed if any;**
 - . The broadcast infrastructure is local and over coax. The Protected content and access to metadata is over the public internet using the AXMEDIS P2P tool and simple HTTP respectively.

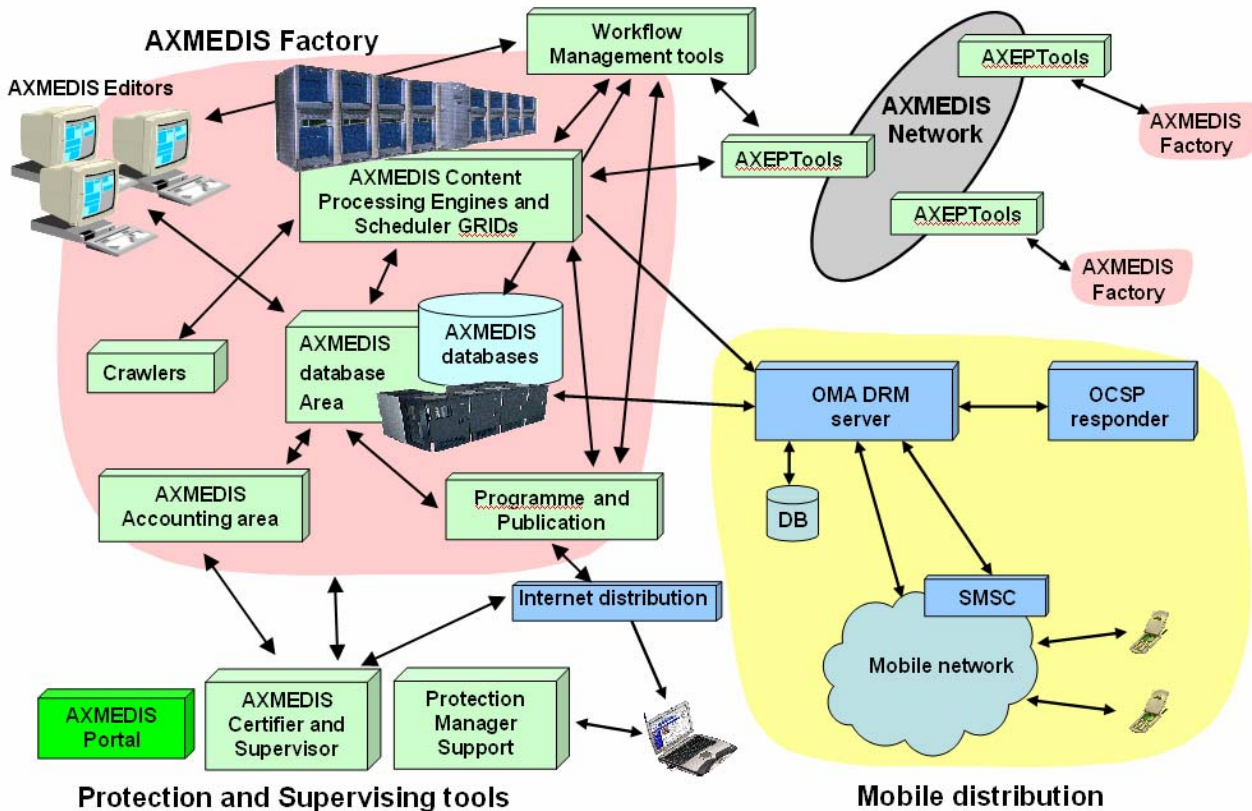
- **Streaming/downloads:**
 - The broadcast demo will be a pre-prepared MPEG2 transport stream with broadcast style programmes.
 - Some of these, and others, will be available for download over P2P as protected objects
- **Players needed:**
 - The 4HOME demonstrator uses PCs to show access to AXMEDIS content by the home user In some cases some PC's will be able to receive DVB-T content for rendering on the PC through the AXMEDIS player.
- **AXMEDIS tools**
 - **List of major AXMEDIS tools:**
 - Broadcast Streaming Interface
 - PMS – Server
 - PMS – Domain Home
 - PMS – Client
 - Domain Manager
 - Domain Registration Manager
 - AXMEDIA client and network support including publication tool
 - Protection Tool Server
 - Protection Processor,
 - DRM Editor and Viewer,
 - AXMEDIS Editor and Viewer
 - AXMEDIS Content Processing (AXCP) – to adapt content to different formats
 - AXEPTool
 - AXMEDIS Programme and Publication Engine
 - **AXMEDIS P2P .**
 - The 4HOME demonstrator content factory will be connected to other content factories through the AXMEDIS P2P network. In addition the home user can access 4HOME content from the content factory as licensed AXMEDIS objects over the AXMEDIA P2P Tool
 - **AXCP usage, yes/no, where and how:**
 - The Content Factory, including the AXCP will be set up at the Telecom Italia laboratory in Turin
 - **Workflow tools usage, yes/no, where and how: TBD**
 - **Programme and publication usage, yes/no, where and how: TBD**

- **PMS/AXCS usage, yes or no, where and how:**
 - The 4HOME demonstrator will utilize a tool server that in turn will register tools on the AXCS.
 - The 4HOME demonstrator will demonstrate the rights managed use of content in the home and within a home domain. This requires the functionality of the PMS modules.
- **AXMEDIS database usage, yes or no, where and how:**
 - A local instance of the AXMEDIS db in the 4HOME Content Factory will be used in order to store customized rules and content, though this may be a central provision.
- **Target Market:**
 - **Broadcasters:** The internet provides a low cost content distribution network with the possibility to support traditional broadcast services in a user friendly way. Additionally, new audiences based on mobile terminals is emerging and a common solution for managing these services from a common technological source is attractive.
 - **Collecting societies** – or other media organisations wishing to encourage and draw-in talent from a distributed artistic community with minimum technological and business thresholds of entry. Such a task can be made efficient through distributing artistic works in a rights managed form over the internet for use by various value change players. The key enabling mechanisms here are security of attribution, articulation of IP Entity being encapsulated and low cost distribution.
 -
- **Description of the business model**
 - The capture of free to air broadcast content for use within the home domain in a rights managed way. Such content will be managed on the home domain through enhancements utilizing the P2P network and permit sharing between homes through the referencing and delivery from the P2P back office of similar items
 - The provision of a P2P ‘catch up’ service in which users can choose content previously broadcast from a P2P catalogue and access fully protected AXMEDIS objects directly.
- **Description of content:**
 - **How many content AXMEDIS objects will be distributed**
 - **TBD**
 - **Who is going to provide digital resources with the needed clearance of rights:**
 - Currently sDae, BBC and TI are considering ways in which content can be made available for the demonstrator.
 - **Content description.**
 - BBC -A number of audio video clips.
 - TI - a limited number of music tracks, and videos.
 - sDae - around 50 videos of differing lengths and in the order of 1000 music tracks.
 - **Kind of resources:**

- BBC – Typical audio video programme material for the end user.
- TI – music (audio only), news and entertaining (audiovisual)
- sDae – video and music
- **Typical Content size for each content type: Audio....., video,**
 - These will be typical audio –visual presentations
- **Final Users/Clients:**
 - **How many final users will be reached.**
 - There will be at least 2 mobile clients shown receiving the content from the AXMEDIS factory and a number of PC clients constituting clients within the home domain (say 1-2 for each partner involved in the home domain aspect, so between 5 -10)
 - There will be a usability testing trial to gain feedback on the utility (usefulness) and ease of use of the system as perceived by end users in conjunction with the newly implemented home domain concept. This will involve observing and tracking how users use the technology and identify parts which do not work well and parts that are well aligned with the user expectation. The tests will be structured to obtain the most meaningful results and try to take into account the short and long term response as the user becomes more expert in the system under test. The test will be carried out on between 12-20 volunteers drawn from within the BBC, schools and possibly other outside volunteers. They may take place within a user testing laboratory within the BBC, or alternatively at the home, work desk or school of the volunteer, depending on the nature of the trial design. In addition, where possible, other value chain users within the BBC will be solicited to help evaluate parts of the system functionality that are not typically exposed to the end user, but more concerned with issues of content or archive re-purposing. Due to the nature of their specialist position within the value chain these are likely to be much fewer in number and cannot be solicited until the time of the trial.
 - **Their description:**
 - Members of the public, school children, co-workers.
 - **Their registration is needed:**
 - The registration procedure will be incorporated as part of the usability trial. It will be advantageous for users to be registered such that they can be given access to more content material that cannot be cleared for all territories or general use.
- **Partners involved and roles:**
 - BBC/ETRI: to provide the Broadcast Interface from DVB-T and metadata service.
 - ETRI: to provide the DRM Tool server for download by the home ETRI contractor will lead the work for distributing protection tools and accessing AXMEDIS DRM tools in AXMEDIS player.
 - Telecom Italia: to setup a content factory source and gateway with the mobile OMA platform
 - PKU: to implement and integrate the Home Domain architecture
 - sDae: to establish the Licence authoring and validation based on IP Entities

9 Annex 2 Demonstrator Fact Sheet for Mobile Distribution

4HOME Demonstrator fact sheet: Mobile distribution



- **Main purpose:**

The AXMEDIS 4HOME demonstrator shows how the AXMEDIS framework can be deployed to support the distribution of content over the mobile distribution channel. At the same time, it shows that the same content is made available in the native AXMEDIS format for download and use on PCs under the same conditions.

The 4HOME demonstrator uses the rights management aspects of the AXMEDIS Framework on the server side and the OMA DRM v2 corresponding features on the client side. To enable this scenario, an OMA gateway is introduced in the AXMEDIS architecture, which translates AXMEDIS Objects and the corresponding licenses from the source format into an OMA compatible format.

- **Review of the architecture integration with AXMEDIS:**

- **Acquisition / providing of content, where and how**

- Content is uploaded to the central 4HOME AXMEDIS content factory by participating partners. This is then adapted to form licensed, registered AXMEDIS objects for distribution over Internet and translated into OMA DRM v2 format for mobile distribution.
 - The format of content delivered to the final users over mobile will be OMA DCF 2.0 (based on ISO Base Media File Format).

- The format of content delivered to the final users over Internet will be the AXMEDIS format (based on MPEG-21).
- At least one content for each of the following types will be tested:
 - Still pictures
 - digital music
 - digital video
- **Production of content, where and how:**
 - Content is produced using the AXMEDIS Editor at each 4HOME partner site and then stored (and shared) in the distributed database provided by the AXEPTool.
 - number of content items produced per day - 1
 - number of content items produced at the same time - 1
- **Processing content, where and how**
 - Content is processed in the AXCP in order to adapt it to distribution on mobile devices. This adaptation includes translating the license from MPEG-21 REL to OMA DRM REL v2, and converting the content from MPEG-21 format to OMA DRM DCF v2.
 - number of content items processed per day -1
 - number of content items processed at the same time -1
- **Protecting content, where and how**
 - Content is protected and registered within the central content factory.
 - Content packaged for the user into enhanced presentations on the user client are bound to the user's home domain or given an AXMEDIS license from the content factory and registration facility.
 - Content adapted to mobile distribution is exported from the AXMEDIS Content Factory and imported into an OMA DRM server. The entity taking care of the protection of the adapted object is the OMA DRM Rights issuer, not the AXMEDIS PMS. At the same time, the original AXMEDIS Object remains under control of the AXMEDIS PMS.
- **Mother licenses are produced, where and how**
 - The content owner generates the AXMEDIS Object and the corresponding mother license using the AXMEDIS Editor, then distributes the protected AXMEDIS Object over the B2B peer to peer using the AXEPTool. The license is stored in the PMS located at DSI premises.
- **Final licenses are produced, where and how**
 - when an end user buys a protected content on the service provider portal, a final license is generated for him/her in AXMEDIS format for usage over the Internet;
 - subsequently, the rule to convert the license from AXMEDIS to OMA DRM REL is fired and the new license is sent to the OMA DRM server; the license is further processed by the OMA DRM server before delivery of the Rights Object to the mobile device, in order to provide the necessary protection information in the RO, as required by the OMA DRM v2 specification.

- **Registration of user and devices, where and how**
 - The 4HOME demonstrator uses the AXMEDIS home domain concept to license content to any device within a user home domain. Both users and devices are registered in the AXMEDIS PMS Domain Home and AXCS.
 - In the special case of mobile distribution, devices are registered in the OMA DRM Rights Issuer server by performing a 4-pass ROAP protocol. Users are not registered since OMA DRM binds content to each single device (or to a group of devices) owned by the end user. However, for purposes like charging and billing, the identity of the user can be obtained from the SIM card that is inserted in the mobile device. This is out of scope of OMA DRM.
- **Distributing content, where and how**
 - Content is made available over IP in AXMEDIS format and on the mobile distribution channel in OMA DRM format, selectable by the end user from an appropriate user interface.
- **Accounting collection and action monitoring, where and how**
 - metering is not supported in OMA DRM v 2.0; it will be introduced in OMA DRM v 2.1 which is not yet a published specification.
- **Description of the effective installation**
 - **Servers**
 - The mobile demo will utilize an OMA DRM 2.0 Server which will be adapted to operate as an OMA Gateway in order to be able to receive content from the AXMEDIS platform. The server is composed by an OMA DRM Rights Issuer which implements the ROAP protocol; an OCSP responder which checks and validates the certificates; a root Certification Authority issuing certificates for devices, Rights Issues and OCSP responder.
 - The distribution of AXMEDIS content over the internet to the home users is achieved over the AXMEDIS AXMEDIA P2P tool for P2P distribution of protected content to the home.
 - **Portals**
 - a simple portal for experimental access to OMA protected content will be setup. This will be based on Web or Wap technology.
 - **Distribution infrastructure needed if any;**
 - The OMA distribution infrastructure is provided locally by Telecom Italia.
 - **Streaming/downloads:**
 - The mobile demo is based on a download and play model. Streaming is not supported.
 - **Players needed:**
 - The 4HOME demonstrator uses PCs to show access to AXMEDIS content by the home user and 2 mobile Nokia hand terminals.

- The mobile demo will use two Nokia N91 commercial terminals equipped with certificates issued by the above mentioned root Certification Authority. Details on the Nokia N91 features are available at <http://forum.nokia.com/devices/N91>
- **AXMEDIS tools**
 - **List of major AXMEDIS tools:**
 - PMS – Domain Home
 - PMS – Client
 - Domain Manager
 - Domain Registration Manager
 - AXMEDIA client and network support including publication tool
 - Protection Tool Server
 - Protection Processor,
 - DRM Editor and Viewer,
 - AXMEDIS Editor and Viewer
 - AXMEDIS Content Processing (AXCP) – to adapt content to different formats
 - AXEPTool
 - AXMEDIS Programme and Publication Engine
 - **AXMEDIS P2P .**
 - The 4HOME demonstrator content factory will be connected to other content factories through the AXMEDIS P2P network. In addition the home user can access 4HOME content from the content factory as licensed AXMEDIS objects over the AXMEDIA P2P Tool. Mobile users cannot access the AXMEDIS P2P and will get protected content via GPRS or UMTS.
 - **AXCP usage, yes/no, where and how:**
 - The Content Factory, including the AXCP will be set up at the Telecom Italia laboratory in Turin, and will be used to adapt AXMEDIS contents and licenses for distribution over the mobile channel based on OMA technology.
 - **Workflow tools usage, yes/no, where and how:**
 - Possibly used for content production and translation to OMA, in the 4HOME Content Factory
 - **Programme and publication usage, yes/no, where and how:**
 - Possibly used for content production and translation to OMA, in the 4HOME Content Factory
 - **PMS/AXCS usage, yes or no, where and how:**
 - The 4HOME demonstrator will demonstrate the export of content from AXMEDIS to OMA. Content initially under control of the PMS will be translated by the AXCP into OMA format and exported on TI OMA DRM server.

- **AXMEDIS database usage, yes or no, where and how:**
 - A local instance of the AXMEDIS db will be installed in the 4HOME Content Factory in order to store customized rules and content adapted to mobile distribution. Initially a central provision may be used.
- **Target Market:**
 - **Mobile content service providers.** The OMA standard provides a basis for the provision and sale of high value content over the mobile platform. This requires a robust back end for the appropriate pre-processing of content.
- **Description of the business model**
 - simple sell of music tracks: unlimited licenses, each track is sold individually.
 - “all you can eat” model: the user pays a monthly fee, which entitles him/her to play whatever music track he/she likes, for one month. The license expires at the end of the month.
- **Description of content:**
 - **How many content AXMEDIS objects will be distributed:**
 - 10-20 content objects will be selected and translated from the AXMEDIS Format to OMA DRM format and will be available both on the mobile distribution channel as OMA DRM objects and on the Internet as AXMEDIS Objects.
 - More objects will be available on the Internet distribution channel in AXMEDIS format only.
 - **Who is going to provide digital resources with the needed clearance of rights:**
 - Currently sDae, BBC and TI are considering ways in which content can be made available for the demonstrator.
 - **Content description.**
 - BBC -A number of audio video clips.
 - TI - a limited number of music tracks, and videos.
 - sDae - around 50 videos of differing lengths and in the order of 1000 music tracks.
 - **Kind of resources:**
 - BBC – Typical audio video programme material for the end user.
 - TI – music (audio only), news and entertaining (audiovisual)
 - sDae – video and music
 - **Typical Content size for each content type:**
 - short clips that can be stored on mobile devices without requiring memory expansions. Typically 3-4 MB music tracks, 6-8 MB video clips. Bigger files may require unfeasible download time.
- **Final Users/Clients:**
 - **How many final users will be reached.**

- There will be at least 2 mobile clients shown receiving the content from the AXMEDIS factory and a number of PC clients constituting clients within the home domain (say 1-2 for each partner involved in the home domain aspect, so between 5 -10)
- **Their description:**
 - skilled users, members of the technical staff.
- **Their registration is needed:**
 - Registration of mobile devices on the OMA Rights Issuer is automatically performed via the 4-pass ROAP protocol.
 - Registration of SIM cards authenticating the identity of the end user is performed by the mobile operator as a normal commercial procedure.
- **Partners involved and roles:**
 - BBC/sDae/TI: to provide test content in AXMEDIS Format
 - Telecom Italia: to setup a content factory source and gateway with the mobile OMA platform
 - PKU: to implement and integrate the Home Domain architecture