



Automating Production of Cross Media Content for Multi-channel Distribution

www.AXMEDIS.org

DE7.2.1

*Training material for the introduction of
user aspects of AXMEDIS production tools*

Basic AXMEDIS Training Courses

Version: 2.1

Date: 13/11/2005

Responsible: UNIVLEEDS (kia@computer.org) (revised and approved by DSI)

Project Number: IST-2-511299

Project Title: AXMEDIS

Deliverable Type: Public

Visible to User Groups: Yes

Visible to Affiliated: Yes

Visible to Public: Yes

Deliverable Number: DE7.2.1

Contractual Date of Delivery: M14

Actual Date of Delivery: 13/11/2005

Work-Package contributing to the Deliverable: WP2, WP3, WP4, WP5, WP7, WP11

Task contributing to the Deliverable: WP2, WP3, WP4, WP5, WP7, WP11

Nature of the Deliverable: Report

Author(s): All partners

Abstract:

This report presents training and demo materials for the introduction of user aspects of AXMEDIS framework and tools. It provides the first basic AXMEDIS Training Course with particular focus for the first AXMEDIS International Conference (AXMEDIS2005). The main structure of the training and demo materials can be divided into three sections, representing the key topics of the project, with: (i) general overview (ii) production and (iii) distribution.

Keyword List:

Training, Demonstration, Dissemination, Production, Distribution, Protection, User aspects

Table of Contents

1	EXECUTIVE SUMMARY AND REPORT SCOPE (UNIVLEEDS, ALL)	4
2	INTRODUCTION (UNIVLEEDS, ALL)	4
3	PUBLICATION OF TRAINING MATERIAL (ALL)	4
3.1	VALIDATION OF TRAINING MATERIAL (UNIVLEEDS, DSI, ALL)	4
3.2	AXMEDIS TRAINING SLIDES (ALL)	5
3.2.1	AXMEDIS Preface for the Slides	6
3.2.2	AXMEDIS Final Tutorial/Demonstration Slide	6
3.3	AXMEDIS TRAINING DOCUMENTATION (ALL)	7
3.3.1	AXMEDIS Preface for Documentation	7
3.4	PUBLICATION VIA THE WWW SITE (EXITECH)	7
3.4.1	Portal Organisation for Training Material and Documentation (EXITECH)	8
3.4.2	Specification for Uploading and Downloading Tutorial Content (EXITECH)	8
4	TRAINING/TUTORIAL PREPARATIONS FOR AXMEDIS2005	10
4.1	AXMEDIS TRAINING COURSE: AXMEDIS OVERVIEW (DSI, TISCALI, EUTELSAT, AFI)	10
4.1.1	Course objectives	10
4.1.2	Intended audience	11
4.1.3	Course Content	12
4.1.4	Course Details	14
4.2	AXMEDIS TRAINING COURSE – AXMEDIS PRODUCTION (ILABS, XIM, DSI, IRC)	14
4.2.1	Course objectives	14
4.2.2	Intended audience	14
4.2.3	Course Content	14
4.2.4	Course Details	15
4.3	AXMEDIS TRAINING COURSE - AXMEDIS DISTRIBUTION (TISCALI, EUTELSAT, ILABS, UNIVLEEDS)	15
4.3.1	Course objectives	15
4.3.2	Intended audience	16
4.3.3	Course Content	16
4.3.4	Course Details	16
5	ADDITIONAL AXMEDIS TRAINING COURSES AND PLANS (ALL)	16
5.1	TRAINING COURSE: AXMEDIS INFRASTRUCTURE AND ARCHITECTURE (DSI, EXITECH. ...)	17
5.2	TRAINING COURSE: USER-CENTRED PRODUCT CREATION AND VALIDATION (ACIT)	17
5.3	TRAINING COURSE: AXMEDIS BUSINESS (TISCALI. ...)	18
5.4	TRAINING COURSES FOR DECISION MAKERS (DSI, EXITECH, TISCALI, XIM, ACIT, CRS4, FUPF, ILABS, EUTELSAT, FHGIGD, UNIVLEEDS, COMVERSE)	19
5.5	TRAINING COURSE: AXMEDIS CONTENT (ILABS)	19
5.6	TRAINING FOR PRODUCING CONTENT WITH THE AUTOMATIC TOOLS (UNIVLEEDS)	20
5.7	TRAINING ON PROTECTION ASPECTS AND DRM RULES DEFINITION (FUPF, FHGIGD)	20
5.8	TRAINING COURSES ON INSTALLATIONS, CONFIGURATIONS, MAINTENANCES AND MANagements (DSI, EXITECH, TISCALI, ...)	21
6	DEMONSTRATION ACTIVITIES (UNIVLEEDS, ALL)	21
6.1	CONTENT MODEL AND PRODUCTION OVERVIEW	22
6.1.1	AXMEDIS Model (DSI)	22
6.1.2	AXMEDIS editor/authoring/renderer (DSI, SEJER)	23
6.1.2.1	Contextualisation of AXMEDIS editor/authoring/render	23
6.1.2.2	Demonstration of AMEDIS Editor Scenarios	24
6.1.2.3	Demonstration of AXMEDIS Editor Tools	25
6.1.3	AXMEDIS Content Processing (DSI)	32
6.1.3.1	Contextualisation of the AXMEDIS Content Processing Area	32
6.1.3.2	Demonstration of AXMEDIS Content Processing Area Scenarios	33

<i>DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools</i>	
6.1.3.3	Demonstration of AXMEDIS Content Processing Area Tools35
6.1.4	Content Gathering (FOCUSEEK)42
6.1.4.1	Contextualisation of the focuseek gathering system.....44
	The searchbox platform is used by the AXMEDIS system to gather information from Content Management System of various types.....44
6.1.4.2	Demonstration of gathering Scenarios44
6.1.4.3	Demonstration of gathering Tools44
6.1.5	AXMEDIS Database (EXITECH).....53
6.1.5.1	Contextualisation of AXMEDIS Database54
6.1.5.2	Demonstration of AXMEDIS Database Tools56
6.2	ADVANCED CONTENT PRODUCTION ASPECTS81
6.2.1	SMIL Editor and Player.....81
6.2.2	Adaptation (FHGIGD, DIPITA, EPFL, etc)83
6.2.3	Fingerprint (FHGIGD, DIPITA, EPFL, etc)85
6.2.4	Workflow (IRC, HP, XIM).....87
6.3	CONTENT DISTRIBUTION AND SUPPORTING TOOLS97
6.3.1	AXEPTool (CRS4)97
6.3.1.1	Contextualisation of the AXEPTool area.....97
6.3.1.2	Demonstration of Scenario: Sharing content in more than one factory with AXEPTool.....97
	Requirement:97
6.3.1.3	Demonstration of Tools99
6.3.2	Programme and Publication (UNIVLEEDS)102
6.3.2.1	Introduction.....102
6.3.2.2	Contextualisation of P&P Area102
6.3.2.3	Demonstration of P&P Scenarios.....103
6.3.2.4	Demonstration of P&P Tools104
6.3.3	Query on Demand (FHGIGD)110
6.4	CONTENT PROTECTION TOOLS113
6.4.1	DRM Tools (FUPF)113
6.4.1.1	Edition and visualisation of DRM information113
6.4.2	Content consumption scenario.....119
6.4.3	Legal Aspects (AFI).....123
6.4.3.1	Demonstration of Scenarios123
6.4.4	Certifier and Supervisor (DSI).....124
6.4.4.1	Contextualisation of Certifier and Supervisor124
6.4.4.2	Demonstration of Certifier and Supervisor Scenarios125
6.4.5	Accounting and Reporting (EXITECH).....126
6.4.5.1	Contextualisation of CAMART and AII.....126
6.4.5.2	Demonstration of AII Scenarios and mock-ups129
6.5	DISTRIBUTION.....131
6.5.1	Distribution to PC (TISCALI, SEJER)131
6.5.1.1	Demonstration of Internet Distribution via Web Scenarios132
6.5.1.2	Demonstration of Internet Distribution via Web Tools132
6.5.2	Distribution to Mobiles (COMVERSE, DSI, ILAB, IRC)137
6.5.2.1	Demonstration of licence translation for the mobile scenario.....137
6.5.2.2	Demonstration of licence authorisation for the mobile scenario138
6.5.3	Distribution via Kiosks (ILABS).....139
6.5.4	Distribution via Satellite Data Broadcast (EUTELSAT)140
6.5.4.1	Demonstration of Satellite Distribution Scenarios141
6.5.4.2	Demonstration of Satellite Distribution Tools142
7	AXMEDIS TRAINING MATERIAL PUBLICATION (ILABS, UNIVLEEDS, DSI, ...)147
8	BIBLIOGRAPHY147

1 Executive Summary and Report Scope (UNIVLEEDS, ALL)

This report presents the first version of basic AXMEDIS training and demonstration materials for the AXMEDIS project. It includes materials for an overview introduction to AXMEDIS framework with the architecture and benefits, production tools, protection tools and distribution tools. This version of the basic tutorial courses is focused on the contents for the First AXMEDIS International Conference to be held in Florence, Italy, 30 November to 2 December 2005. The materials can be divided into three sections: (i) AXMEDIS general overview; (ii) AXMEDIS production, and (iii) AXMEDIS Distribution.

2 Introduction (UNIVLEEDS, ALL)

This report presents the first version of basic training courses and materials of the AXMEDIS project. The organisation and management of training aspects are not straight forward due to the complexity of the project. In the same way as the development of the framework and the integration of all the modules and packages, training materials are developed by different partners who are responsible for different parts of the development and are integrated together in this document.

In order to provide an easy to understand and consistent training and demonstration materials that would be more effective for the general dissemination, it is vital that the format and stylistic aspects are organised and recommended in the early stage of the development.

In addition to training materials for the introduction of user aspects of AXMEDIS production tools, public training activities with tutorials and demonstrations, a set of guideline and templates for the creation of the training related materials including presentation slides, document styles and general questionnaires are prepared and available online at the project portal for the preparation of the training and demonstration materials. This version of the basic training courses are particular focused on the AXMEDIS2005 International Conference to be held 30 November to 2 December 2005 in Florence, Italy. Specific sessions have been allocated during the conference to present the materials as presented in this report.

Template resources for different Training and Demonstration materials available online at the project website include:

- AXMEDIS DE7-2-1 template resource – Training Manual blueprint, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1183
- AXMEDIS DE7-2-1 template resource – Training Slides, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1184
- AXMEDIS DE7-2-1 template resource – Training Event Questionnaire, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1185
- AXMEDIS DE7-2-1 template resource – Training Attendance sheet, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1186

3 Publication of Training Material (ALL)

3.1 Validation of training material (UNIVLEEDS, DSI, ALL)

In order to ensure a wide usage, distribution and effective adaptation of the AXMEDIS platform, it is important to create tutorial material with the following features:

- easy to follow
 - use simple and clear English; short sentences;
 - use appropriate language for the audience (technical/non-technical depending on the targeted audience)
 - step-by-step guide

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- material should be well supported by visuals (use pictures or screen snapshots to demonstrate steps, if possible)
 - make use of diagrams to help explain components/relationships/steps etc.
 - use screenshots when discussing software components
- avoid conveying information through colour
- use high contrast colours when you must convey information through colour
- use bold rather than italic or underline to emphasise text
- avoid upper case blocks of text (hard to read)
- divide content into sections and make it clear where one point/section ends and another begins
- complete and with clear purpose
- include some form of validation to measure the user's progress and understanding
- All training material will include the following elements:
 - short introduction to the AXMEDIS project with project URL and contact.
 - clearly states
 - the purpose of this tutorial (aims and objectives)
 - how this tutorial fits into the scope and structure of AXMEDIS (i.e. setting the context)
 - the expected “learning outcomes” on completion of the session

All training materials such as tutorial notes and course documentation must confirm to a standard style. Templates for producing these materials are available on the project website. UNIVLEEDS and DSI support general editing duties on request.

3.2 AXMEDIS Training Slides (ALL)

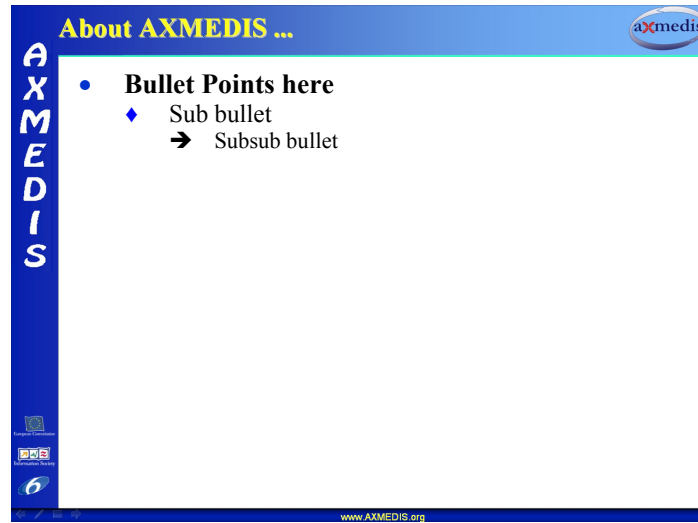
The layout of the training course slides includes the following compulsory slides and layout to provide a consistent look and feel:

- Cover Page
 - Project name
 - Project URL
 - Name of training course
 - Version
 - Training course URL (if available)
- Preface
 - Introduction and overview of the AXMEDIS project
- Table of contents
 - The sections in the Training
- Introduction of training course
 - Show where this fits into the overall AXMEDIS framework, to help the participants to quickly get a feel of the AXMEDIS framework. If appropriate, highlight specific components on a standard architecture diagram used in the introduction and overview to provide contextualisation information
 - Establish main goals of the training material
- Training slides
 - Keep the layout/design simple
 - Use legible fonts
 - Use bullets or numbered lists
 - Keep to one subject per slide
 - Do not squeeze too much information onto one slide (6 bullet points is optimal)
 - Break up long presentations (maybe with some multimedia aids) to keep trainee's attention
 - When using bullet points, avoid more than two levels of nesting

- Closing section
 - Conclusion/summary of training material
 - Show that the initial goal has been met
- Last slide:
 - Other related courses available
 - Contact details

3.2.1 AXMEDIS Preface for the Slides

This section specifies the generic preface section for all tutorial and demonstration slides to provide the introduction and overview of the AXMEDIS project.



3.2.2 AXMEDIS Final Tutorial/Demonstration Slide

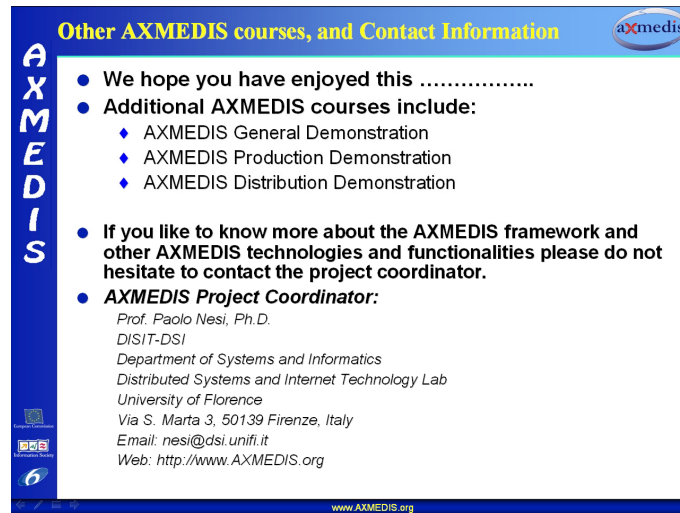


Figure: A sample screen snapshot of the final AXMEDIS tutorial slide

3.3 AXMEDIS Training Documentation (ALL)

This section specifies the style and format of the AXMEDIS training documentations. A Microsoft Words template is available on the project web portal as specified in the Introduction section of this report.

All training material is to adhere to the following layout

- Cover Page
 - Project name
 - Project URL
 - Name of training course
 - Version: x.y
 - Training course URL (if available)
 - Author(s) and affiliation(s)
- Preface
 - Introduction and overview of the AXMEDIS project
- Table of contents
 - The sections in the tutorial
- Introduction of training course
- Preparation: resources required and installation guide (if necessary)
- Training course material
- Summary and FAQ
- References
- Glossary
- Other related courses available

3.3.1 AXMEDIS Preface for Documentation

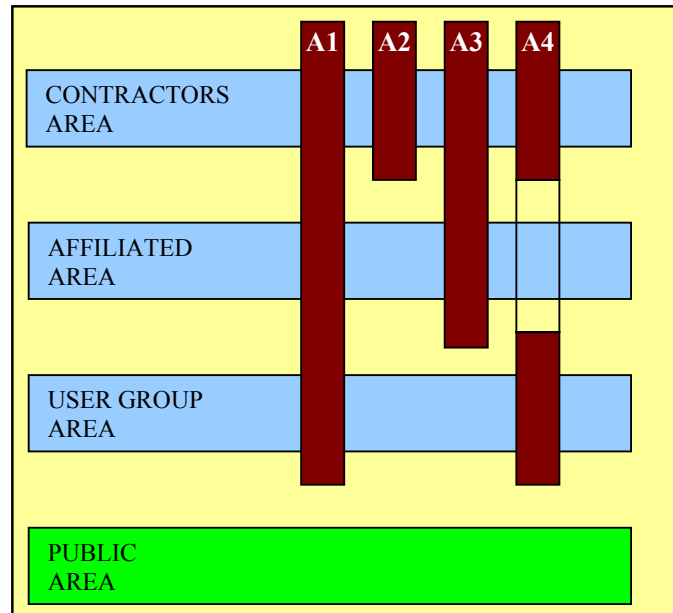
This section specifies the generic preface section for all tutorial and demonstration documentation providing the introduction and overview of the AXMEDIS project.

3.4 Publication via the WWW site (EXITECH)

The AXMEDIS web portal is used as a support for the distribution of the training material. A special activity section on the project portal is designated for this purpose.

3.4.1 Portal Organisation for Training Material and Documentation (EXITECH)

The AXMEDIS portal is mainly structure by “areas” and “activities”. The activity designated to the training can be managed by the activity administrator in order to gain visibility in more than one areas (see the web specification for more details).



The “WP7 –Training” activity contents are organised by the activity administrator in a folder structure that is ordered by the material classification. Current organisation of the folder is as followed:

generic
Material
— AXMEDIS Overview
— Production Overview
— Distribution Overview
— Other Tutorials
Model and Forms

Each folder contains all the appropriate training materials in all formats.

3.4.2 Specification for Uploading and Downloading Tutorial Content (EXITECH)

In order to download the Tutorial contents, a user has to be registered on the AXMEDIS portal because the training activity is located in the internal areas (for Project Contractors, User Group members and Affiliated members). The number of downloads for each training content will be recorded for project assessment purposes.

Uploading of training materials are performed by registered project partners with valid account for the project web portal. As stated in the AXMEDIS portal specification, the upload procedure is structured in the following steps:

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Login on the web portal
- Enter into the WP7-Training activity (from the personal area)
- Select upload doc link from the top of the page
- The training activity is already selected and the user has only to
 - Select other activity where the content should be visible (if any)
 - Select the folder where the content has to be posted

Select activity:	Select destination group:
TP2 Requirement analysis	generic
AXMEDIS Specification	generic
Management	generic
Meetings	generic
Assessment	generic
Dissemination	generic
Tech-watch	generic
AXFW Maintenance	generic
Call for Take UP	generic
Test Cases, spec and collection	generic
TP2-3-1 UserGroup	generic
Content Spec and Manag	generic
Research Guidelines and Spec	generic
Exploitation	generic
TP4 and WP5	General on AXFW and Guidelines
Standardisation	generic
TP9-Dist Internet	generic
TP9-Dist Satellite	generic
TP9-Dist Mobiles	generic
TP9-Dist Kiosks	generic
TP7-Training	generic
TP9 CMS integration	generic
TP9 Cont Proc Integ	generic

At this point only the content visibility has to be completed

Document visible for:
(each tick automatically means tick for every row atop)

Contractors	<input checked="" type="checkbox"/>
Affiliated	<input type="checkbox"/>
User Group	<input type="checkbox"/>
Public area	<input type="checkbox"/> select destination group: generic

In the example presented above, the uploaded material will only be visible for the contractors even if the WP7 activity is publicly available.

Note that there is restriction on the size of the material. Files that are over 40 MB cannot be uploaded directly. This limit will be monitored and adjusted based on the performance of the portal throughout the course of the project. The limit is indicated on the upload page.

To upload any files over the size threshold, contact the webmaster for assistant.

4 Training/Tutorial Preparations for AXMEDIS2005

This section discussed the Training/Tutorial plan for the first AXMEDIS International Conference (AXMEDIS2005) based on the experience from the first demonstration of the results (of the 6 months work) at the 2nd User Group meeting (5-6 September 2005).

From the feedbacks and discussions with the User Group (UG) experts at the UG meetings, particularly during the 2nd UG meeting where the first non-internal demonstrations were held, it is evident that simple and high level contextualisation is required in order to bring over this complex and comprehensive framework. Demonstration/training requires a concise set of focuses. Due to the complex nature of the framework which encompasses a wide range of technologies with many application areas including both the value-chain for B2B and B2C, the interconnections and individual sections can be very easily misunderstood and confused.

The following main topics represent the main focus for the first public Training plan to be carried out at the AXMEDIS 2005 International Conference:

- AXMEDIS Overview -- a general introduction to AXMEDIS
- AXMEDIS Production
- AXMEDIS Distribution

The general introduction to AXMEDIS will be the key theme for the tutorial at AXMEDIS2005. The Production and Distribution sections have less detail and focus, in comparison to the General Overview section because most of the production and distribution tools are under development at the time of writing. Additionally, it is not useful for the public audients/participants to learn about low level technical design and development issues which are more appropriate to be presented as technical papers.

The overall design policy recommends each tutorial for the responsibility of a small group of partners, forming a working group, to develop the specific tutorial. On completion, the materials and outcomes are distributed to all partners for comments and feedbacks. All comments and feedbacks are analysed for iterative refinement by the working group in order to polish and improve the tutorials.

All public tutorials, demos and training related materials should use the approved standard style as recommended above, which includes templates for:

- manual and document (in MS Word)
- presentation (in PowerPoint)
- attendance sheet
- sample short questionnaire

4.1 AXMEDIS Training Course: AXMEDIS Overview (DSI, TISCALI, EUTELSAT, AFI)

4.1.1 Course objectives

This is an overview training course which provide overall high-level introduction to the AXMEDIS Framework, starting from the overall project aims and objectives, and provide various holistic views to show the potential applications and plausible usage with examples.

This training course is the first of a series to provide an overview of the AXMEDIS solutions and architecture.

- The Overview tutorial aims to provide answers to the following question for the attendees:
 - What is AXMEDIS?
 - Which is the State of the Art of the sector?

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Which are the innovations/advantages of AXMEDIS?
- What can I do with AXMEDIS?
- Why should I adopt AXMEDIS?
- Which are the major tools of AXMEDIS?
- How can I use AXMEDIS for my business?
- Which are the implications and benefits in using AXMEDIS?
- Who can help me to know and learn more?
- etc
- Duration: approximately 3.5 to 4 hours

By the end of the course, attendees will understand:

- The core AXMEDIS concepts of peer-to-peer for collaborative business-to-business production, automatic content formatting, DRM and workflow management;
- The basic lifecycle of an AXMEDIS object through a typical workflow from production, through aggregation/integration and cross-channel distribution to the consumer;
- The components of the AXMEDIS Framework and their respective functions and relationships;
- Suitable applications of AXMEDIS tools in content creation, integration and distribution;
- An overview of the schedule of AXMEDIS training courses being developed, and therefore where to look for more detailed information, for example concerning AXMEDIS APIs, business issues, etc.;
- The key challenges of AXMEDIS solutions
- The interaction of AXMEDIS components with existing technologies
- The impact of AXMEDIS in the content production and distribution value chain;
- The implication for content owners and users
- Choosing the right content for the right channel;
- The essential role of AXMEDIS DRM solutions and licensing models;
- The benefit of the tracking and monitoring process in rights collection and sharing;
- Overview of AXMEDIS B2B and B2C strategies and solutions.

4.1.2 Intended audience

Although the course will present the “big picture” of AXMEDIS from a mainly technical perspective, it will do so at a sufficiently high level and with minimal jargon for it to be relevant to business and end-user audiences as well as technical experts.

The Overview tutorial is directed towards the following attendees:

- Individuals interested to have a general overview of AXMEDIS solution and tools
- Business Managers interested in assessing if AXMEDIS is suitable for their business or not, and if it is of beneficial to their business or not, etc.
- Technical Managers interested to have a first overview
 - the impact of AXMEDIS technology in their solutions
 - the technical innovation of AXMEDIS with respect to the state-of-the-art
- Research Managers interested to have an overview of the whole architecture and assessing the advantages of using the AXMEDIS framework for:
 - testing and developing new solutions, new algorithms, etc., reducing costs of infrastructure creation
 - Promoting and distributing their new solutions for a large audience

Audience should include people coming from the following sectors:

- Content producing, publishing and distribution
- Licensing companies
- New Media Strategy and Business Development
- Emerging Platforms and Technologies
- Collective rights management
- Anti-Piracy Initiatives

- Product Marketing
- Multimedia Content
- Mobile, broadcaster and Interactive domains

4.1.3 Course Content

The training course on AXMEDIS General Overview is structured as followed:

- Preface
 - Purpose of this Tutorial
 - Schema of the Tutorial and Purpose
- What is AXMEDIS
 - What is AXMEDIS?
 - Main general Objectives
 - Business Opportunities
- Business Opportunities
 - Content and channels
 - Multi-channel Problems
 - Applications of Content Distribution with protection
- General State of the Art
 - The Value Chain
 - Rights Management
 - Terminology
 - Motivation for DRM
 - Single channel distribution and DRM
 - Content Elements and Distribution
 - Combination of elements
 - Open model
 - Governed objects
 - Augmented licence
 - Third Party Distribution Examples
 - Who are the major players
 - Microsoft Windows Media
 - Apple I-Tune
 - Super-distribution concept
- Needs and Limitations
 - The effective User Needs and AXMEDIS Motivations
 - Business Rules, a way to formalize allowed rights
 - Content distribution, the multi-channel needs
 - Limitations of the Current DRM solutions
 - The Convergence, and the Multi-channel Problems
 - Convergence, the Interoperable Content
 - Present Needs for Accessing New Markets
 - EX: Convergence, the Interoperable Licence, DRM
 - EX: Convergence, the Unified Licence, DRM
 - Real Time Content Processing
 - B2B Rights Management
 - Windows Media vs AXMEDIS
 - AXMEDIS and the value chain
- AXMEDIS Content Packaging
 - AXMEDIS Package
 - AXMEDIS Object Elements
 - AXMEDIS Object
 - AXMEDIS Object: structure
 - Digital Item Declaration example
- AXMEDIS Content Protection and Tracking

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Authentication and certification
 - registration
- Licence Definition and Production
 - AXMEDIS MPEG-21 REL Licence
 - MPEG-21 REL data model
 - ODRL vs MPEG-21
 - AXMEDIS Licence Production
 - AXMEDIS The Protection and Control Process
- Protection Information & low level technologies
- Tracking: Supervision and Control
 - AXMEDIS Control and Supervision
 - List of Events/Actions on rights Exploitation
 - Providing Action Log and Statistical Information
 - Black Lists Management
- AXMEDIS Content Production and Processing
 - Content Authoring, Editing
 - AXMEDIS Editor and Viewer
 - Automating Content Processing
 - Content production with GRID support
 - Workflow
 - Workflow integration with AXMEDIS
 - Content Gathering/Ingestion
 - Content Management Systems Crawler
 - AXMEDIS Objects Management Systems
 - AXMEDIS database
 - Content description and indexing
 - Technical queries
 - AXMEDIS Distribution tools
 - AXEPTool for P2P B2B distribution
 - Programme and Publication tool
 - AXMEDIS Client tools
 - AXMEDIS clients
 - AXMEDIA tool for P2P content sharing on C2C
- Examples of AXMEDIS Applications
 - General architecture for content distribution
 - Architectures for Single Channel
 - Distribution on Internet, the TISCALI demonstrator
 - Distribution via Kiosks, the ILABS demonstrator
 - Distribution via Satellite data broadcast, the EUTELSAT Demonstrator
 - Multi-channel Architecture

The course will include many practical examples, case studies and scenarios in order to explain the physical operation of the AXMEDIS Framework in practice, as well as presenting at a high level the theory and technology behind AXMEDIS including

- MPEG21
- DRM
- workflow management
- automated formatting
- specific targeted courses including courses for librarians, archivists, museums and theatre staff
- etc.

4.1.4 Course Details

As an introduction to this overview course, a high-level presentation on the aim and objective of AXMEDIS is included. From the project aim and objectives, focus will be moved to expected results and impacts (including the specific project take-up action) covering all the major benefits coming from the adoption of AXMEDIS solution, framework and tools. Overall architecture will be introduced in some detail starting from the framework infrastructure to pass then to tools (both at level of suites and at level of individual ones) in order to provide the attendee not only with the overall picture but also with a quite detailed and precise understanding of the adopted terminology and provided overall functionalities. In this high level (even though detailed) exposition separate sections will be devoted to P2P, DRM and distribution channels. Finally attendee attention will be focused on the foreseen demonstrators in terms of aims, scope and schedule so to give a clear picture of which are the addressed channels, what and how will be demonstrated, how it will be possible to benefit from such demos (especially in relation to take-ups).

4.2 AXMEDIS Training Course – AXMEDIS Production (ILABS, XIM, DSI, IRC)

4.2.1 Course objectives

The aim of the course is to give at the attendees a better idea of what can be created using the AXMEDIS production tools and framework, and which are the major benefits.

The main objective for this Training course is to present the AXMEDIS production chain, possibly starting from a comparative approach with the traditional one for content production. Focus will be on the process and how the various tools and solutions support the specific needs of each portion of the process. For this reason the course will interleave generic introductions with practical demonstrations.

4.2.2 Intended audience

This course is addressed to people who work in content factories, as new tools and mechanisms implemented in AXMEDIS will be presented to them.

At the end of this course/demonstration, attendees will understand:

- What is AXMEDIS – in terms of the overall Framework as well as the individual tools?
- Which are the main components and tools available for content production in AXMEDIS?
- How to use the main tools of the AXMEDIS suite?
- How to integrate AXMEDIS into current production workflows?
- Design implications for multi-channel delivery
- The benefits of adopting AXMEDIS in respect to traditional production methods

4.2.3 Course Content

The training course on AXMEDIS Production is structured as followed:

- Introductory note
- Overview of AXMEDIS production
 - Content classification, formats and constraints
 - Content production
 - Automated content production
- The AXMEDIS objects
 - Structure and evolution
 - Metadata
 - Complex object hierarchies
- Crawling Scenarios
- AXMEDIS Content Processing scenarios
- Interacting with the database, make a query
- The main AXMEDIS tools:

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- AXMEDIS Editor
- Integration with external editors and viewers
- The Main processing functionalities
 - Composition
 - Formatting
 - Adaptation
 - Protection
- Automating the process
 - Scripting and scheduling
 - Example rules and schedules
- AXMEDIS CP Scheduler and GRID
- Relationships with your database
- References and further info

4.2.4 Course Details

In the introductory note the tutorial will be presented with related objectives and expected outcomes for the target audience. Then an overview of AXMEDIS production will be provided. Such overview will start with a panoramic focusing on content classification, formats and constraints in the market in order to allow a better understanding of the differences among the various distribution channels. The traditional content production process will be described pointing out differences introduced by AXMEDIS so to lead to a better understanding of the automated content production. Then the AXMEDIS object structure and evolution will be presented along with related metadata, and some example hierarchical objects will be described in order to explain the concept of complex objects.

Given the fact that AXMEDIS operates in an already established market a section will be devoted to the crawling scenarios that are foreseen for the automatic feeding of content. Focus will then be moved to AXMEDIS content processing scenarios in order to explain how the presented tools and solution operate and support the user. Acting with the database will be the following theme of discussion, where how to make a query will be presented. The main tools will then be presented starting from the AXMEDIS Editor and its integration with external editor and viewers. This will allow to easily passing to the description of the Main processing functionalities: Composition, Formatting, Adaptation and Protection.

At this point the attendee will have a clear vision of the overall system and it will be possible to face how to automating the process, that is thanks to scripting and scheduling. Some example scripts will be worked through to demonstrate the capability. AXMEDIS CP Scheduler and GRID will be introduced to then address the relationships with users' database. Lastly references and further info will be provided.

4.3 AXMEDIS Training Course - AXMEDIS Distribution (TISCALI, EUTELSAT, ILABS, UNIVLEEDS)

4.3.1 Course objectives

This tutorial focuses on presenting the AXMEDIS framework for distributing content including the initial results and functionalities for integration and the use of AXEPTools. The aim of the course is to show to the tutees what happens after the production of AXMEDIS Objects: which are the different channels of distribution included in the AXMEDIS framework, and their peculiarities. This course will focus on content selection and retrieval, content aggregation and formatting, delivery channel selection, and distribution mainly in terms of procedural steps and activities to be performed to achieve the desired result.

4.3.2 Intended audience

The course is addressed to people interested in the distribution of content across multi-channels. Within AXMEDIS, the content can be formatted to support various media, before being sent to one of the several supported distribution channels/devices.

At the end of this course/demonstrations, attendees will understand all preliminary steps and see how to:

- search and select content, using the Query support and the AXEPTool,
- acquire and aggregate content basing own actions on PAR and licences,
- set up programme for publications for multi-channel distribution
- select a distribution channel and distribute content

Furthermore the attendee will understand and learn about the distribution process of AXMEDIS objects using the Programme and Publication Area and how to use the tools available such as the P&P Editor to utilise the functionality of P&P, and the On-Demand interface

4.3.3 Course Content

The training course on AXMEDIS Distribution is structured as followed:

- Brief overall introduction to AXMEDIS
- Searching and selecting content
- Selecting a distribution channel
- Content retrieval & fruition
- Examples of Distribution
 - Using the P&P Editor to create and edit programmes to deliver objects over various distribution channels (see Demonstration of Scenarios, Content Distribution and Supporting Tools, Programme and Publication, Scenario 1 later in the report)
 - “On-demand” request for AXMEDIS Object and how it utilises the P&P Area

4.3.4 Course Details

First of all, the currently available distribution channels for the AXMEDIS demonstrators are described with the importance of multi-channel distribution. This course presents details of the overall AXMEDIS system with respect to distribution of multimedia objects by providing the needed background info to properly understand terminology and relations among tools.

The tutorial will focus on the search and selection activities to be performed as initial step of the delivery and fruition process. This includes searching and selection with respect to the various distribution channels covering issues (directly or indirectly) coming from cross- / multi-channel distribution; satellite data broadcast showing the potentials of this distribution channel, regarding the B2B and the B2C environments; and dealing with content and retrieval issues examined from the distributor point of view (therefore mainly on constraints and related problems)

5 Additional AXMEDIS Training Courses and Plans (ALL)

With the development of the demonstrators, more detailed courses will be developed from the three main sections as described earlier, particularly in the production and distribution areas. This section presents the planning and organisation of these training courses to be developed in line with the development and integration of the AXMEDIS framework and tools.

5.1 Training Course: AXMEDIS Infrastructure and Architecture (DSI, EXITECH....)

Course objectives

The aim of the course is to provide general information about AXMEDIS architecture and its integration in several different productive and distribution contexts to the participant of the course. The main concepts of this particular course are to provide in-depth operative details to the technical experts on how to create, protect, and distribute the content using the AXMEDIS infrastructure and architecture.

Intended audience

This course is addressed to technical people interested in using AXMEDIS for the creation, protection and distribution of content in different delivery channels.

Course content

The training course on AXMEDIS Infrastructure and architecture will have the following structure and argument addressed:

- General overview of AXMEDIS goals
- AXMEDIS impacts on the value chain
 - Crawling
 - AXMEDIS Factory
 - AXPTool and AXMEDIS factory integration
 - Publication and distribution
 - AXMEDIS Certification and Verification
 - AXMEDIS Database
- The AXMEDIS Content Model
 - What is an AXMEDIS object?
- The main AXMEDIS Scenarios
- The application of AXMEDIS
- The distribution channels of AXMEDIS
- AXMEDIS editors and viewers

The responsible partner who leads the preparation and presentation of this course is DSI.

5.2 Training Course: User-Centred Product Creation and Validation (ACIT)

Course objectives

Participants from prospective user companies will learn to test and validate the AXMEDIS framework and applications in order to assess the benefits and added value for their domain and to generate further feedback about their needs and expectations. By the end of the course, attendees will understand:

- how to plan and manage user validation
- the major steps involved in planning and carrying out user validation
- how to select and apply appropriate methods and tools

As a compliment to the other courses on AXMEDIS framework and applications, this course will introduce to basic knowledge in user-centred product creation and provide hands-on training and experience in user validation and appropriate methods. The objectives of the course are:

- to empower decision makers to make early decisions about the adoption of AXMEDIS framework and applications based on the validation results.
- to ensure that managers are aware of usability quality and commit themselves to user validation
- to build key skills needed to plan the validation of the AXMEDIS framework and applications and of demonstrators within the AXMEDIS project, in Take-up-actions and in other, external projects

The course will be integrated closely with technical training, and will include coordination and the fostering of cooperation with other relevant FP6 projects.

Intended audience

The course will be relevant to potential users - technology/business-users as well as end-users. The main focus will be on users from SMEs. Decision makers, managers and technical experts will be introduced to the quality of use issues of the AXMEDIS framework and tools.

Course content

The course will consist of the following themes together with practical examples:

- Introduction to user-centred product creation
- User needs and requirements analysis techniques
- Usability inspection
- Approaches to user testing with focus on the evaluation of efficiency and effectiveness
- User satisfaction measurements
- Cost / benefit analysis

Responsible partner(s) who will prepare and present the course include ACIT and XIM.

5.3 Training Course: AXMEDIS Business (TISCALI...)

Course objectives

This course is to provide information and guidelines concerning business applications, scenarios and other business related issues. The responsible partner who will lead the preparation and presentation of this course is TISCALI.

Main goal will be to guide AXMEDIS actors on how to benefit in terms of business value by the use of the tools and models provided by the AXMEDIS framework.

Primarily the training will be focused on business model tools and market analysis instruments required in order to assess the economic opportunities provided by AXMEDIS.

Training sessions may provide the definition of business models and relevant Key Performance Indicators based on production, distribution of content as well as licensing and integration of applications. They may also include also business cases as analyzed within the AXMEDIS project.

Intended audience

This course is addressed to people conducting business activities in companies that may be interested in the use of AXMEDIS.

Course Content

In this activity several training material will be developed, including training sessions to the specified target users. The activity is planned to be replicated every 8 months. Training material developed will include:

- Manuals for describing of the different business models and business cases available
- Training sessions for:
 - o How to determine the KPIs in content production?
 - o How to determine the KPIs in content distribution?
 - o How to determine the KPIs in software licensing and integration?
 - o Which market analysis instruments are better suited for the market assessment in content production?
 - o Which market analysis instruments are better suited for the market assessment in content distribution?
 - o Which market analysis instruments are better suited for the market assessment in software licensing and integration?

The training material will evolve as more tools and business models are supported by AXMEDIS, based on the periodicity of this activity.

5.4 Training Courses for Decision Makers (DSI, EXITECH, TISCALI, XIM, ACIT, CRS4, FUPF, ILABS, EUTELSAT, FHGIGD, UNIVLEEDS, COMVERSE)

Intended audience

This course is aimed at Decision Makers, i.e. potential future buyers and adopters of AXMEDIS-derived tools, technologies and services. Decision Makers may have technical backgrounds, but may also be from creative, publishing or distribution business backgrounds, and hence the course will be designed to be of value to non-technical as well as technical Decision Makers.

Course objectives

The aim of the course for decision makers is to provide an understanding of the system architecture, hardware, software tools, procedures and the documentation for AXMEDIS. This course is designed to make potential users aware of the AXMEDIS applications from a user perspective.

Course content

This course will refer to the workflow and business process of the users and demonstrate the benefits and added value of AXMEDIS applications integrated into their domain. By taking practical examples with relevant 'hot topics' (for example, the challenge of competing with illegal P2P consumer downloads), the course will provide awareness of costs, risks, and benefits of adopting AXMEDIS.

The same Training courses as in the above section for Technical Experts will be used to create the Training courses for Decision Makers, with the additional focus on business related materials while reducing the technical coverage and explanation which will be provided by reference to the Technical Training courses. For Decision makers, the courses will include overview sections to discuss the benefits and values of the AXMEDIS applications, with a vision of new workflows, human resource requirements, business benefits and trends

Partners to be committed to setting up and maintaining tutorials include DSI, EXITECH, TISCALI, XIM, CRS4, FUPF, ILABS, EUTELSAT, FHGIGD, UNIVLEEDS, and COMVERSE.

5.5 Training Course: AXMEDIS Content (ILABS)

Course objectives

This course is focused on how to prepare and create the content rather than on the whole mechanism to make the content in automatic manner. This last point is addressed in the section above.

Intended audience

This course is addressed to content creators working in content factories.

Course content

The course will be based on the principles exposed in DE3.1.3 – AXMEDIS Content Aspects Specification, DE8.2.1 – Content Selection Guidelines, DE8.3.1 – Multilingual guidelines and technical solutions and DE8.4.1 – AXMEDIS Editorial Format Guidelines and basic examples.

The training course will have the following structure and argument addressed:

- An introduction to content aspect design

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Effective & efficient content design (covering readability, understandability and accessibility issues)
- Content classification & formats (encompassing text, images, audio, video, animations and multimedia)
- Authoring tools and content development applications
- Creating multimedia content from existing material exploiting subtitling and synchronization, audio compression, multi platform delivery and packaging features; covering also language related issues, accessibility and traceability
- Metadata at an overall level but also at a specific one especially for classification keywords and DRM specific metadata
- Constraints coming from kind of terminal, fruition mode, connection bandwidth, content readability and understandability
- Other issues including supported content and channel related content issues
- References

Exercises will be interleaved with training content and will be designed using content reported in DE8.1.1 – Content for Test Cases and Validation and available on the project website plus the info reported in DE8.5.1 – Collection of editorial formats and DRM rules for multi-channel that will help complete the process as well as cover the aspects of DRM and distribution-specialised editorial formats.

The responsible partner who will lead the preparation and presentation of this course is ILABS.

5.6 Training for Producing Content with the Automatic Tools (UNIVLEEDS)

This training is to be managed by UNIVLEEDS during M19 to M48 under WP7.5. It is to include the following areas: content production, content modelling, formatting modelling, insertion of technical metadata, exploitation of metadata, usability training, definition of new formats, definition of new content production styles, set up and customise the process for content production on-demand, with the automatic tools.

In this subWP, UNIVLEEDS will collaborate with all the relevant partners to develop the training materials and established relations with relevant user organisations to promote this training. Up-to-date AXMEDIS prototypes and modules will be collected and studies in order to create a set of training materials. These materials aim to provide the awareness and understanding of AXMEDIS tools and technologies, and to provide training to relevant users on how to use them. This activity is planned to be iteratively enhanced at 6 monthly intervals.

5.7 Training on Protection Aspects and DRM rules definition (FUPF, FHGIGD)

This training is to be managed by FUPF during the period of M19 to M48 under WP7.4.

This activity will cope with the description of how DRM rules can be used for the definition of business models for different kinds of content, distribution channel, etc.

They include the definition of licensing models for different distribution channels, different rights of usage and conditions associated to them, tracking of the actions licenced, and, in general, how we can define DRM rules over content that is later on monitored and finally exploited. The use of the DRM model at all levels of the distribution chain, but specially on the end user device and tools will be a critical issue.

In this activity several training material will be developed, including training sessions to the specified target users. The activity is planned to be replicated every 8 months. Training material developed will include:

- Manuals for describing the definition of DRM rules
- Training sessions for:
 - o Explaining how to define business models using DRM rules
 - o Explaining how to use tools implemented for the definition of DRM rules

- Manuals for the training sessions

The training material will evolve as more tools and business models are supported by the DRM rules, based on the periodicity of this activity.

5.8 Training Courses on Installations, Configurations, Maintenances and Managements (DSI, EXITECH, TISCALI, ...)

Additionally, with the development and integration of the AXMEDIS framework, the following training courses will be developed for future requirements:

- Training on System Installation and Configuration (DSI, EXITECH, ...)
- Training on System Maintenance and Management (DSI, EXITECH, ...)
- Training on Working with the AXMEDIS System (DSI, ALL, ...)
- Training on Content Management and AXMEDIS compliant CMS (TISCALI, EXITECH, DSI, ...)

6 Demonstration Activities (UNIVLEEDS, ALL)

With the collaboration of all partners, demonstration activities of AXMEDIS is under development during M12 to M48 with the WP7. This section discusses the set-up and methods for delivering the activities for demonstrating the AXMEDIS framework and architecture, tools, functionalities and scenarios.

This can be divided into the following schema:

- Content Model and Production Overview
 - AXMEDIS Model (DSI)
 - Slides presentation
 - AXMEDIS editor/authoring/renderer (DSI, SEJER)
 - AXMEDIS Content Processing (DSI)
 - Content Gathering (FOCUSEEK)
 - AXMEDIS Database (EXITECH)
- Advanced Content Production Aspects
 - SMIL and Style (EPFL, DSI)
 - Adaptation (FHGIGD, DIPITA, EPFL, etc)
 - Fingerprint (FHGIGD, DIPITA, EPFL, etc)
 - Workflow (IRC, HP, XIM)
- Content Distribution and Supporting Tools
 - AXEPTool (CRS4)
 - Integrating B2B with the AXMEDIS Factory (CRS4, EXITECH)
 - Programme and Publication (UNIVLEEDS)
 - Query on Demand (FHGIGD)
- Content Protection Tools
 - DRM Tools (FUPF)
 - Legal Aspects (AFI)
 - Protection Processor (DSI)
 - Certifier and Supervisor (DSI)
 - Accounting and Reporting (EXITECH)
- Distribution examples
 - Distribution to PC (TISCALI, SEJECT)
 - Distribution to Mobiles (COMVERSE)
 - Distribution via Kiosks (ILABS)
 - Distribution via Satellite Data Broadcast (EUTELSAT)

6.1 Content Model and Production Overview

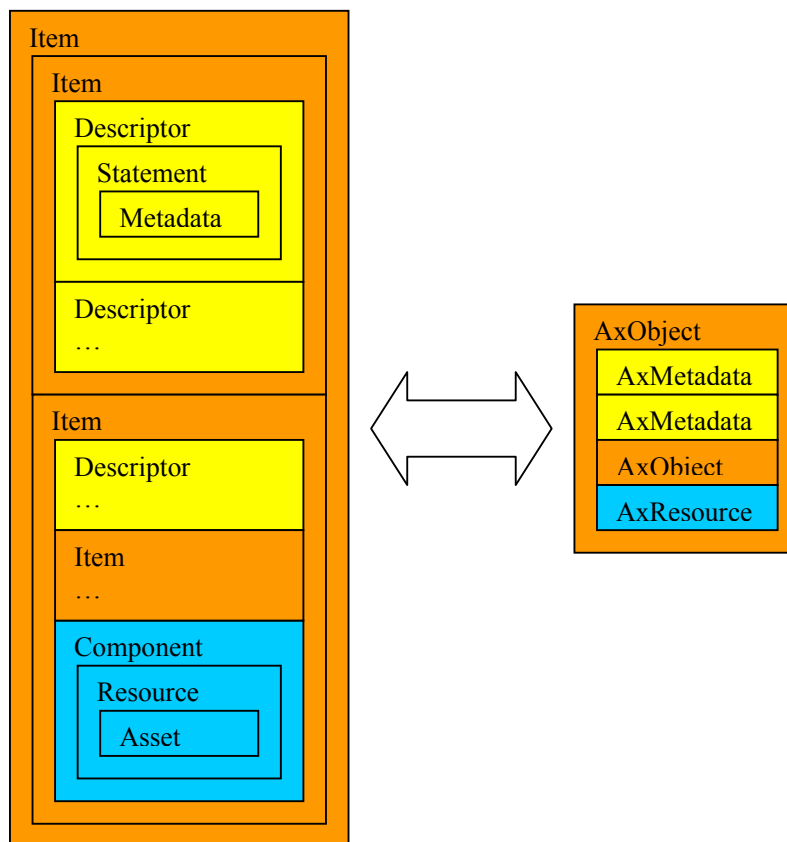
6.1.1 AXMEDIS Model (DSI)

The AXMEDIS Model is the part of the AXMEDIS project which provides:

- specification on how AXMEDIS objects has to be modelled and
- a software library to securely and trustworthily manipulate objects.

In particular, the AXMEDIS object model is a profiling of the MPEG21 Digital Item Declaration (DID) standard. That is, an AXMEDIS object is an MPEG21 digital item having a given structure which satisfies the requirements gathered by the partners in terms of simplicity and flexibility of the model and needed metadata for the trading of digital resource.

MPEG21 digital items are basically structured documents. In particular, the structure is a tree-like structure. A digital items is composed of several types of element. Each type has a given and specific semantic which allows to give a common structure and meaning to all digital items. Therefore, also AXMEDIS objects have a tree-like structure. This structure is simpler than the MPEG21 one as depicted in the figure below.

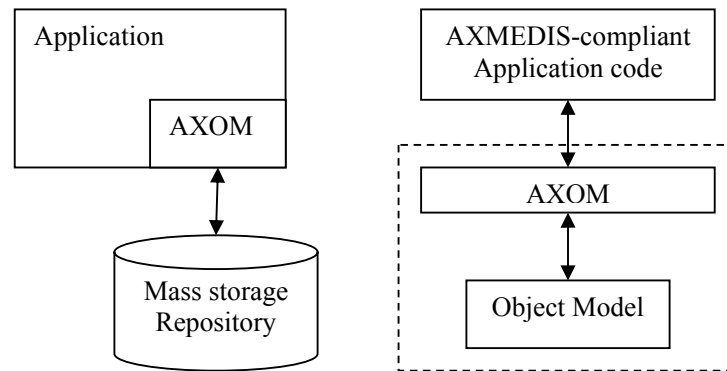


In the figure above, each colour identifies related elements between the two hierarchies.

The library, called AMEDIS Object Manager (AXOM), provides tools to create an object model of MPEG21 documents and to manipulate it. Upon this MPEG21 object model, the AXOM transparently (for the user) creates an AXMEDIS object model, i.e. a simplified model (with respect to the MPEG21 model) which exposes specific functionalities studied for AXMEDIS objects.

Moreover, the AXOM has been designed to create an indirection layer between the application and AXMEDIS objects, i.e. an application which uses the AXOM to access to an AXMEDIS object cannot directly manage the object. Instead, the application makes its requests to the AXOM which is in charge of filtering these requests w.r.t. the rights owned by the user. In that way, the AXOM makes the application a

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools
 trusted environment where protected resources are manipulated only if the needed rights have been obtained.
 The picture below depicts how application, AXOM and document model interacts together.



It is impossible to contextualise the AXMEDIS Model because its software implementation (i.e. the AXOM) is used by almost all tools of the AXMEDIS framework and it has to be used by all the application which would like to manipulate AXMEDIS objects. For the same reasons, there are not specific scenarios or demonstrator for the AXMEDIS Model because it is involved in all scenarios and demonstrators of the AXMEDIS project.

Instead, the AXMEDIS Model is better described in the training course “Working with the AXMEDIS System”. In this course, how to use the AXOM to create AXMEDIS-complaint application is explained through practical programming examples (see the related section).

6.1.2 AXMEDIS editor/authoring/renderer (DSI, SEJER)

The demonstration of AXMEDIS editor, authoring and render involves the manual creation of AXMEDIS objects. Normally creation of AXMEDIS Objects will be done using the AXMEDIS Content Processing tools allowing mass production of AXMEDIS objects, however AXMEDIS editor can be used to:

- Create prototype objects;
- Inspect the objects created using automatic tools;
- Finalise objects.

6.1.2.1 Contextualisation of AXMEDIS editor/authoring/render

AXMEDIS Editor is used in the AXMEDIS Factory for the manual production of objects, inspection of automatically created objects and AXMEDIS Objects finalization. The AXMEDIS Editor activation can be done through the *Workflow Management tools* allowing to schedule the manual operations to be performed for the objects production. The AXMEDIS Editor interacts with the *AXMEDIS Database* to get AXMEDIS Objects from the database operate on it (i.e. add a new resource, edit metadata, edit licence, use content processing tools, etc. etc.) and upload the new version in the database.

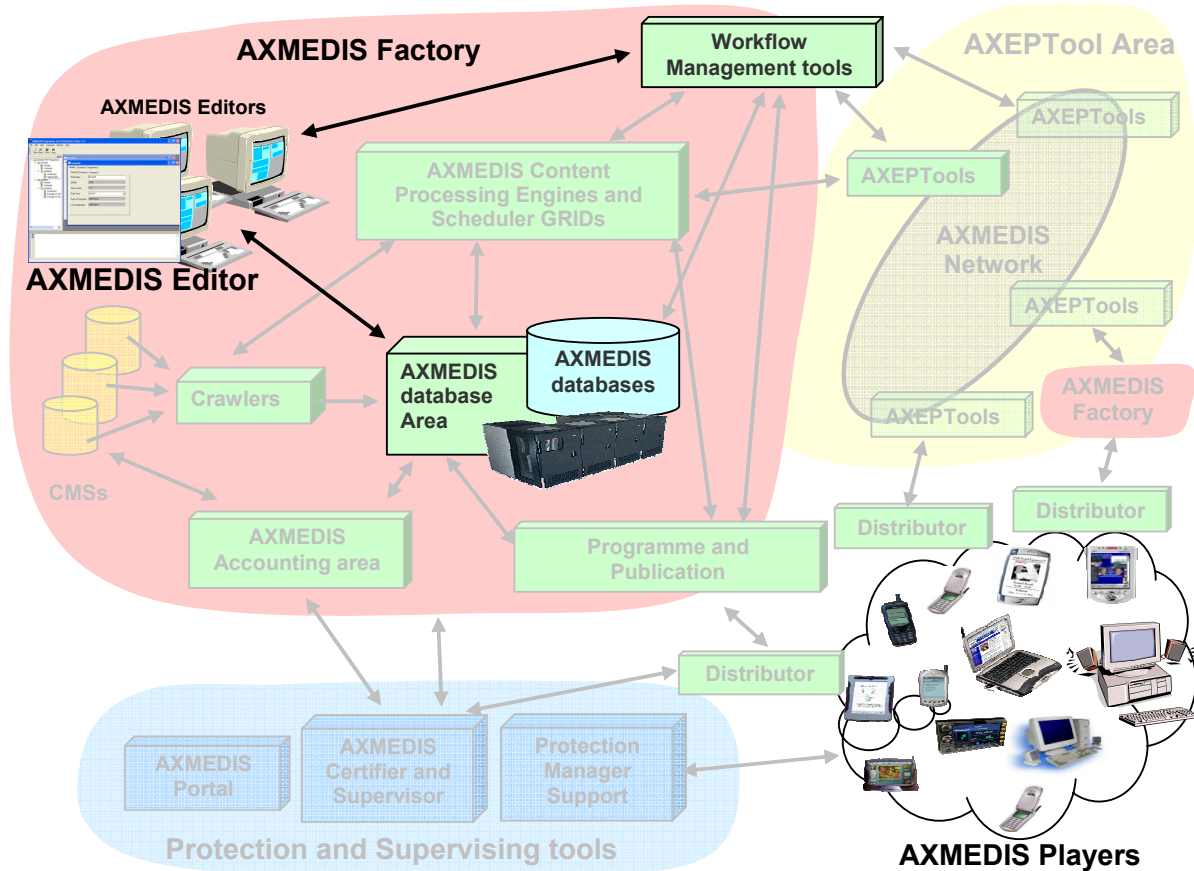


Figure: AXMEDIS Architecture highlighting the AXMEDIS Editors Area

6.1.2.2 Demonstration of AMEDIS Editor Scenarios

Scenario 1: Creation of a Basic AXMEDIS Object

1. The actor creates a new object
2. The actor adds a new resource file (audio, video, document)
3. The actor edits the metadata associated with the resource (Dublin Core)
4. The actor edits the Potential Available Rights (licence) using the DRM editor
5. The actor uploads the object in the AXMEDIS Database

Scenario 1a: Creation of a Basic AXMEDIS Object controlled by the Workflow Manager

The same activities of Scenario 1 are handled by different persons

1. Actor 1 connects to the Workflow Manager which opens the AXMEDIS Editor with a new object
2. Actor 1 adds a new resource
3. Actor 1 uploads the object in the Database
4. Actor 2 connects to the WF Manager which opens the AXMEDIS Editor on the metadata editor
5. Actor 2 adds metadata for the object
6. Actor 2 uploads the object in the Database
7. Actor 3 connects to the WF Manager which opens the AXMEDIS Editor on the DRM editor
8. Actor 3 adds potential available rights for the object
9. Actor 3 uploads the object in the Database

Scenario 2: Creation of a thumbnail

1. The actor opens an AXMEDIS Object containing an image from the File system
2. The actor invokes a content processing plugin to produce a reduced version of the image

3. The actor saves the object on File system

Scenario 3: Creation of a Composed AXMEDIS Object

1. The actor creates a new AXMEDIS object
2. The actor makes a query on the Database to find basic AXMEDIS Objects
3. The actor adds references to the selected objects
4. The actor reorders the objects in a meaningful way
5. The actor creates a SMIL presentation using the Visual Editor (optional)
6. The actor adds metadata
7. The actor uploads the object in the Database

Scenario 4: AXMEDIS Object fruition

1. The actor opens an AXMEDIS Object saved on the File System with the AXMEDIS Player or with the Mozilla Plugin

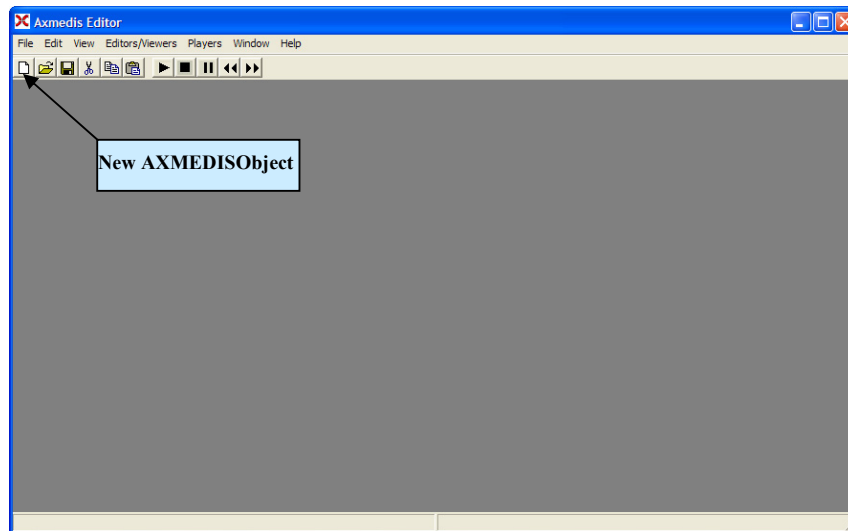
6.1.2.3 Demonstration of AXMEDIS Editor Tools

Scenario 1: Creation of a Basic AXMEDIS Object

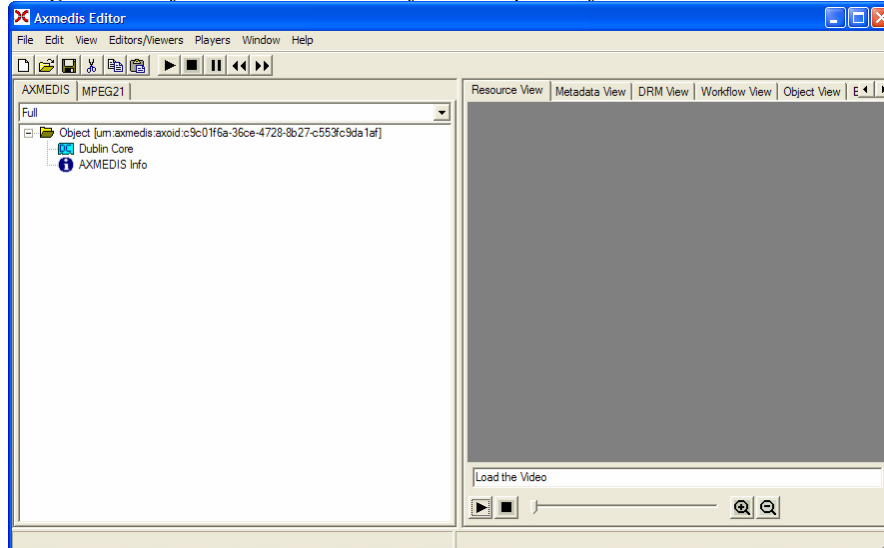
A Content Owner wishes to create a Basic AXMEDIS Object containing a digital resource (audio, video, document, etc.) that he/she wants to sell over the P2P B2B network for the integration within other multimedia content or he/she wants to use inside his/her own multimedia content.

Creating an Object

To create an AXMEDIS Object, the user starts the AXMEDIS Editor and from the opening screen creates a new Object by selecting “New” from the tool bar or using the File menu (File → New) or using the keyboard short cut “Ctrl-N”.

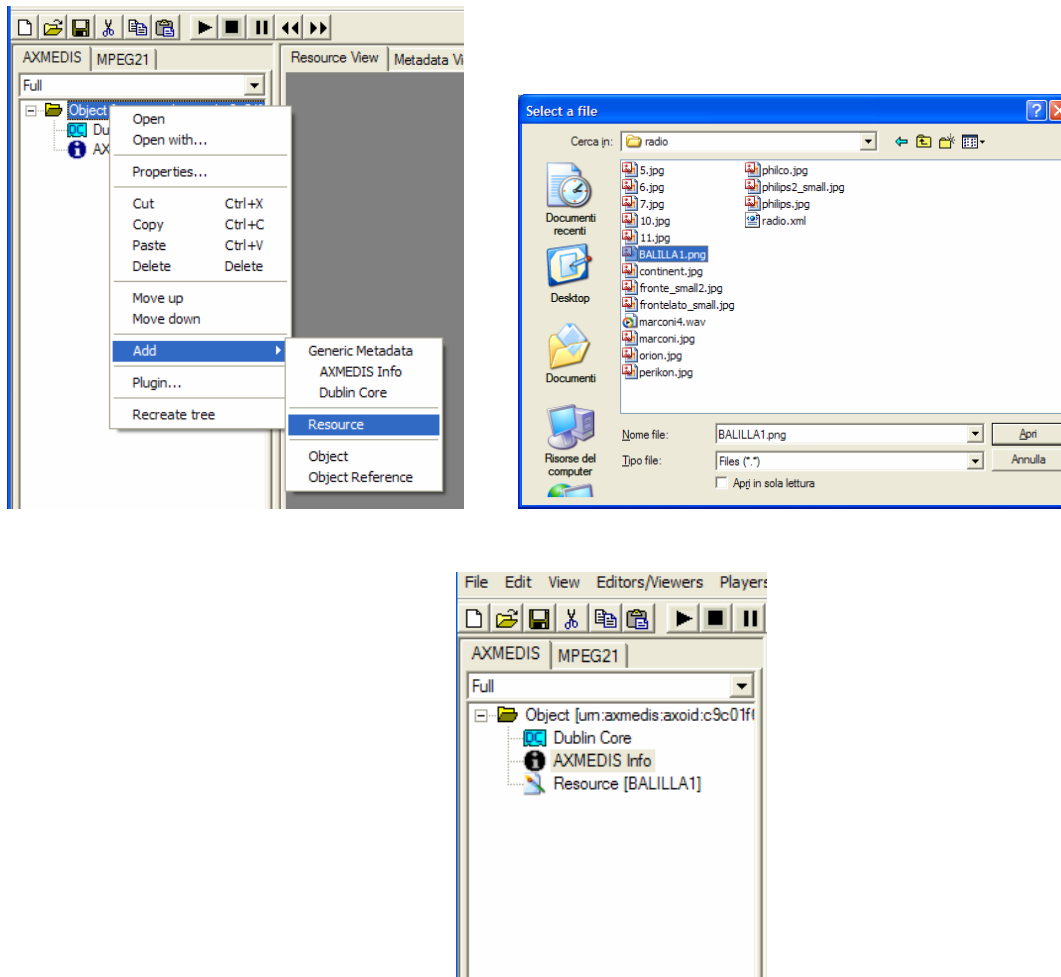


After the object has been created, the new object is ready for editing. On the left side a ‘tree view’ shows the structure of the object and on the right side different views of the object are present.

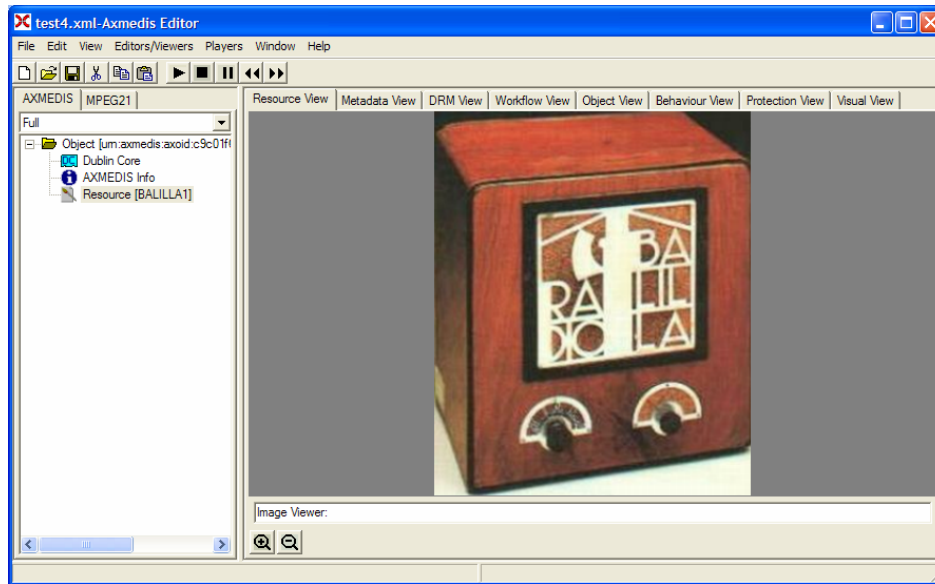


Adding a new resource

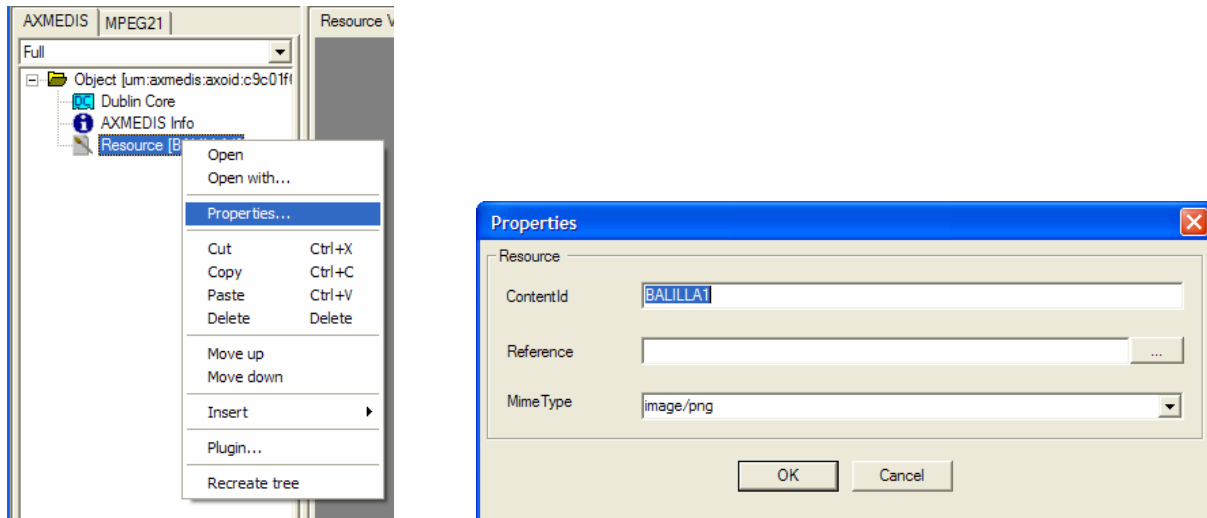
A new resource can be added simply dragging the file resource/s on the object hierarchy or using the context menu. In the last case a dialog let the user select the resource to add.



Double clicking on the resource the resource viewer shows the resource content.

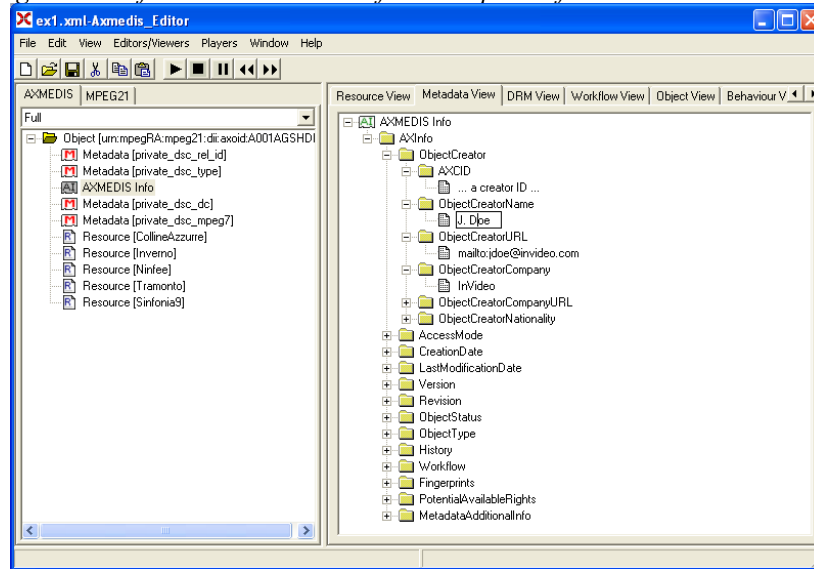


The properties of the resource can be edited using the context menu on the resource item in the hierarchy



Editing metadata

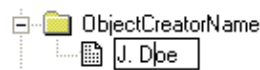
Double clicking on the metadata (AXMEDIS Info, Dublin Core or a generic metadata) opens the tree like view of the MPEG-21 standard metadata. The Dublin Core metadata item can also be edited by right clicking and bring up a menu to add and remove elements and a dialog box to edit the values and attributes of the metadata editor.



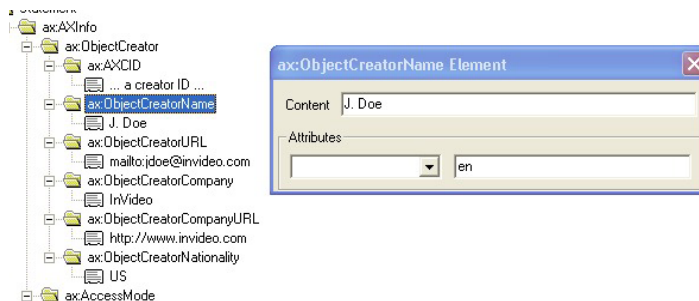
Inaccessible metadata return a dialog box specifying the XML is not accessible in this Metadata.



The metadata view includes the Metadata Elements (shown as folders) and the Element values (shown as a page icon). Editing the metadata element is a process of clicking in the value and editing the node as shown below.

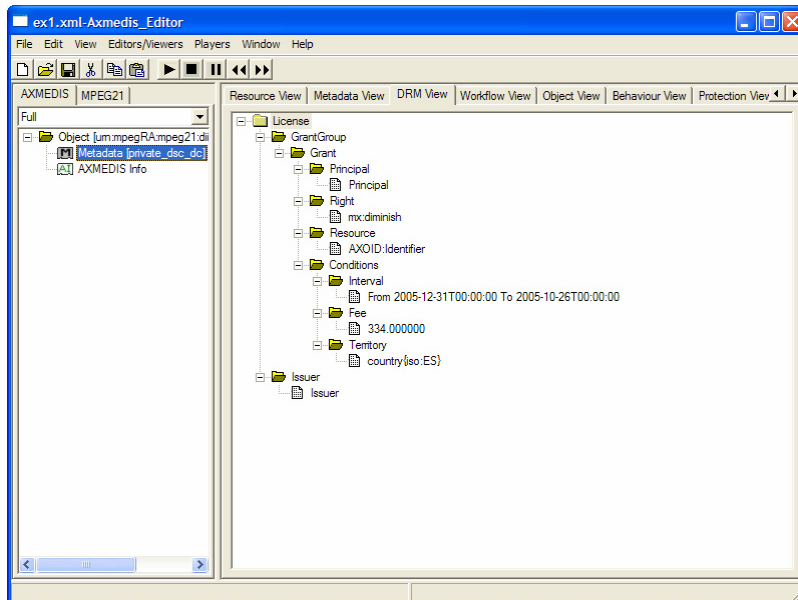


Double clicking on a tree folder in the metadata tree view opens a dialog box that can be used to again set the element value, as shown as a page icon in the tree view, and attributes for the metadata element selected.



Editing Potential Available Rights

Using the "Open With..." item in the context menu associated with an object and selecting the DRM View, the DRW editor & viewer is opened.



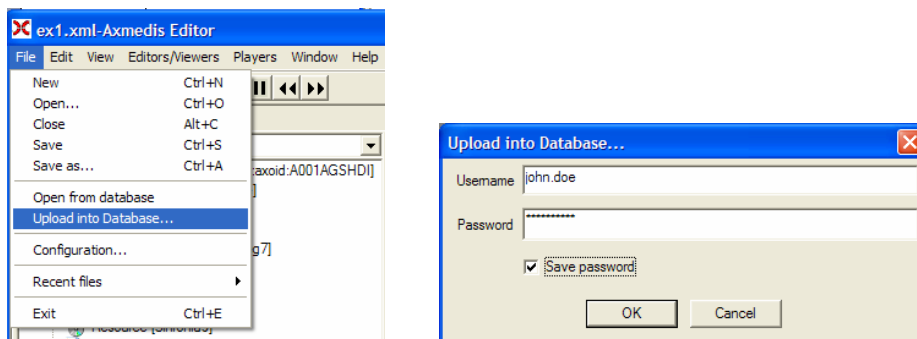
In the DRM editor/viewer the user can view and modify the Potential Available Rights (PAR) associated with the object.

In the figure above, the structure of Potential Available Rights can be seen in a tree-like shape. PAR's follow the structure of an MPEG-21 REL licence, like licences used in AXMEDIS, but they can have some empty fields that will be filled when a real licence for a final user or distributor is created.

For a complete tutorial on how to use content protection tools, and specifically DRM tools, refer to section 6.4, Content Protection Tools.

Uploading on the Database

The user can upload the object created into the database using the "File/Upload into Database" menu item, it asks for the user credentials for upload and then the upload is performed.



Scenario 2 Creation of a Thumbnail

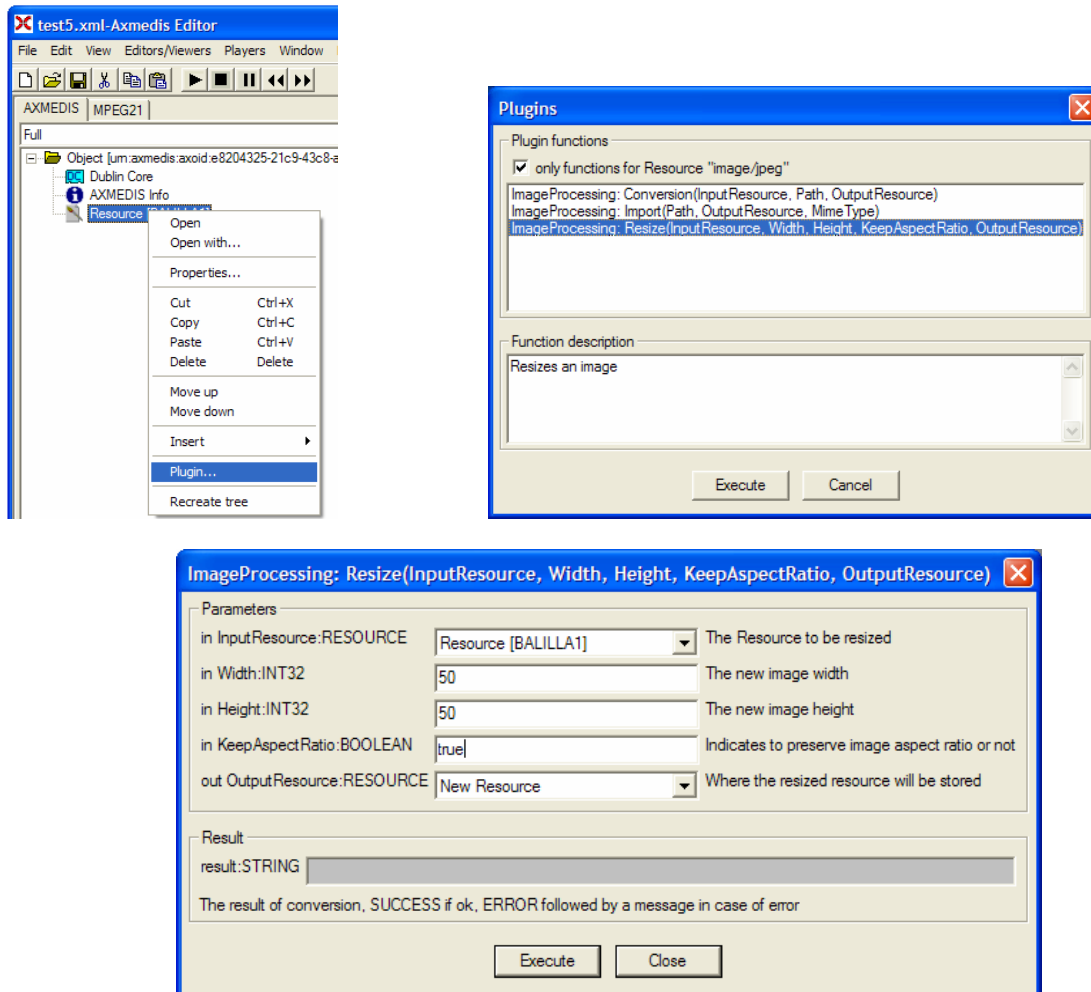
The content producer wants to add to the object a thumbnail of an image resource inside the object.

Opening an object

An object can be loaded from file system using the Open... menu item. It will ask for a file to be loaded.

Calling a content processing plugin

To call a content processing plugin on a specific resource the context menu item "Plugin..." can be used. It will present the list of plugin functions that can be used for the specific resource. Selecting the plugin function and then pressing the Execute button a dialog to get the parameters needed by the plugin is presented. The user can enter the value for the parameters, the user can select the resources to be used, the width and height of the new image, can specify to preserve aspect ratio and then pressing the Execute button the plugin is executed and the result shown in the result text box.

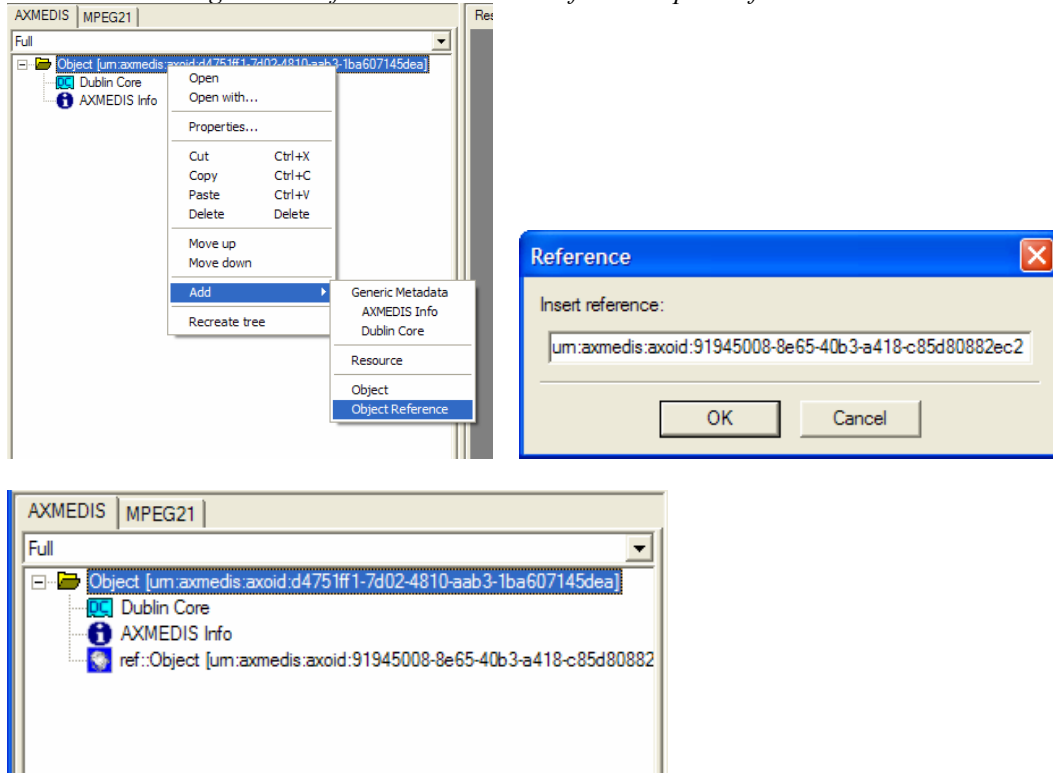


Scenario 3 Creation of a composed AXMEDIS object

The creation of a composed AXMEDIS object is similar to the creation of a basic AXMEDIS object rather than add a resource the user adds an object or a reference to an object. Currently the user may add an object reference using the AXOID or he/she may add a new object inside the composed object.

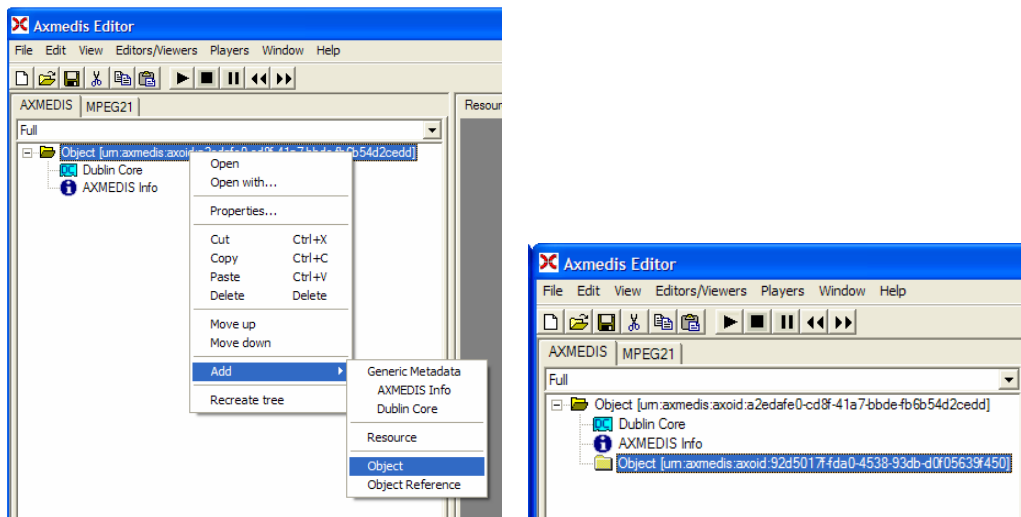
Adding an Object Reference

To add an Object Reference the context menu can be used, which currently asks for an AXMEDIS AXOID



Adding a new Object inside another object

To add a new object inside another object the context menu can be used which adds a new object and then the object may be populated with resources, metadata or other object.



WARNING: this part is very draft it will be realized using the query support to search for objects in the database, at the time of writing the query support user interface has not been integrated.

6.1.3 AXMEDIS Content Processing (DSI)

The demonstration of the AXMEDIS Content Processing Area involves two sections, the first demonstration is of the AXCP Rule Editor to create, edit, save and activate the AXCP Rule; and the second is of the AXCP Rule Engine to execute content processing rule (composition and formatting, protection, adaptation, and perform “On-demand” processing.

Introduction

The AXMEDIS Content Processing Area has been designed to provide a set of digital content processing tools to aid content designers to:

- efficiently collect the components needed for producing composed objects, using a direct access and query to the database;
- find/produce alternatives/adaptations for the components that present potential distribution problems (too big files);
- structure the components, highlighting the relations among them;
- bind the structure of content and content collection to some presentation and formatting styles;
- format/adapt content for a variety of channels,
- protect digital content according to MPEG-21 models and related AXMEDIS tools, including licence production;
- process and produce metadata and managing the estimation of fingerprint and descriptors;
- import from other sources (e.g., existing CMSs) content and metadata to be used for the production process.

To this end, the AXMEDIS Content Processing Area is governed by executable rules, called *AXCP Rules*. These Rules are used to script/program the activities to be performed for producing, processing and protecting digital contents in automatic and secure manner.

More complete and refined mechanisms used for the content manipulation and analysis directly enforced in the AXCP Rules are used for creating more realistic and interesting objects to be marketed at both B2B and/or B2C levels. At B2B level, the produced objects can be a coarse collection of resources and information, while those that have to be made available at Consumer level need to be better processed in order to define the layout and distribution aspects.

The Content Processing Area provides support for the protection of AXMEDIS objects and the licence generation and verification. In this context, the Content Protection includes:

- Applying Protection to AXMEDIS object: encryption, scrambling, compression, fingerprint and creation of new Protection Information
- Generating a licence from licence model and additional information.
- Invocation of some verification algorithms about licences and available rights to simulate the usage from the user site.

Once the AXMEDIS objects are created, they are stored into the database that can be used for the distribution. The tools provided are also capable to incorporate the content and the related metadata (images, video, audio, documents, etc.), already available in the Content Management Systems (CMS) of the factory. The metadata coming from the CMS has to be adapted for the creation in the AXMEDIS Database of the basic assets (AXMEDIS objects) to be used for content processing

6.1.3.1 Contextualisation of the AXMEDIS Content Processing Area

The AXMEDIS Content Processing Area sits in the AXMEDIS Factory (as shown and highlighted in the following figure). The creation of AXCP Rule to specify the content processing activity can be created using the AXCP Rule Editor. This editor creates an AXCP Rule that when activated specifies when, how and where to deliver an AXMEDIS Object to the AXCP Rule Engine using the processing engines, workflow and the database area in the AXMEDIS factory. The same functionalities in the AXMEDIS Content

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools
 Processing Area can be used when requesting content processing activity On-Demand such adaptation, protection and licence generation request for different type of digital resources.

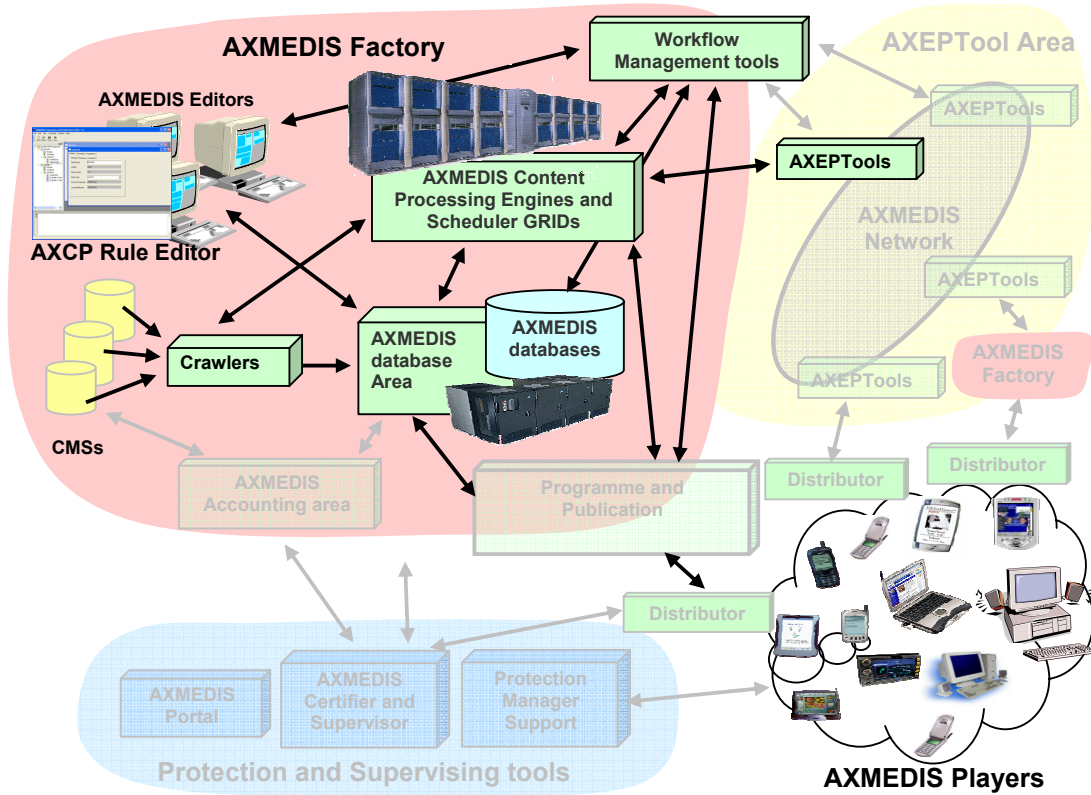
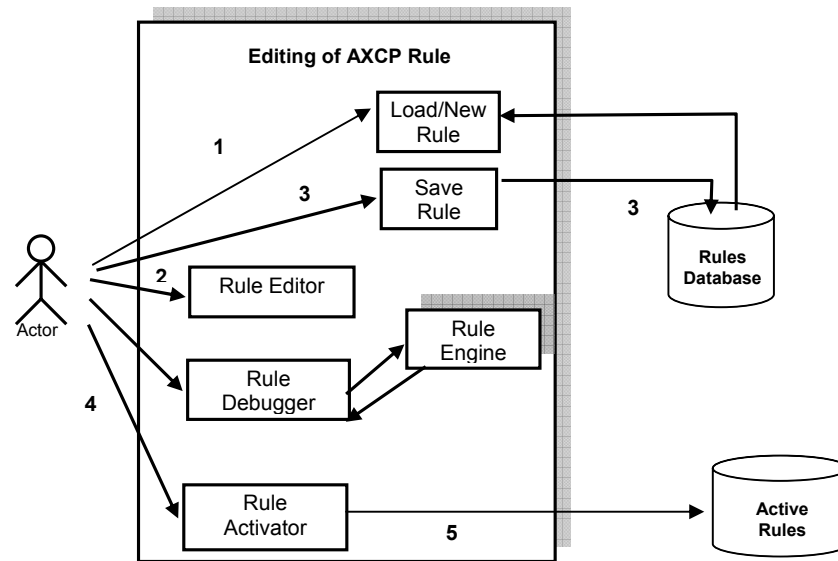


Figure: AXMEDIS Architecture highlighting the AXMEDIS Content Processing Area where the AXCP Rule Engine is continuously running and the AXCP Rule Editor for creating and editing AXCP Rule.

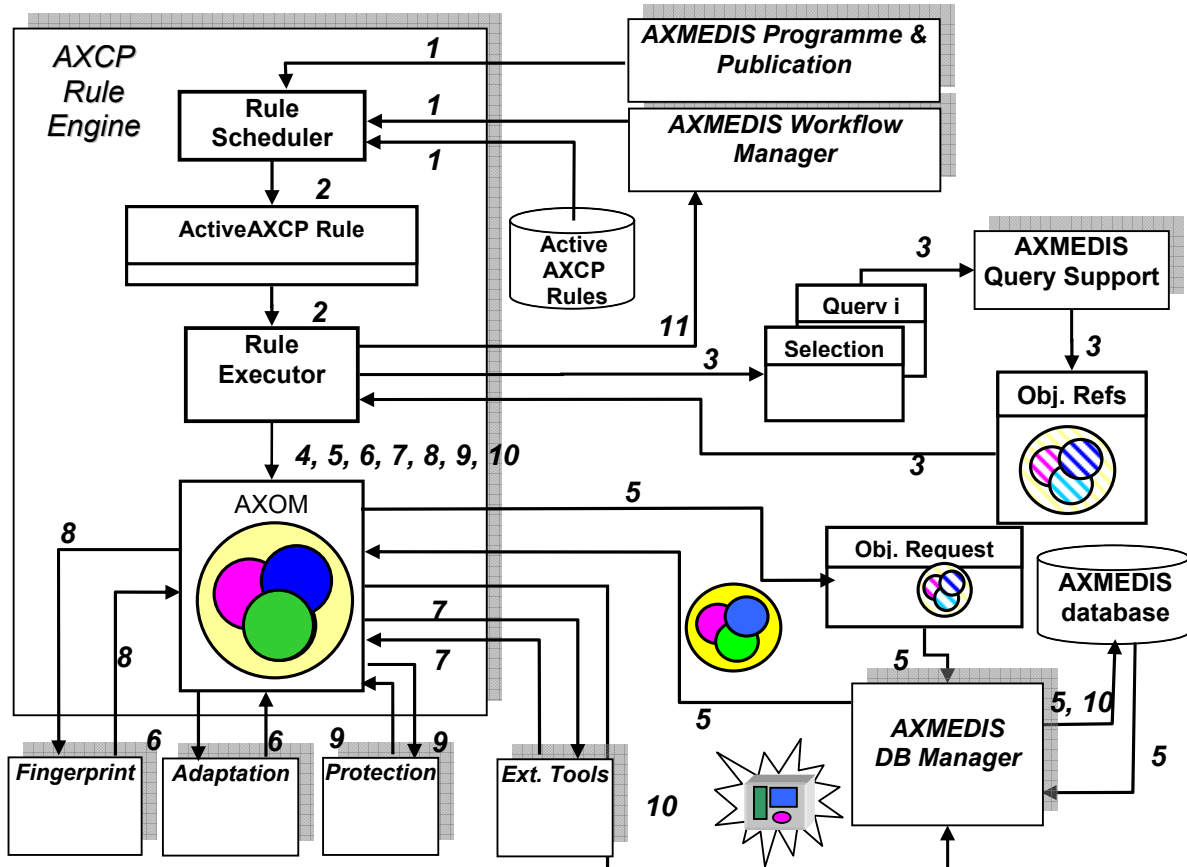
6.1.3.2 Demonstration of AXMEDIS Content Processing Area Scenarios



Scenario on Editing of AXCP Rule

Scenario description for the activation and modification of an existing formatting rule:

1. The actor loads an existing AXCP rule from the rules database or starts with a new rule
2. The actor edits the rule by the AXCP Rule Editor
3. The actor saves the new rule into the rules database
4. The actor activate the rule
5. The rule is sent to rules repository (AXCP Rule Engine).



Scenario on Automatic Content Processing – Rule Engine

Scenario description:

1. **Start process.** The AXCP Rule Engine receives a running rule request coming from the AXMEDIS Workflow Manager, or from the AXMEDIS Publication & Programme, or from the internal scheduler that activates an AXCP rule.
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.
4. **AXOM usage.** An embedding object request with the relative object reference is sent to the AXOM to perform the inclusion.
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 perform a formatting paradigm or a set of customised formatting parameters. This phase could:

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- a. Perform adaptation algorithm (change resolution, change dimension, time or spatial best fitting, etc...)
 - b. Apply spatial and temporal constraints specified in the rule (i.e. graphical layout, temporisation, transitions effects, etc...)
 - c. Perform synchronisation algorithm (audio and text audio and images, video and text, etc...)
 - d. Convert the whole formatted object into a specific output format (i.e. MPEG4).
7. **External tools calling.** This request allows calling external functionalities available on external formatting tools. In this way some formatting operation can be delegated and performed in other formatting environment. These calls are properly coded in the formatting rule and refer to portion of code written by using for example the script language available on the external tool.
 8. **Fingerprint request.** This request is performed via AXOM in order to apply the fingerprint to the formatted object according to the fingerprint parameters specified in the rule.
 9. **Protection request.** This request is performed via AXOM. A protection request is sent to the Protection tool in order to apply protection to the new AXEMDIS formatted object.
 10. **Storing AXMEDIS object.** The new formatted AXMEDIS Object is stored into the AXMEDIS Database.
 11. **End process notification.** The End of the formatting process is notified to the AXMEDIS Workflow Manager.

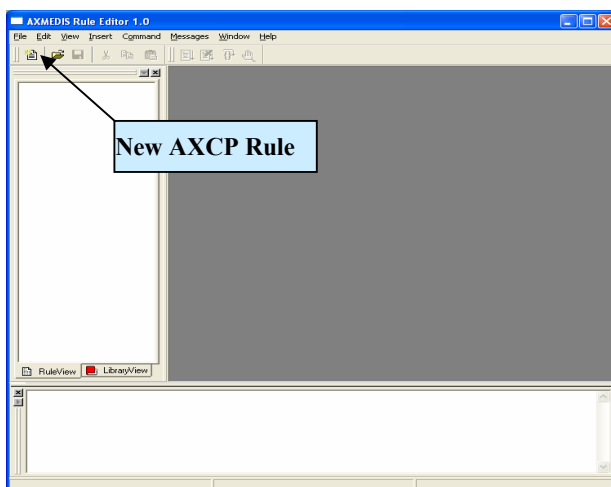
6.1.3.3 Demonstration of AXMEDIS Content Processing Area Tools

Scenario 1: Editing on AXCP Rule – AXCP Rule Editor

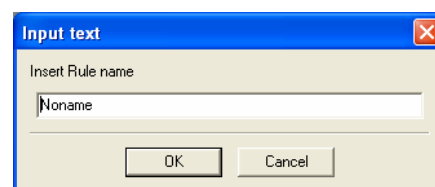
A content producer or manager wishes to create a content processing rule called AXCP Rule for manipulating/creating AXMEDIS multimedia objects. The AXCP Rule Editor is the editor for writing AXCP Rule. The AXCP Rule Editor GUI is a MDI window that manages a rule document at a time. The editor provides a set of tools and views to help the user during the editing and building of rule, writing JavaScript code. By opening the AXCP Rule Editor the user can create a rule to specify the time of firing. Once a rule has been created, the user can save it for re-editing at a future time or test the rule. Testing the rule enables the rule to be checked and debugged for any arising problems when executing the Javascript code and allow the user within the editor to make changes to successfully activating the rule.

Creating a new rule

To create a rule, the user starts the AXCP Rule Editor and from the opening screen creates a new Rule by selecting “New” from the tool bar or using the File menu (File → New) or using the keyboard shortcut “Ctrl+N” (see Screenshot 1). On requesting a new rule, the dialog box pops up and the user enters the name of the draft AXCP Rule and selects “OK” (see Screenshot 2).

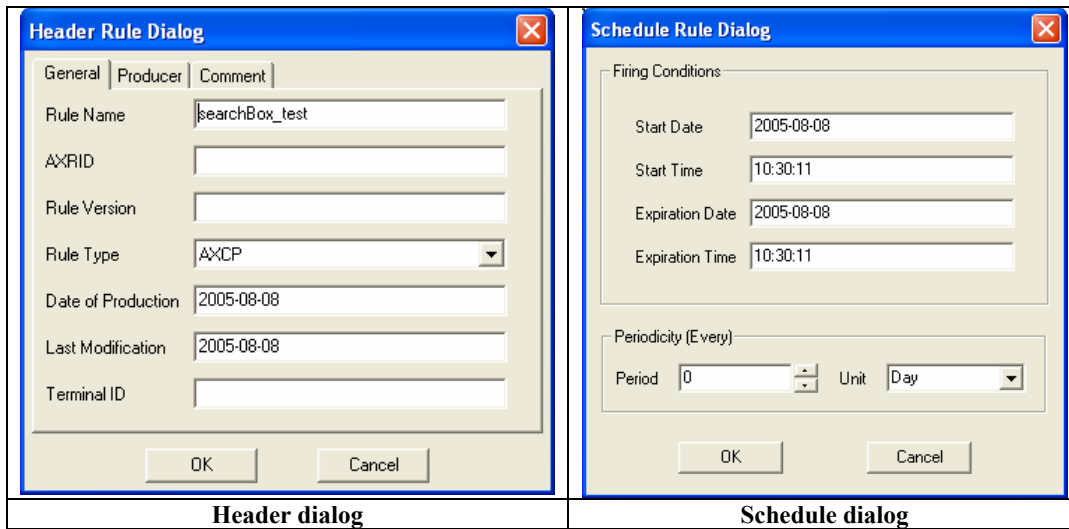


Screenshot 1 - Open a new rule

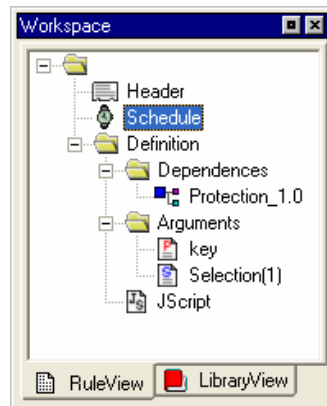


Screenshot 2 – Input rule name dialog

After the rule name has been entered, the new rule is ready for editing with the ‘tree view’ used as a workspace and the editing dialogs for editing the rule details and making a rule schedule (see Screenshot 3).



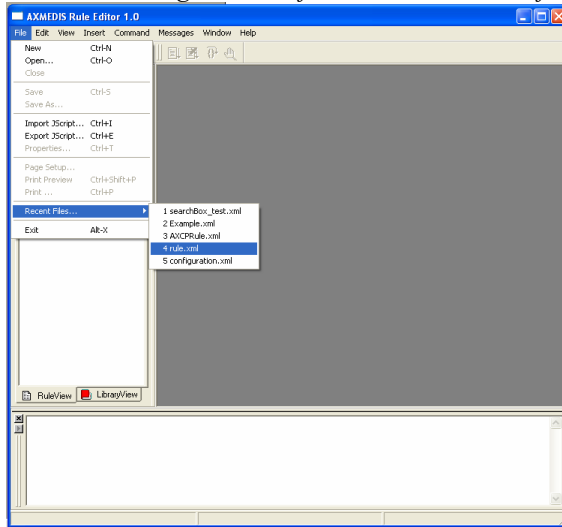
- Workspace Area is a resizable docking panel and includes a notebook control where the rule view is shown. Such view is a tree view. A dynamic popup menu is available for a quick access to functions that allows the quick management of items (edit and view metadata, delete, Insert, Cancel, Move up/Down, Rename, Open/Edit, ...). Appropriate icons allow identifying intuitively components of rule in view. In the following picture the actual structure of the Rule View area is depicted:



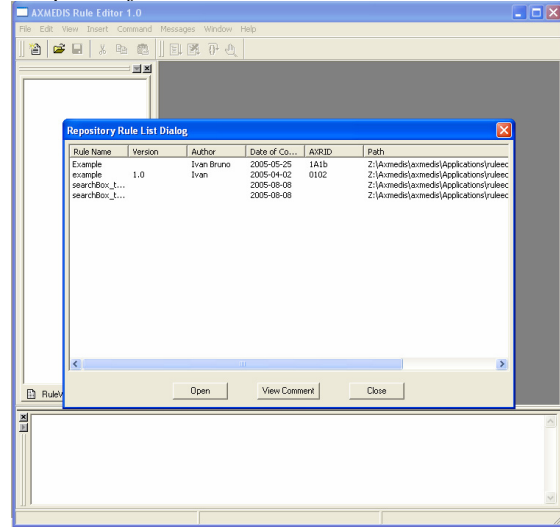
Screenshot 3

Loading an existing rule

Instead of creating a new rule, the user may wish to edit an existing draft rule. The user may have saved it as a file somewhere or saved it in the AXCP Rule Repository. By selecting ‘open’ or ‘Rule List’, as seen in screenshot 4 and 5, the programme manager can load an existing draft rule from the Repository or elsewhere on the system.



Screenshot 4

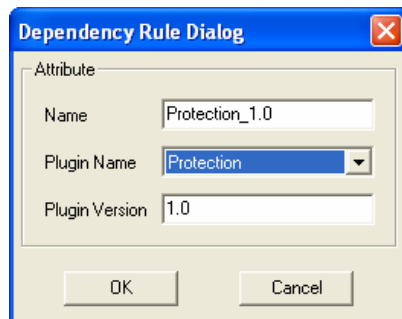


Screenshot 5

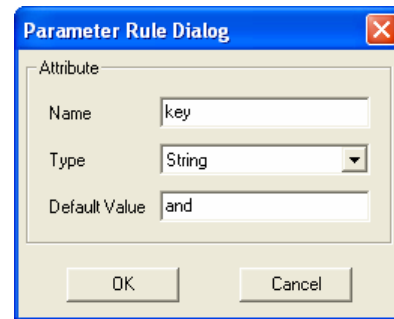
Editing a rule

The user can use the set of dialog and text editor to edit the AXCP rule (as shown in screenshot 6), and edit rule data such as:

- Parameter dialog for editing the attributes of a rule parameter
- Dependency dialog for editing the attributes of a AXMEDIS PlugIn
- XML Selection Editor (XML viewer/editor for the XML representation of selections) and JavaScript Editor based on Scintilla Editor for text/javascript code editing. It provides full editing capabilities (copy, cut, paste, redo, undo, syntax highlighting, etc...), print preview, page setup and print functionalities, syntax highlighting, brace highlighting, folding/hiding of lines, breakpoint insertion/removal, visualisation of line numbers



Dependency dialog

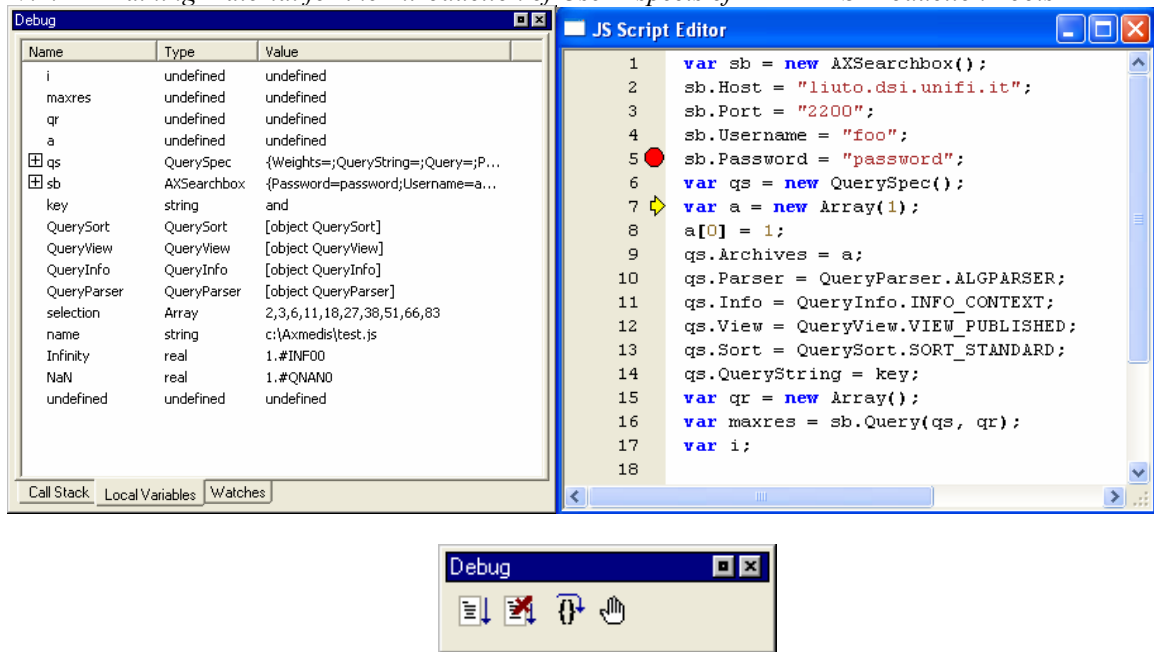


Parameter dialog

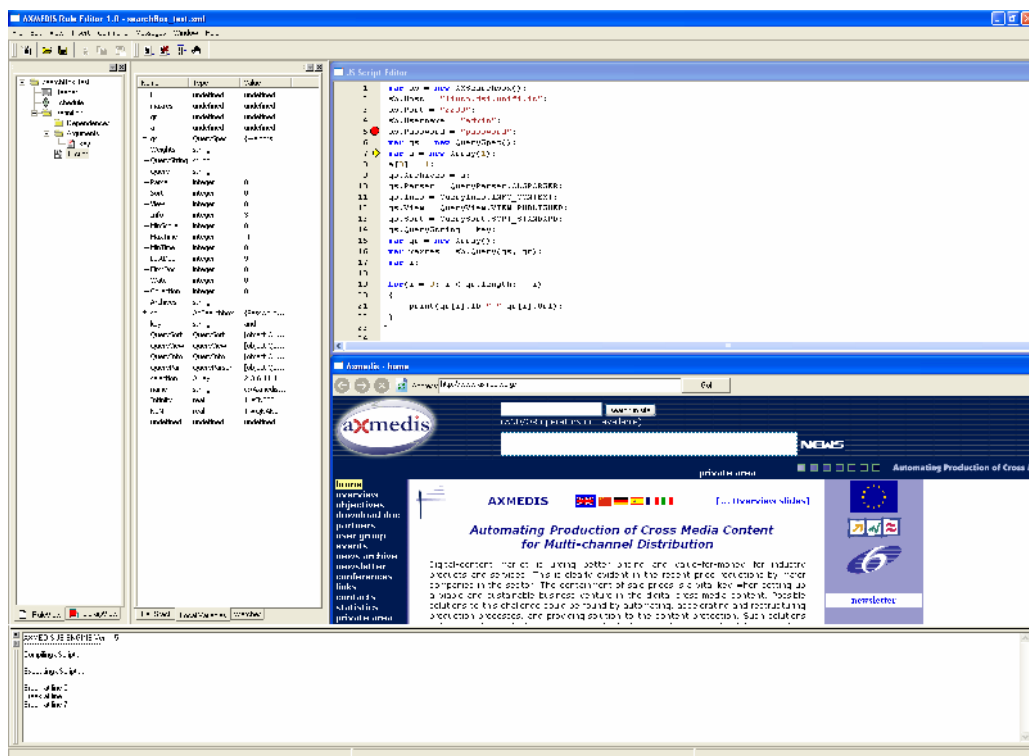
Screenshot 6

Debugging Rule functionalities

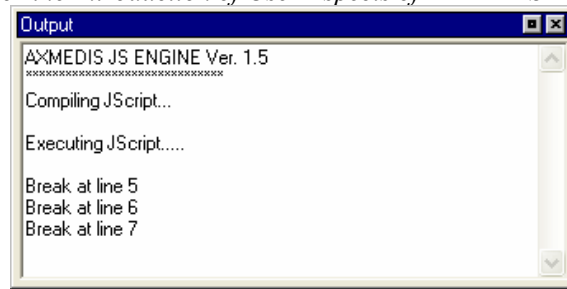
The editor provides functions to add/remove breakpoints (F9), start debug (F5), next breakpoint (F5), step over (F10), stack calls monitoring, local variables visualization. In the Print output window textual output will provide internal errors or communication when script runs and monitoring the debug session



Screenshot 7 - Main navigation interface of a Debug session



Screenshot 8 - Main view of the AXMEDIS Rule Editor GUI



Activating a rule

The current version of the prototype does not allow installing a rule in the rule engine or notifying the completeness of the rule to the workflow manager. The activation will be done manually in the scenario 2.

Conclusion

Using the AXCP Rule Editor, the user can write an AXCP Rule and edit it to set when and how the digital contents have to be processed. Once checked, the activated rule is sent to the AXCP Rule Engine which uses AXMEDIS tools to run the javascript that describes the content processing procedure.

Learning Outcomes

At the end of this tutorial session, the user will understand:

- How the AXMEDIS Content Processing Area fits within the AXMEDIS architecture
- How to use the AXCP Rule Editor to write/check rule and javascript code.
- An user will be able to create a new rule and edit an existing rule;
- Test/debug a rule

Scenario2: Automatic Content Processing – AXCP Rule Engine

The AXCP Rule engine was divided in two main components:

- **Rule Scheduler (Server Side)** – It consists of the an internal Scheduler and Dispatcher. It performs the operations of rule installation, rule firing, rule executor discovering and management, rules dispatching, communication with the AXMEDIS environment, etc....
- **Rule Remote Executor (Client Side)** – It is the executor of rules and consists of a script engine based on JavaScript (JS) SpiderMonkey released by Mozilla. It runs the JavaScript code associated with rule.

The Scheduler GUI is the main window that allows interacting with the Scheduler. Referring to the snapshot, it is constituted of:

1. A menu bar
2. Two main areas where the list of rules and the list of remote executors are displayed.
3. A status bar where the current clock and the current date are displayed.

Rule Name	AXRID	Rule Version	Rule Status	Job ID	Executor ID	Start Time	Start Date	Periodicity	Number of Runs
searchBox_t...			completed	9	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	10	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	11	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	12	-1	16:05:11	09/23/05	0	1
searchBox_t...			running	13	2	16:05:11	09/23/05	0	0
searchBox_t...			completed	14	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	15	-1	16:05:11	09/23/05	0	1
searchBox_t...			running	16	3	16:05:11	09/23/05	0	0
searchBox_t...			completed	17	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	18	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	19	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	20	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	21	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	22	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	23	-1	16:05:11	09/23/05	0	1
searchBox_t...			running	24	8	16:05:11	09/23/05	0	0
searchBox_t...			completed	25	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	26	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	27	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	28	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	29	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	30	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	31	-1	16:05:11	09/23/05	0	1
searchBox_t...			completed	32	-1	16:05:11	09/23/05	0	1
searchBox_t...			running	33	7	16:05:11	09/23/05	0	0
searchBox_t...			completed	34	-1	16:05:11	09/23/05	0	1
searchBox_t...			running	35	9	16:05:11	09/23/05	0	0
searchBox_t...			running	36	6	16:05:11	09/23/05	0	0
searchBox_t...			delayed	37	-1	16:05:11	09/23/05	0	0
searchBox_t...			delayed	38	-1	16:05:11	09/23/05	0	0
searchBox_t...			delayed	39	-1	16:05:11	09/23/05	0	0
searchBox_t...			delayed	40	-1	16:05:11	09/23/05	0	0

Executor Name	IP	CPU	Clock	OS	Transfer Rate	HD Space	Status	Rule ID	Executor ID	Workload p...	Start Time	End Ti
DISIT-01	192.168.0.197	intel	1800	Windows N...	-1	1073741824	busy	2	1	0.000000	15:04:38	15:04
DISIT-04	192.168.0.105	intel	1800	Windows N...	-1	452984832	busy	13	2	0.000000	15:04:05	15:04
DISIT-03	192.168.0.52	intel	1800	Windows N...	-1	89128960	busy	16	3	0.000000	15:27:33	15:27
DISIT-02	192.168.0.43	intel	1800	Windows N...	-1	84934656	busy	5	4	0.000000	15:45:09	15:45
MIRKOFANI	192.168.0.64	intel	1800	Windows N...	-1	1073741824	busy	6	5	0.000000	15:53:06	15:53
VENOM-WORK	192.168.0.103	intel	1800	Windows N...	-1	2147483648	busy	36	6	0.000000	16:01:28	16:01
M386	192.168.0.49	intel	1800	Windows N...	-1	2147483648	busy	33	7	0.000000	16:05:11	16:05
DISIT-05	192.168.0.102	intel	1800	Windows N...	-1	0	busy	24	8	0.000000	16:25:49	16:25
HOMER	192.168.0.101	intel	1800	Windows N...	-1	2147483648	busy	35	9	0.000000	16:35:48	16:35

Menu bar – It provides the access to the following set of implemented functions:

1. Program

- Add rule* – Load a rule in the scheduler
- Launch scheduler* - Start the scheduler activity.
- Stop scheduler* - Stop the scheduler activity.
- Restore* - Backup Copy of the last jobs list.
- Minimize* - It reduces at icon on the taskbar.
- Exit* - Close the application.
- Start Grid Peer functions* – It starts the grid peer network communication support

2. Settings

- Preferences* - Open an editable dialog with the set of configuration parameters.

3. View

- Refresh* – Update the list of jobs and list of remote executors.
- Arrange* – Repainting modes of tables in the main frame
 - Top* – It shows only the top table (Table of rules)
 - Bottom* – It shows only the bottom table (Table of executors)
 - Vertical* – It shows tables vertically
 - Horizontal* – It shows tables horizontally
- Rule Properties...* - Open a Rule Properties dialog.
- Executor Profile...* - Open an Executor Profile dialog.
- Logs...* - Open a dialog to show the list of log messages

4. Commands

- Activate Rule* - Put in the “ACTIVE” status the current selected inactive rule.
- Deactivate Rule* - Put in the “INACTIVE” status the current selected active rule.
- Kill Rule* - Kill the current execution of the current selected rule.
- Pause Rule* - Put in pause the execution the current selected rule.
- Resume Rule* - Resume the execution of the current selected rule.
- Remove Rule* – Remove the rule from the list of rules

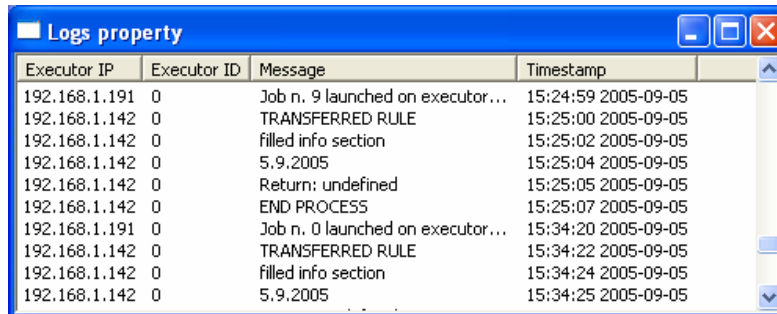
- g. *Suspend Rule...* - Open a dialog to edit the temporal interval for rule resuming and then suspend the current selected rule.

5. ?

- a. *Help* - Open the On Line help.
- b. *About* - Open a dialog with credits.

Logs Dialog

This dialog allows viewing the logs of scheduler activity.



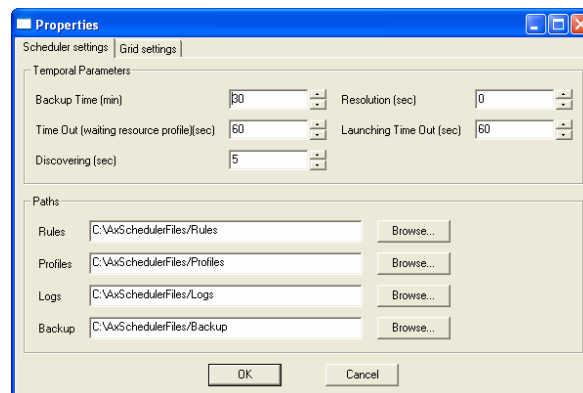
Executor IP	Executor ID	Message	Timestamp
192.168.1.191	0	Job n. 9 launched on executor...	15:24:59 2005-09-05
192.168.1.142	0	TRANSFERRED RULE	15:25:00 2005-09-05
192.168.1.142	0	filled info section	15:25:02 2005-09-05
192.168.1.142	0	5.9.2005	15:25:04 2005-09-05
192.168.1.142	0	Return: undefined	15:25:05 2005-09-05
192.168.1.142	0	END PROCESS	15:25:07 2005-09-05
192.168.1.191	0	Job n. 0 launched on executor...	15:34:20 2005-09-05
192.168.1.142	0	TRANSFERRED RULE	15:34:22 2005-09-05
192.168.1.142	0	filled info section	15:34:24 2005-09-05
192.168.1.142	0	5.9.2005	15:34:25 2005-09-05

Properties Dialog

It is a tabbed dialog that allows editing settings parameters regarding the scheduler activity (*Scheduler settings*) and the GRID support (*GRID settings*).

Scheduler settings – It consists of a set of configuration parameters contains settings about:

- **Backup Time** - Backup interval for logging the set of submitted rule and tracing operations. It is expressed in minutes.
- **Time Out** - Time out on client activity. It is expressed in seconds.
- **Time Resolution** - Time Resolution of the scheduler. It is expressed in seconds.
- **Refresh Time** - Time Resolution for discovering new rule executors
- **Rules Path** - Rule Repository Path
- **Log Path** - Log Repository Path
- **Profile Path** - Executor Profile Repository Path
- **Backup Path** – The path where the scheduler periodically saves the current rules list.



Properties

Scheduler settings | Grid settings

Temporal Parameters

Backup Time (min) [30] Resolution (sec) [0]

Time Out (waiting resource profile)(sec) [60] Launching Time Out (sec) [60]

Discovering (sec) [5]

Paths

Rules: C:\VaxSchedulerFiles\Rules [Browse...]

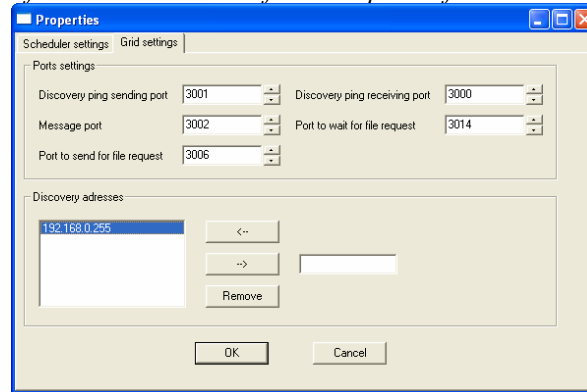
Profiles: C:\VaxSchedulerFiles\Profiles [Browse...]

Logs: C:\VaxSchedulerFiles\Logs [Browse...]

Backup: C:\VaxSchedulerFiles\Backup [Browse...]

OK Cancel

Grid settings – It provides a set of settings to setup the communication support. It allows to define the number of ports to use when receiving file, messages, sending files, responding to the discovering request. It allows also to define IPs of LANs to use when the scheduler performs the discovering of peers.



Activating and stopping a rule

When the user is happy with his AXCP Rule and validated it with a quick and/or full trial, the final option is to activate the AXCP Rule in the AXCP Rule Engine. In the current version of the prototype the user has to install manually the AXCP in the Scheduler by selecting “Add rule” in Program menu. The scheduler will process the rule information and at the specified times will distribute the rule to a rule executor. During the running of the rule, the user can also stop it from the Scheduler by selecting “Kill Rule” in the Command menu.

Conclusion

Using the AXCP Rule Scheduler, the user can manually setup the AXCP Rule Engine and install an AXCP Rule in the AXCP Rule Engine. The user can also monitor the whole activity of the engine by means of logs.

Learning Outcomes

At the end of this tutorial session, the user will understand:

- How the AXCP Rule Engine fits within the AXMEDIS architecture
- How to use the AXCP Rule Scheduler to mount rule in the AXCP Rule engine.
- Controlling and Monitoring the scheduler
- Stop the execution of a rule.

6.1.4 Content Gathering (FOCUSEEK)

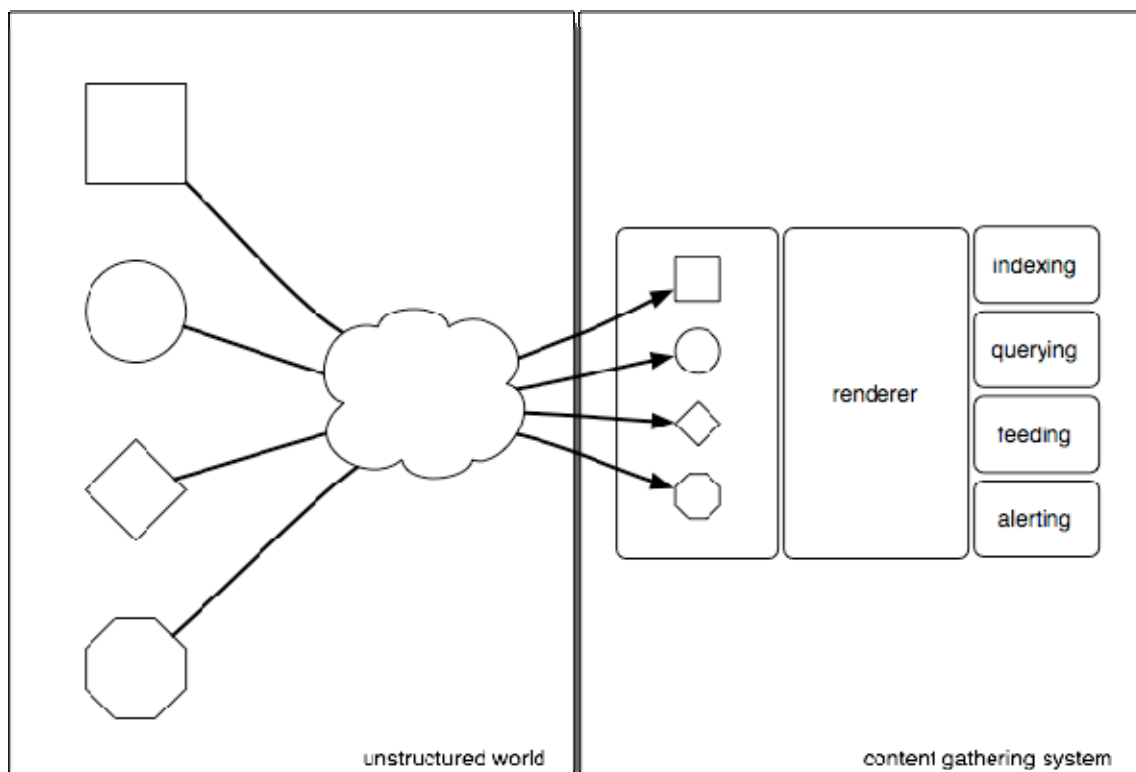
Gathering data from original sources is one of the main problems in digital content integration and delivery. A very typical scenario is to gather information from many, heterogeneous digital sources that are geographically distributed too. Owners of such digitals sources are focused on their original purpose of content production and usually do not provide a standard way to access their archives by other applications. This situation is due to many factors. Basically, information is the core business and concerns of a content provider and he/she normally demands strict control on how it is delivered. As a result, content providers do not really concern if the user would like to use other applications to access their information through standard protocols and formats. This situation is not ideal from the point of view of the user who has many content providers to interface with because he/she is forced to setup and maintain a custom communication with each channel. Such channels are characterised by customised user interfaces and are often very difficult to be integrated with other applications. A possible solution arises from a custom declination of the approach that is currently used by Search Engines for Web and the Web Service technology.

Web Search Engines cannot influence in any way how web sites publish their information so that if an engine wants to build an index of the content provided by some site it must access the web site on his own. The method used by Search Engines to accomplish that task is called “crawling” or “spidering”. A web crawler is a software agent that simulates a real user accessing a web site and read all the information contained in it. In order to succeed with this task a crawler must have a toolbox with any possible “adapter”

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

able to match all access protocols and document formats available on the web. Soon after its formation, the WWW began to support other protocols in addition to the original HTTP and many other document formats other than HTML. Formats like PDF, DOC, Flash and protocols like NNTP, FTP and ODBC (some of which actually predate the HTTP over HTML web standard medium) forced Web crawlers to adapt themselves to the new situation. The basic assumption of a typical Web crawler is that any information source must be treated like a “black box” with no way to contact the webmaster to ask him/her to adapt content for a specific usage. From the Web source point of view a Web crawler is like any other normal user that visits the site. This particular approach is very powerful because it has zero organisational and technical impact on the information sources and for this reason it has been successfully adopted in the enterprise environment too. In any large company or public administration the goal of aggregating content from different and heterogeneous sources (even if they are located and managed by the company itself), is really hard to be accomplished. Exporting data from an existing database means that either or both the organisations providing and using the content has to obtain the necessary authorisations, writing some software and thus allocate some human resources. All those reasons are serious potential point of failure for any content integration project. In this type of scenario a crawling technology can enormously simplify the integration task because the crawler acts exactly like any other authorized user whose accessing procedures are already defined and accepted by all departments of any company.

An interesting way to visualise the content gathering problem is to imagine that in order to acquire information we have to setup a channel connecting the content provider and the users. Using the already discussed “search engine” approach a possible solution is to create a system able to aggregate many different information sources and provide some standard application services to access it. In this way users will only need to know the standard application interface provided by the gathering system.



At the left side of the above picture the heterogeneous world of content provider is sketched. Different shapes represent the different protocols and formats used to access to the content. At the opposite side there is a structured repository that needs to be filled from contents coming from content provider. The middle component is the content gathering module which chooses the right adapter to gather information from any content provider and exposes some standard services:

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- **The Indexing/Querying Service.** Is able to retrieve any piece of information in the repository through a query composed by words or metadata separated by the AND, OR, NOT and NEAR operators typical of any search engine. The indexing is implemented using a full dynamic indexing service in order to take in account when a new content is added to the repository. No index rebuild is needed.
- **The Feeding Service.** Used to automatically feed newly acquired contents through a channel. A very common standard like RSS can be used for this purpose.
- **The Alerting Service.** Generates events to notify that something is changed in the repository. Alerting methods use email messages, Instant Messaging, SMS and Web Service calls.

The above services can be used by a client side component to build any kind of structured object based on the original “raw” information gathered from content providers. Obviously any type of structure provided by the content provider itself will be preserved and indexed too.

6.1.4.1 Contextualisation of the focuseek gathering system

The searchbox platform is used by the AXMEDIS system to gather information from Content Management System of various types.

6.1.4.2 Demonstration of gathering Scenarios

In this section we'll analyze the case of gathering from various types of Seeds, in particular:

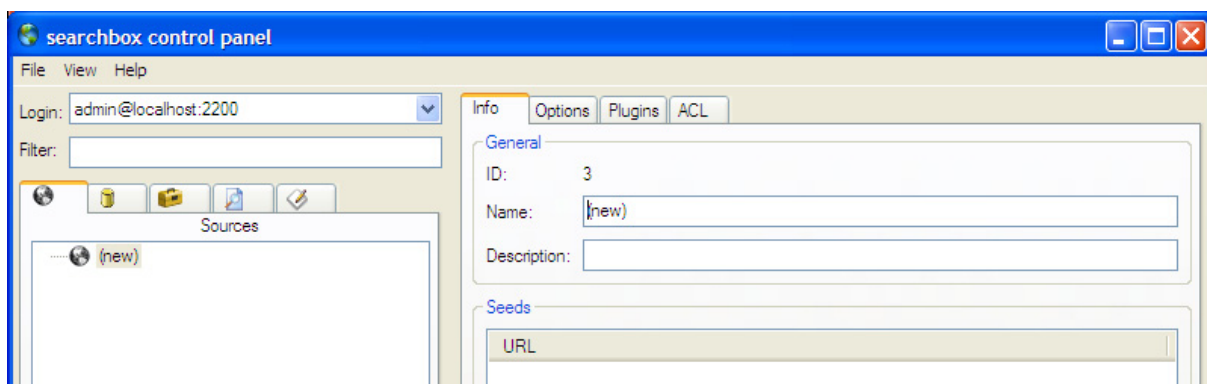
- Gathering from a Web site
- Gathering from a FTP site
- Gathering from a Usenet site
- Gathering from a WebDav share
- Gathering from SMB share
- Gathering from an ODBC

All this scenarios are managed by the searchbox Administration Tool (Control Panel) by adding a new Source containing the Seed of specified type. See next section for details.

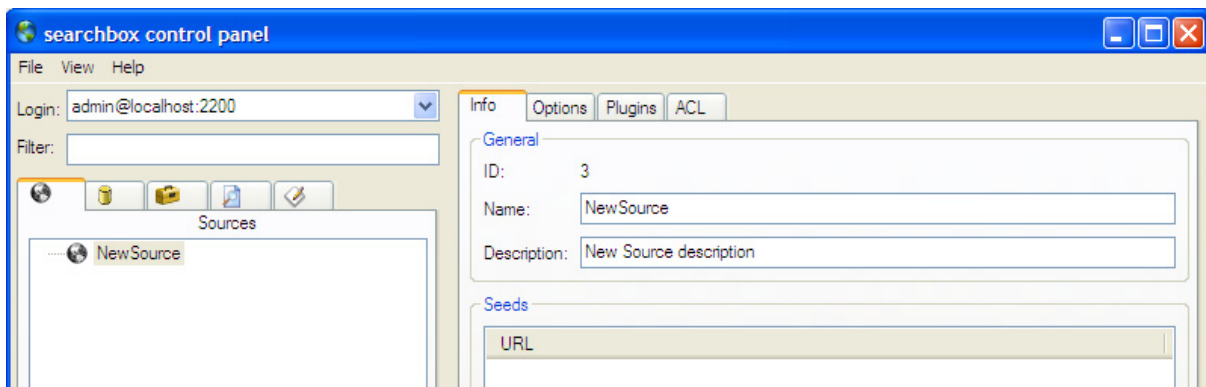
6.1.4.3 Demonstration of gathering Tools

In this section the process of adding a new Source using searchbox Administration Tool (Control Panel) is shown.

Clicking on a new Source button the following empty source is shown

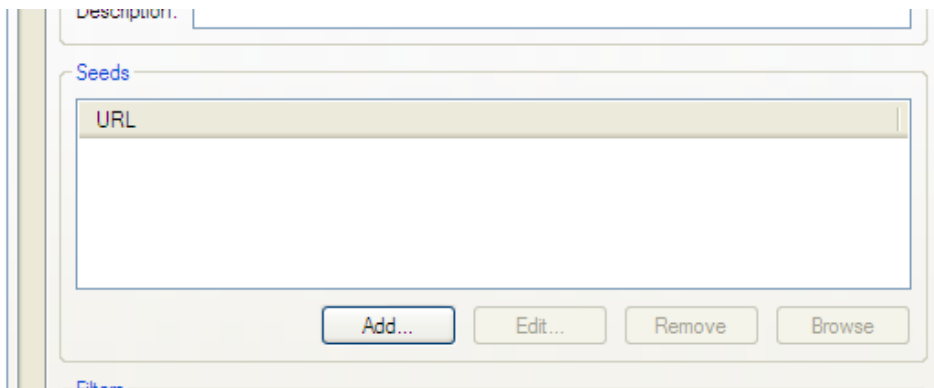


Assign a Name and a Description to the newly created Source and press the Apply button to confirm provided information.

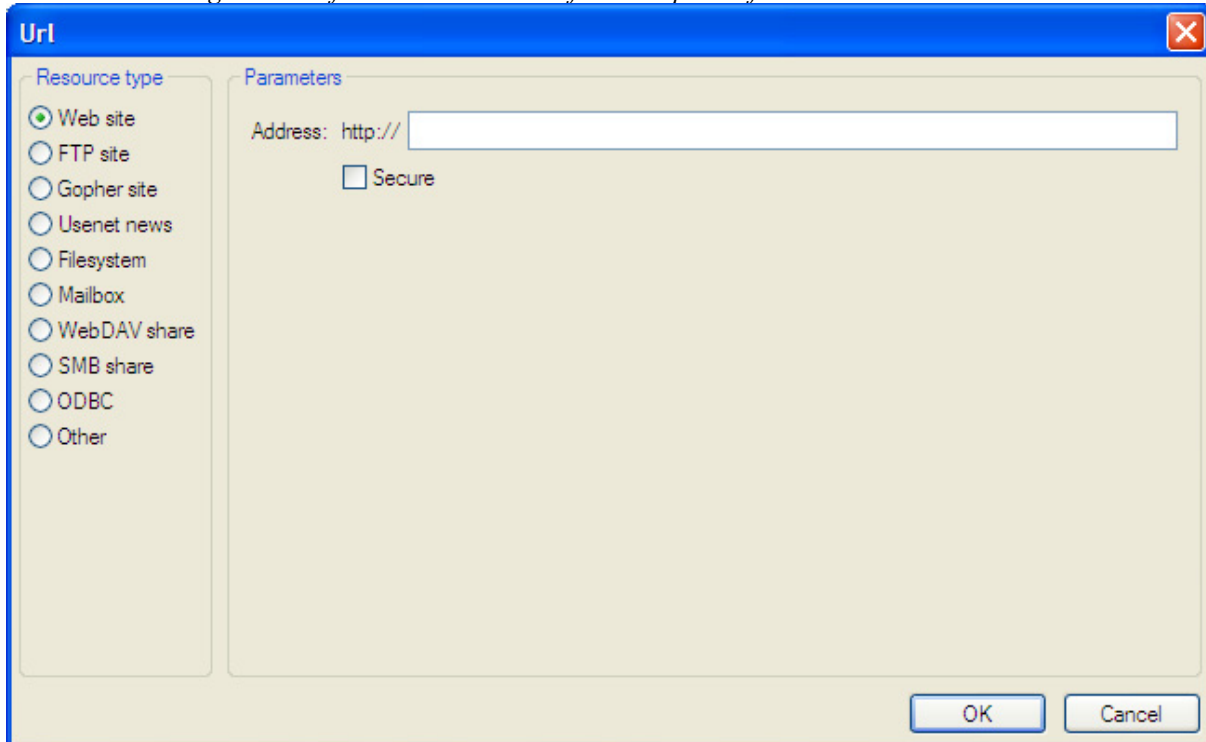


Now that the Source is created you need to add almost one Seed to it.

Just after the creation of a Source the list of Seeds it is composed of is empty.



Clicking on the Add... the following window is shown.

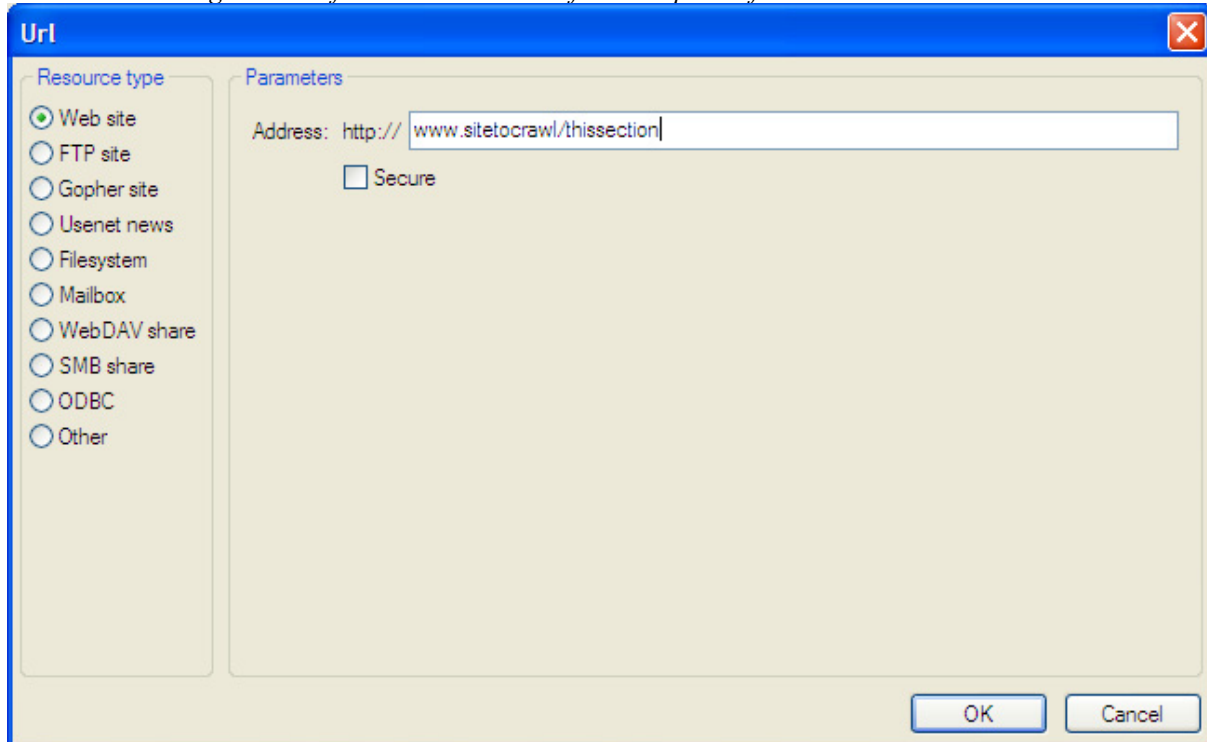


In the left side of the window the list of available types of Seed is possible to add to the current Source.

Here it is a detailed description of the configuration issues of any available type of Seed.

Web site

For a Seed related to a Web Site accessible through HTTP protocol the only information searchbox needs are the URL of the page where the gathering process must begin from and if the site needs a secure access.

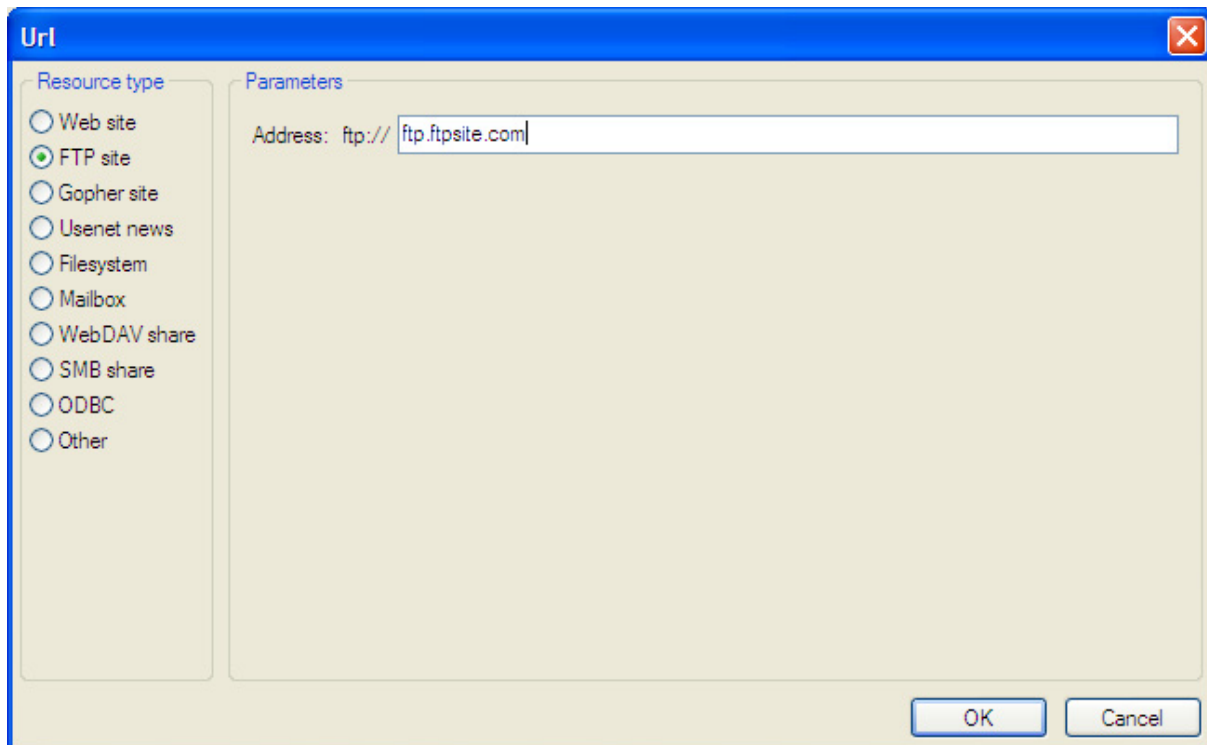


The 'Url' dialog box is shown with a blue title bar and a close button (X) in the top right corner. It is divided into two main sections: 'Resource type' on the left and 'Parameters' on the right. Under 'Resource type', there is a list of radio buttons: 'Web site' (selected), 'FTP site', 'Gopher site', 'Usenet news', 'Filesystem', 'Mailbox', 'WebDAV share', 'SMB share', 'ODBC', and 'Other'. Under 'Parameters', there is a text field labeled 'Address:' containing 'http://www.sitetocrawl/thissection' and a checkbox labeled 'Secure' which is currently unchecked. At the bottom right, there are 'OK' and 'Cancel' buttons.

When the Secure checkbox is checked, the URL prefix will change from http:// to https://

FTP site

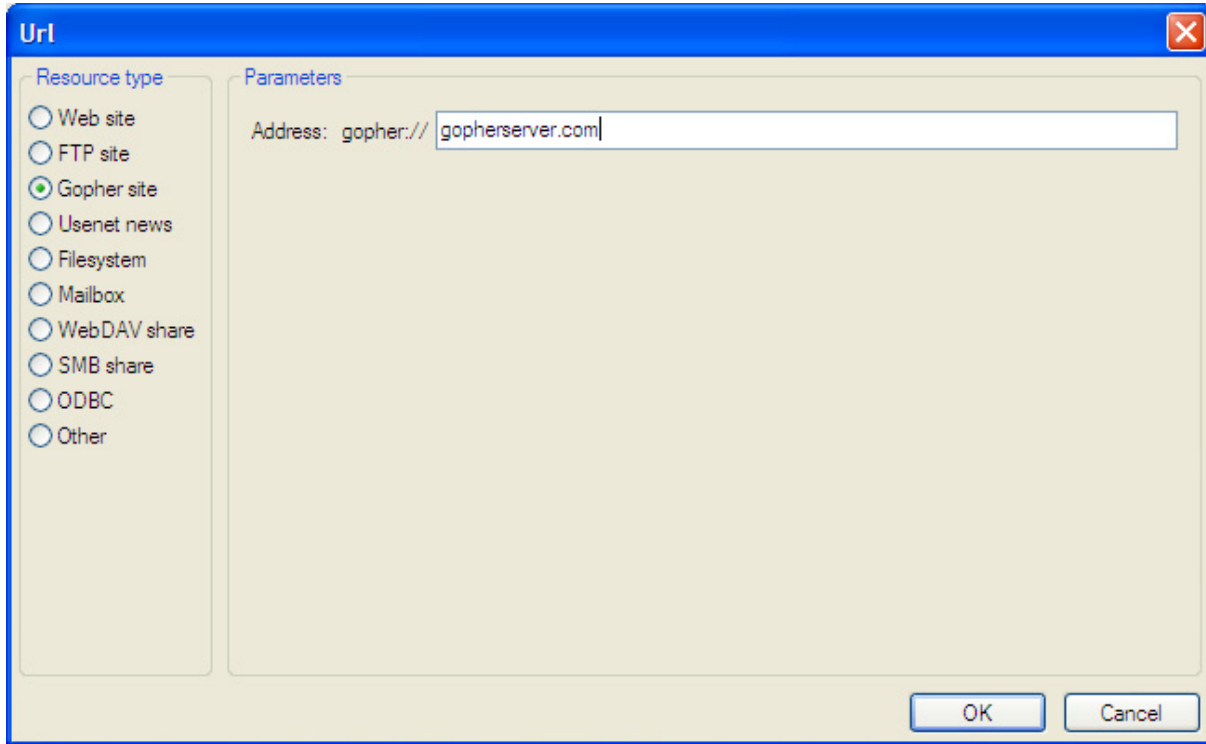
For FTP Seed the only needed parameter is its URL.



The 'Url' dialog box is shown with a blue title bar and a close button (X) in the top right corner. It is divided into two main sections: 'Resource type' on the left and 'Parameters' on the right. Under 'Resource type', there is a list of radio buttons: 'Web site', 'FTP site' (selected), 'Gopher site', 'Usenet news', 'Filesystem', 'Mailbox', 'WebDAV share', 'SMB share', 'ODBC', and 'Other'. Under 'Parameters', there is a text field labeled 'Address:' containing 'ftp://ftp.ftp site.com'. At the bottom right, there are 'OK' and 'Cancel' buttons.

Gopher site

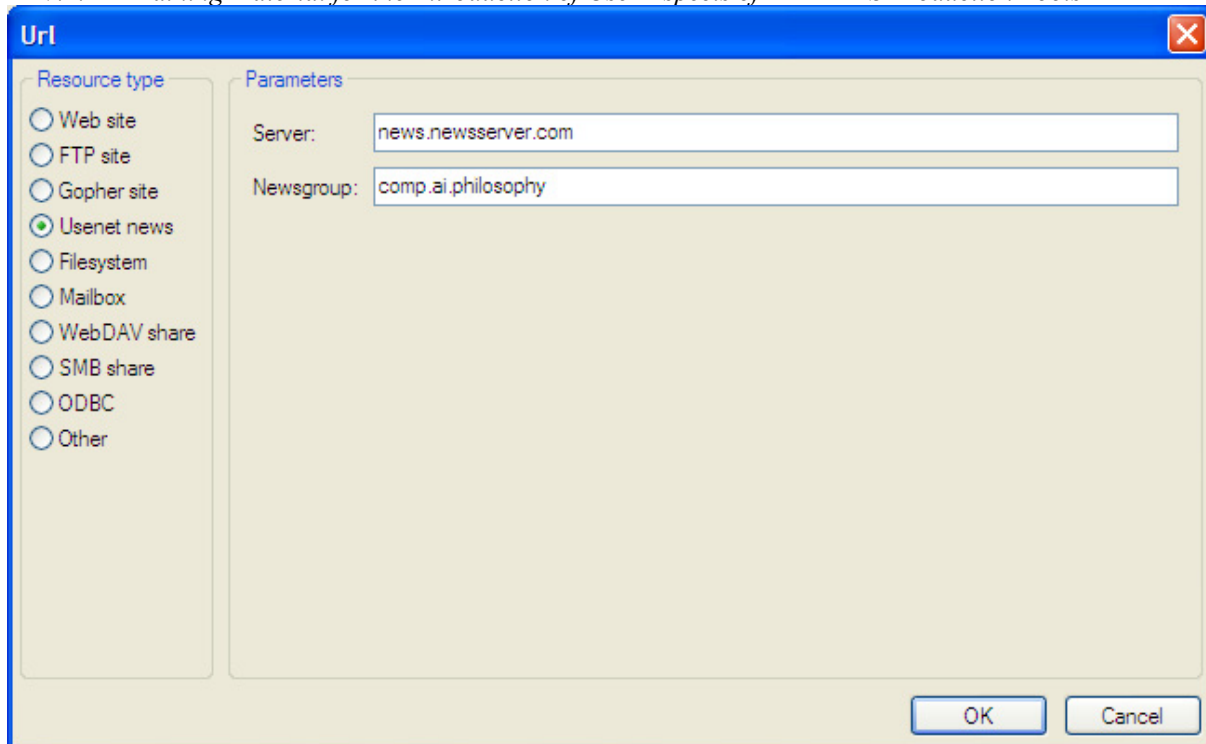
For Gopher sites just the URL is needed



For more information about the Gopher protocol you can see: http://en.wikipedia.org/wiki/Gopher_protocol

USENET site

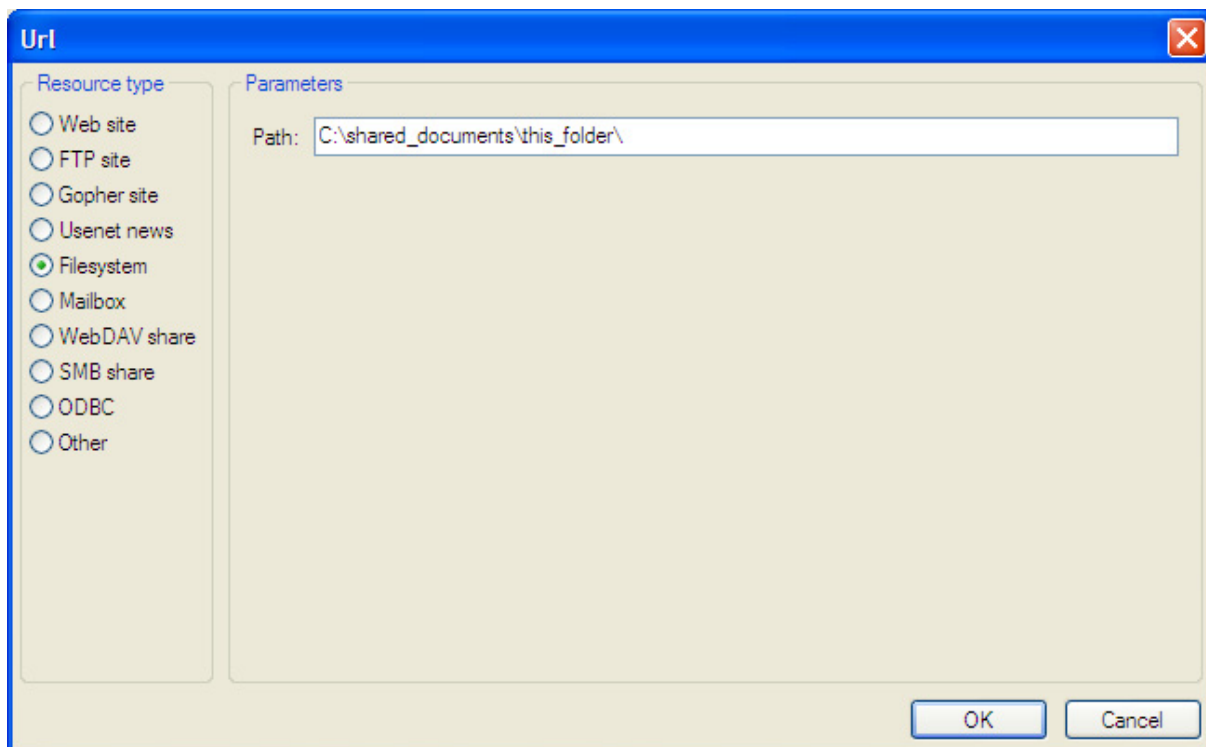
In this case both server name and the name of the newsgroup we want to gather from are needed.



The 'Url' dialog box is shown with a blue title bar and a close button. It contains two main sections: 'Resource type' and 'Parameters'. In the 'Resource type' section, there is a list of radio buttons: 'Web site', 'FTP site', 'Gopher site', 'Usenet news' (which is selected), 'Filesystem', 'Mailbox', 'WebDAV share', 'SMB share', 'ODBC', and 'Other'. In the 'Parameters' section, there are two text input fields: 'Server:' with the value 'news.newsserver.com' and 'Newsgroup:' with the value 'comp.ai.philosophy'. At the bottom right, there are 'OK' and 'Cancel' buttons.

Filesystem

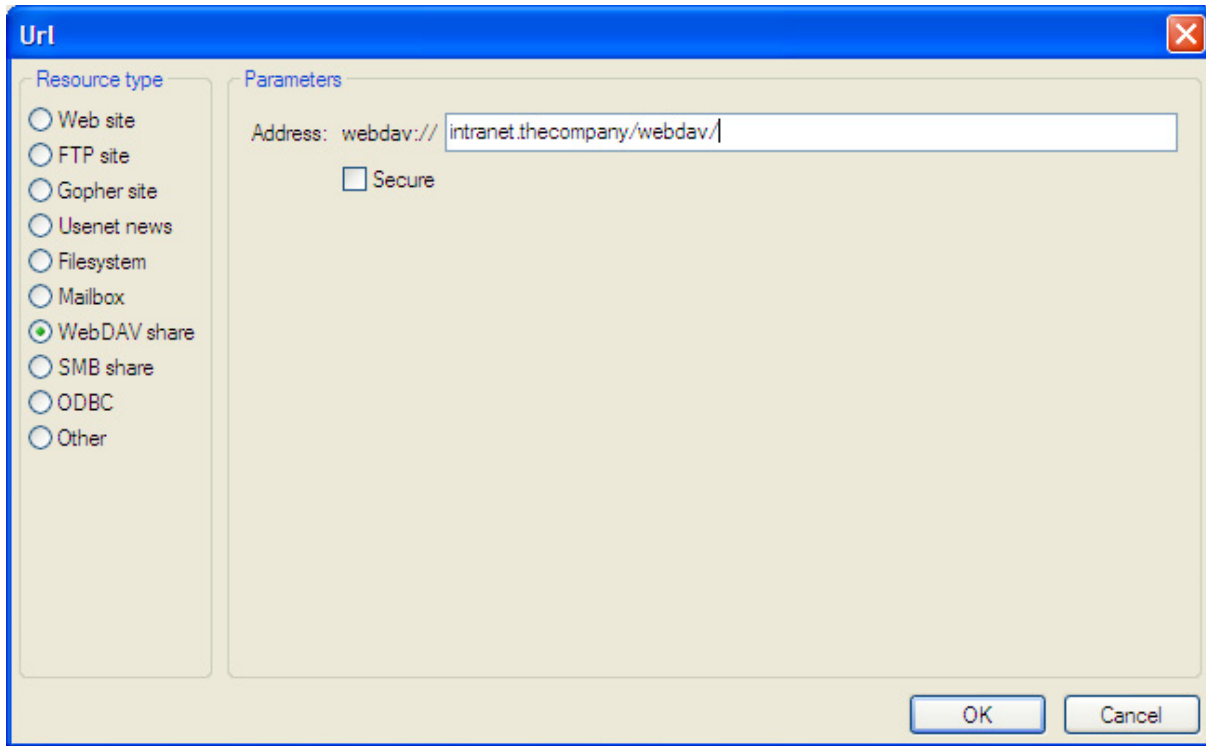
To gather information from a local filesystem. The absolute path of a folder is needed.



The 'Url' dialog box is shown with a blue title bar and a close button. It contains two main sections: 'Resource type' and 'Parameters'. In the 'Resource type' section, there is a list of radio buttons: 'Web site', 'FTP site', 'Gopher site', 'Usenet news', 'Filesystem' (which is selected), 'Mailbox', 'WebDAV share', 'SMB share', 'ODBC', and 'Other'. In the 'Parameters' section, there is a single text input field labeled 'Path:' with the value 'C:\shared_documents\this_folder\' (note the trailing backslash). At the bottom right, there are 'OK' and 'Cancel' buttons.

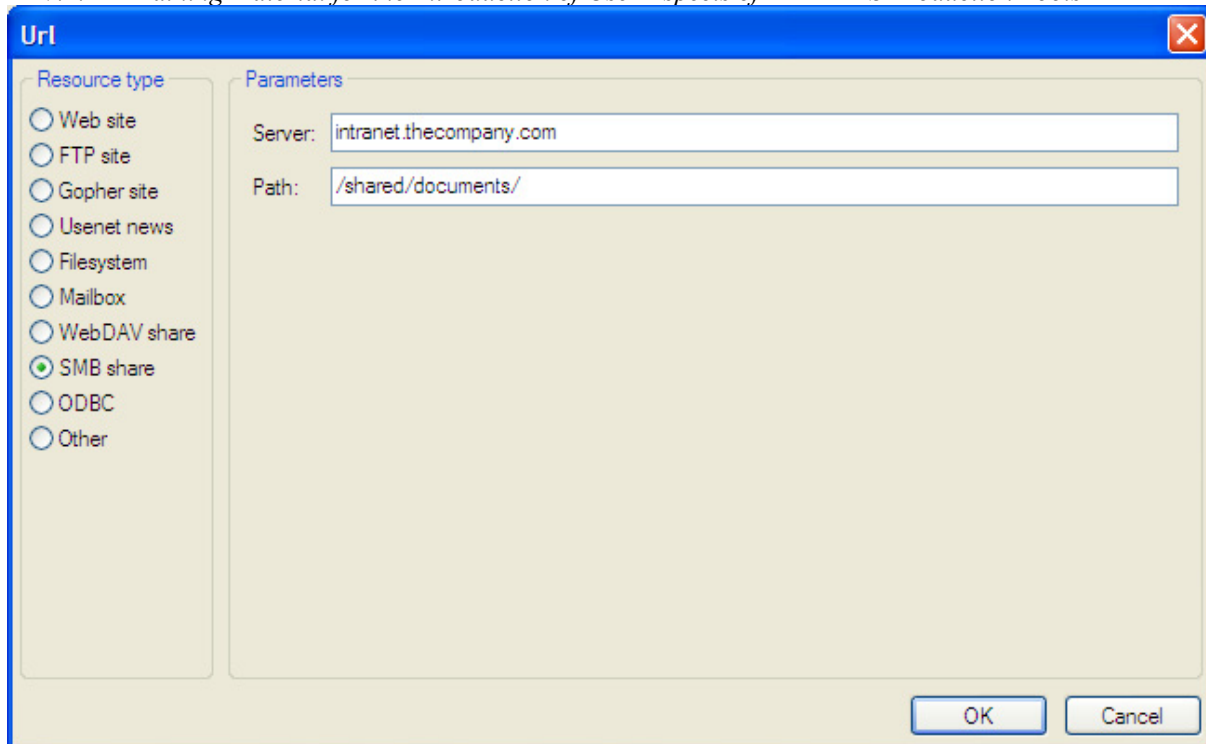
WebDav share

To gather information from a folder shared through WebDav protocol. The absolute URL of the WebDav folder is needed.



SMB share

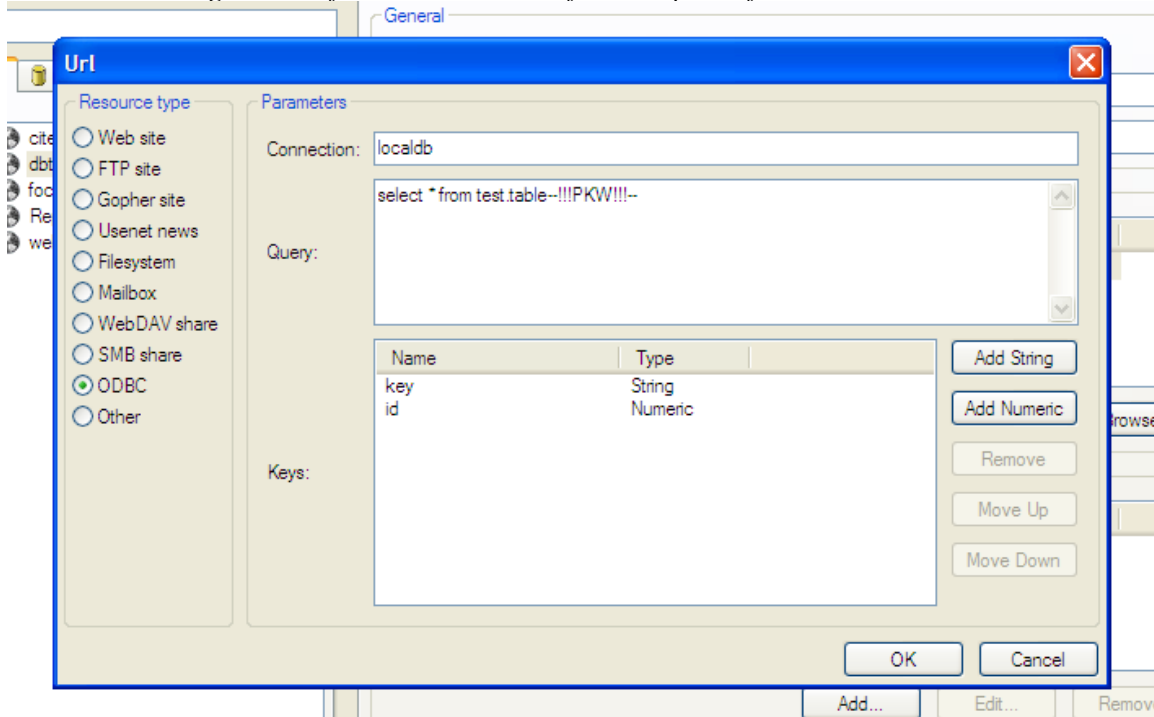
To gather information from a folder shared through SMB protocol. The absolute URL of the SMB folder is needed



ODBC

An ODBC seed is made of three parts:

- The *ODBC connection string*. It is usually a System DSN^[3]. Note that your database might impose access restrictions that will stop searchbox to access it even if you can access the database itself from your desktop. E.g. the database might be configured to refuse queries coming from the searchbox server computer.
- The *query*. This is basically any sql query with the additional requirement that exactly one of the special strings `--!!!PKW!!!--` or `--!!!PKA!!!--` must be present. These special strings are described below.
- The *keys*. This is a subset of the names of the columns of the result set returned by *query* whose value uniquely identifies a row in the result set itself. For example if *query* involves a single table you can use the table primary key. You must specify the order of the fields^[4] and whether each field contains a string or a numeric value.



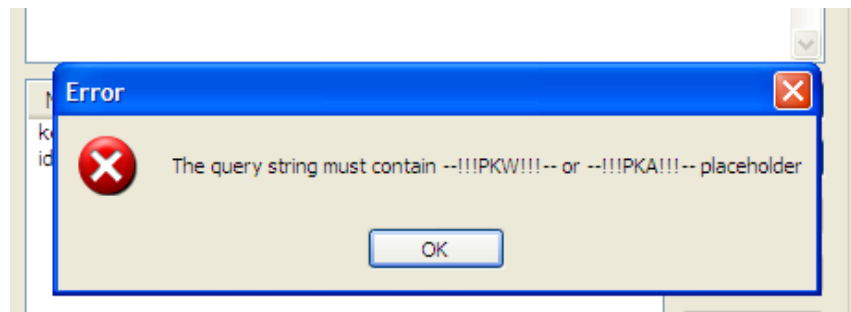
As stated above *query* must contain exactly one of `--!!!PKW!!!--` or `--!!!PKA!!!--`. searchbox will expand these with value taken from the *keys* fields to access specific rows in the database. The two strings are expanded nearly in the same way but `--!!!PKW!!!--` begins with an sql where keyword, while `--!!!PKA!!!--` begins with and. Thus you must use the latter when you have placed a where clause in *query* (typically you append `--!!!PKA!!!--` after your clause) while the former is useful when you don't have any explicit where clause in your query. For example:

select * from tableA --!!!PKW!!!--

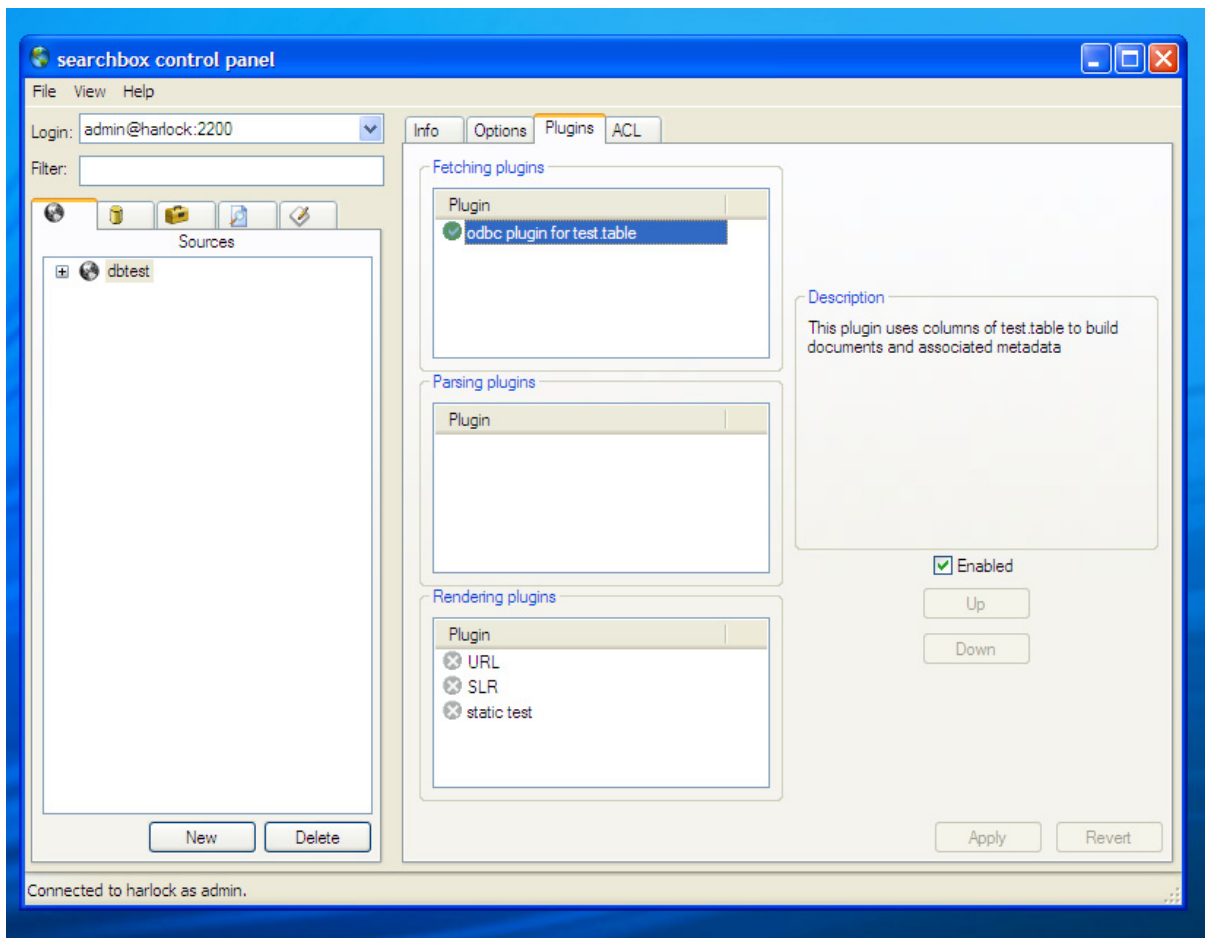
but

**select * from tableB, tableC
where tableB.id = tableC.id
--!!!PKA!!!--**

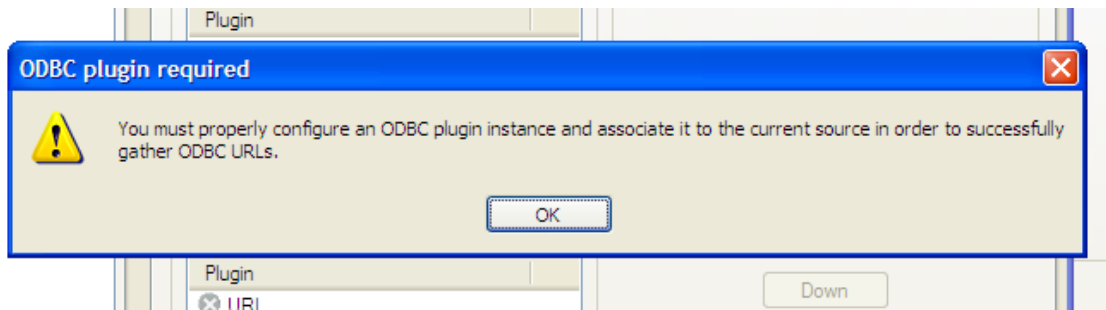
If you don't specify the `--!!!PKW!!!--` nor the `--!!!PKA!!!--` placeholder then the controlpanel will complain.



In order to successfully crawl an odbc source adding a seed is not enough: you also have to specify a set of rules to let searchbox turn the odbc result set into documents. These rules are specified in an **odbc plugin configuration**. When you add an odbc seed to a source you must then enable the suitable plugin configuration on the source.



If you don't, searchbox complains.



6.1.5 AXMEDIS Database (EXITECH)

AXMEDIS Database and related areas such as the Query Support Web service are typical back-end activities that can be difficultly demonstrated, since they:

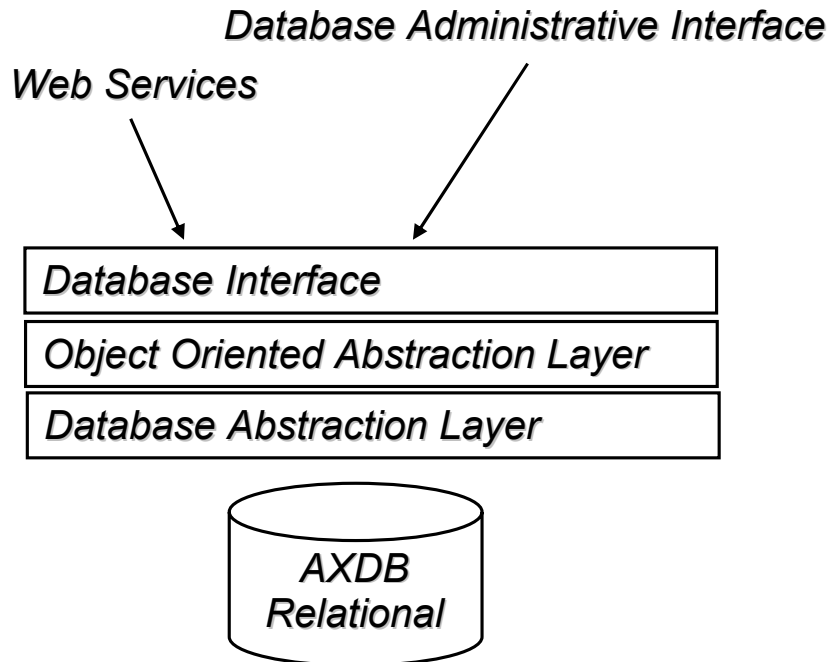
- Have no user interface
- Have complex XML file as input and output

To this end, the demonstration and training must be focused mainly on the architecture of the system showing how the database and the related tools and web services can support the user and his business.

The training is organized in two parts, the first presenting the AXDB and its architecture, while the second presents the Query Support, its architecture, the query format and a small demonstration of the web service that distributes and receives results.

6.1.5.1 Contextualisation of AXMEDIS Database

AXDB is a relational database that is accessible via an object oriented layer as depicted in the next picture:



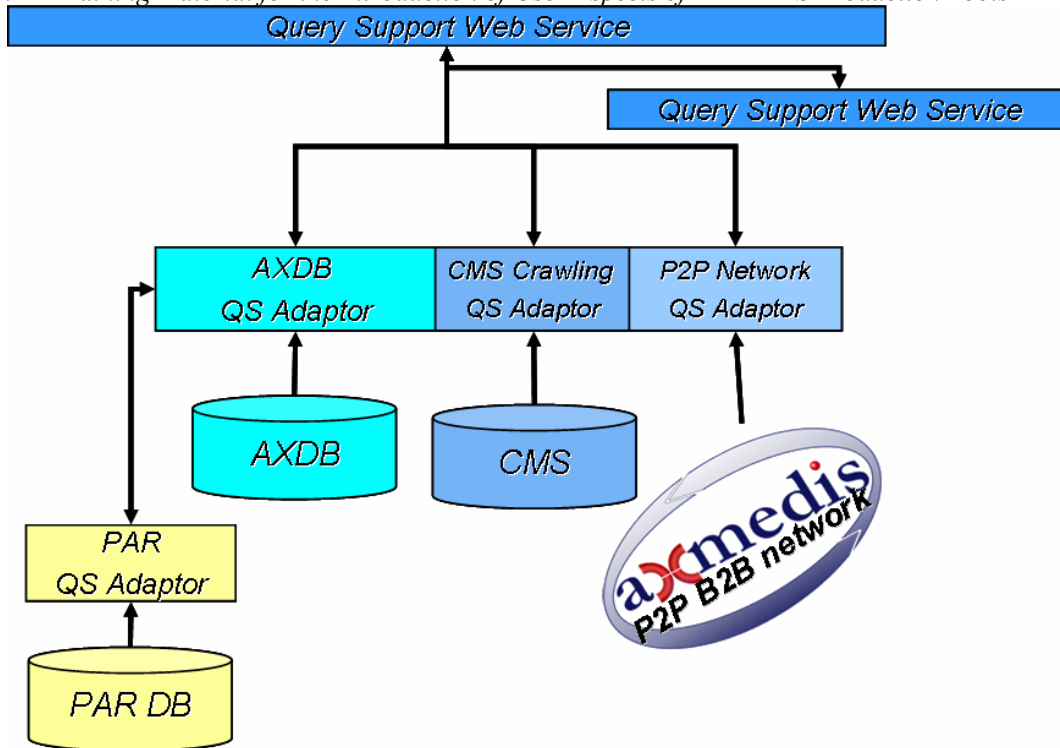
The whole AXMEDIS framework can access directly the API of the Database Interface or can access to a subset of such API that are accessible via Web Service.

The main functionalities that the database area has are:

- Loading an AXMEDIS object from the database, that is obtaining a pointer to a URI where a specified version of a certain object can be retrieved.
- Saving an AXMEDIS object in the database, that is the storing of the object in the database with the indexing of all the metadata that are relevant for the company business
- Factory User and Group Management, that is the creation/elimination/modification of user and groups with the management of the related rights
- Management of Object Versioning
- Management of metadata and data that are relevant for other tools in the AXFW.

The architecture of the Query Support Web service is reported below, where we have three main layers that are:

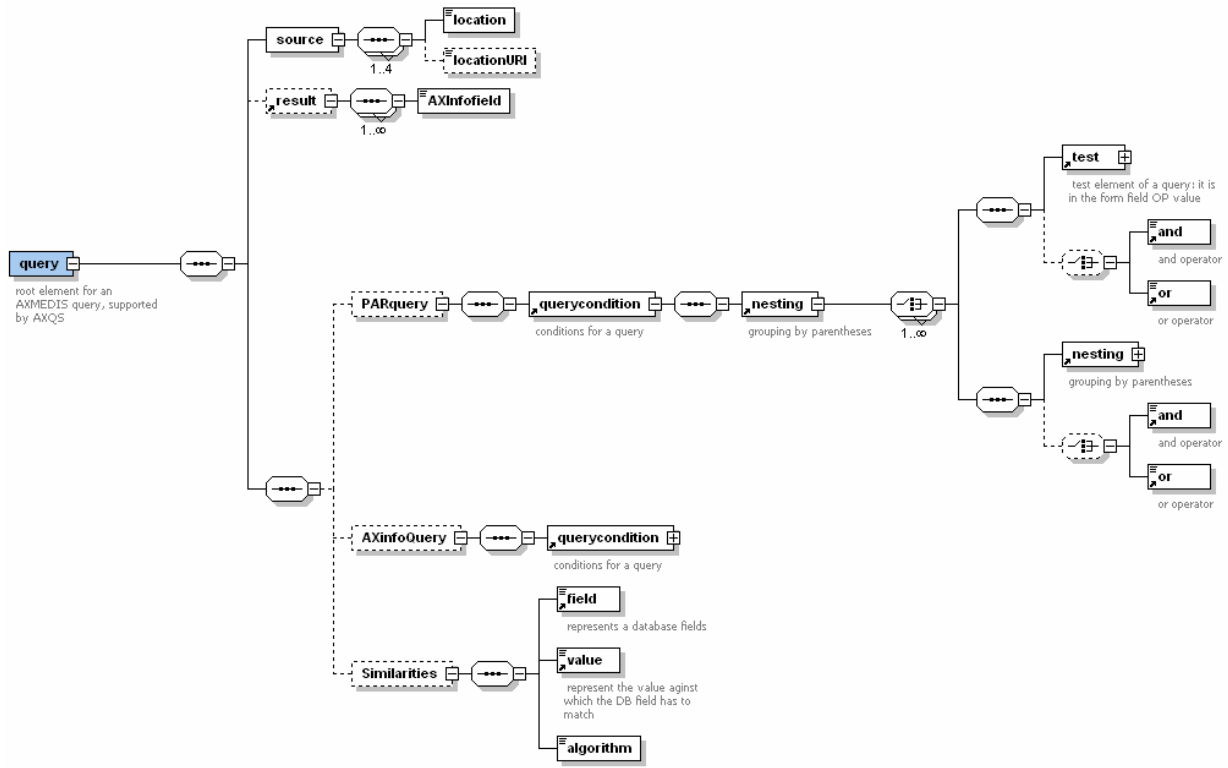
- Data Source Layer (with AXDB, CMS and P2P network)
- Query Support Adaptor layer, that are web services with the same interface in terms of WSDL with respect to the Query Support Web Service
- Query Support Web Service, that is the main Web Service available for public access



The mechanism used for distributing a query and collecting results is quite complex, but can be summarized as follows:

1. Query Support Web Service receives the query from an external process/user
2. Query Support Web Service distributes the query among the different Adaptors and to another query support
3. Each adaptor returns synchronously or a synchronously the results to the QS web service
 - a. CMS crawls CMS
 - b. P2P interrogates the network
 - c. AXDB Adaptor work:
 - i. Resolve the query on the AXDB
 - ii. Forward the query to the PAR Web Service
 - iii. Returns to Query Support Web service the AND between PAR DB and AXDB
 - d. The external query support return the results
4. Query Support collects the results:
 - a. In the case of synchronous query the results are collected in AND and posted to the client
 - b. In the case of a synchronous queries the results are returned to the client as they are received and the client is notified when no more results are present

It is important to define the format for the query and for the results. The query is an XML file that matches the following structure:



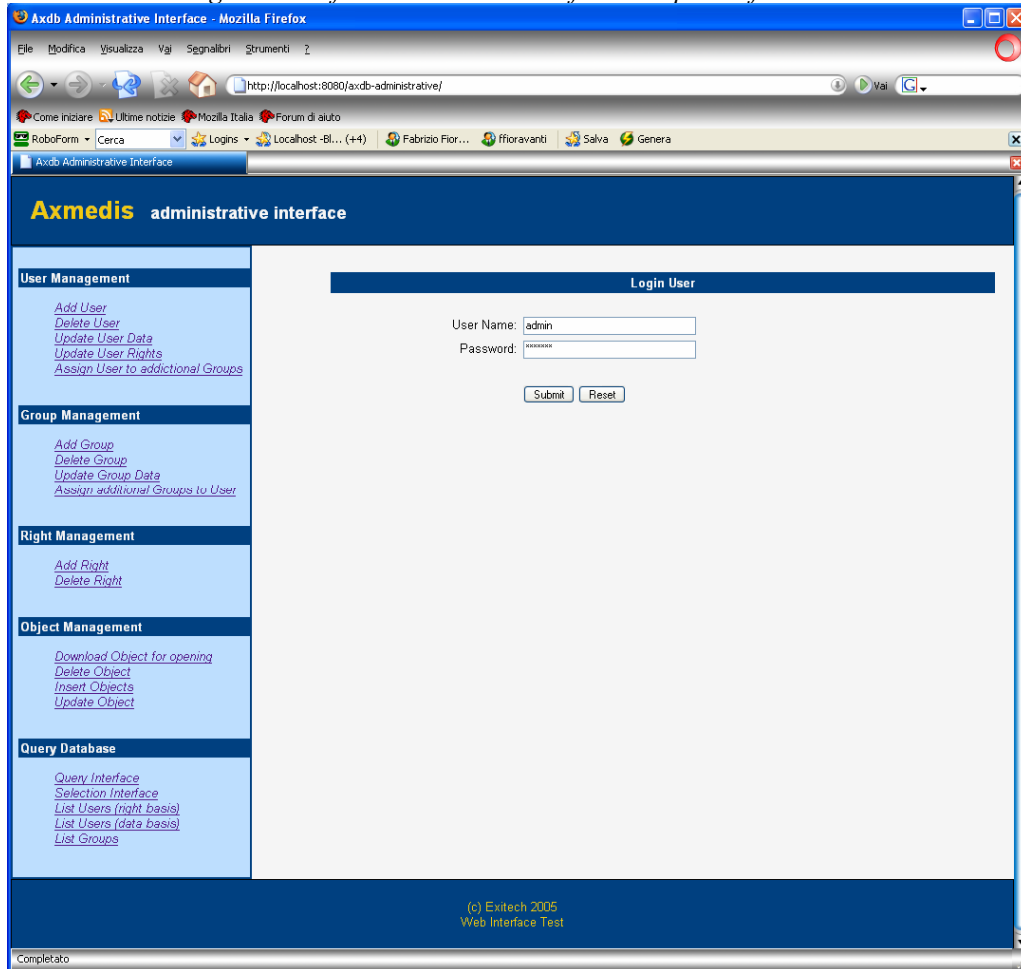
A query result can contain optional errors that if present are reported in the errorfiles array and array of AXObject that fulfill the query criteria. For each AXObject is present the AXOID and information for retrieval in the case of a peer of the P2P network; in the extrainfo are contained all the metadata that have been requested in the AXinfoField part of the query; the rootobjects section contains for each object that fulfils the query, the objects that can be downloaded and that contains the AXOID identified in the result.

6.1.5.2 Demonstration of AXMEDIS Database Tools

As stated in the first part of section **Errore. L'origine riferimento non è stata trovata.** it is very difficult to demonstrate back office activities, and therefore it is only possible to show how the administrative interface of AXDB works, that is the only visible part of the AXDB system and tools set.

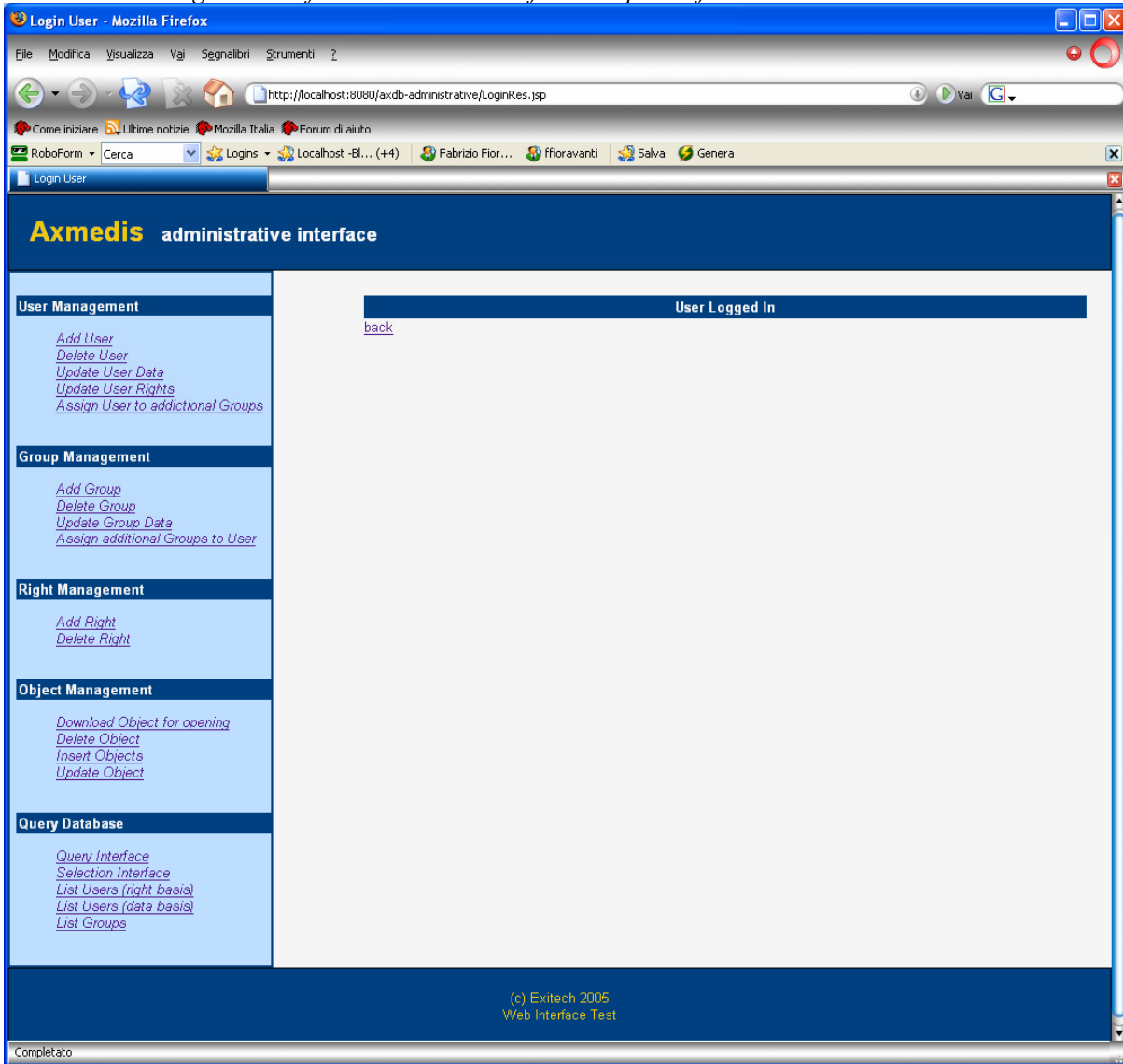
Login

Login interface will allow login for each user registered in the DB. Instead of the user text input it will appear the list of user and the menu on the right will also change of the basis of the user permission.



The web interface will be simple and effective having on the left side a menu bar for choosing operations and on the centre of the page the parameters for accessing to the requested functionality.

In order to increase accessibility no frame will be used.



Each functionality will have a different mask in the centre of the page that allow to operate on the functionality.

In the following the functionality to be implemented will be listed together with some sample snapshots.

User Management

Add user

Axmedis administrative interface

User Management

- [Add User](#)
- [Delete User](#)
- [Update User Data](#)
- [Update User Rights](#)
- [Assign User to additional Groups](#)

Group Management

- [Add Group](#)
- [Delete Group](#)
- [Update Group Data](#)
- [Assign additional Groups to User](#)

Right Management

- [Add Right](#)
- [Delete Right](#)

Object Management

- [Download Object for opening](#)
- [Delete Object](#)
- [Insert Objects](#)
- [Update Object](#)

Query Database

- [Query Interface](#)
- [Selection Interface](#)
- [List Users \(right basis\)](#)
- [List Users \(data basis\)](#)
- [List Groups](#)

Add User Form

User Information:

Name:

Password:

Email address:

Note:

Main Group:

User Rights:

load object ☒

save object ☒

admin db ☐

All enabled ☐

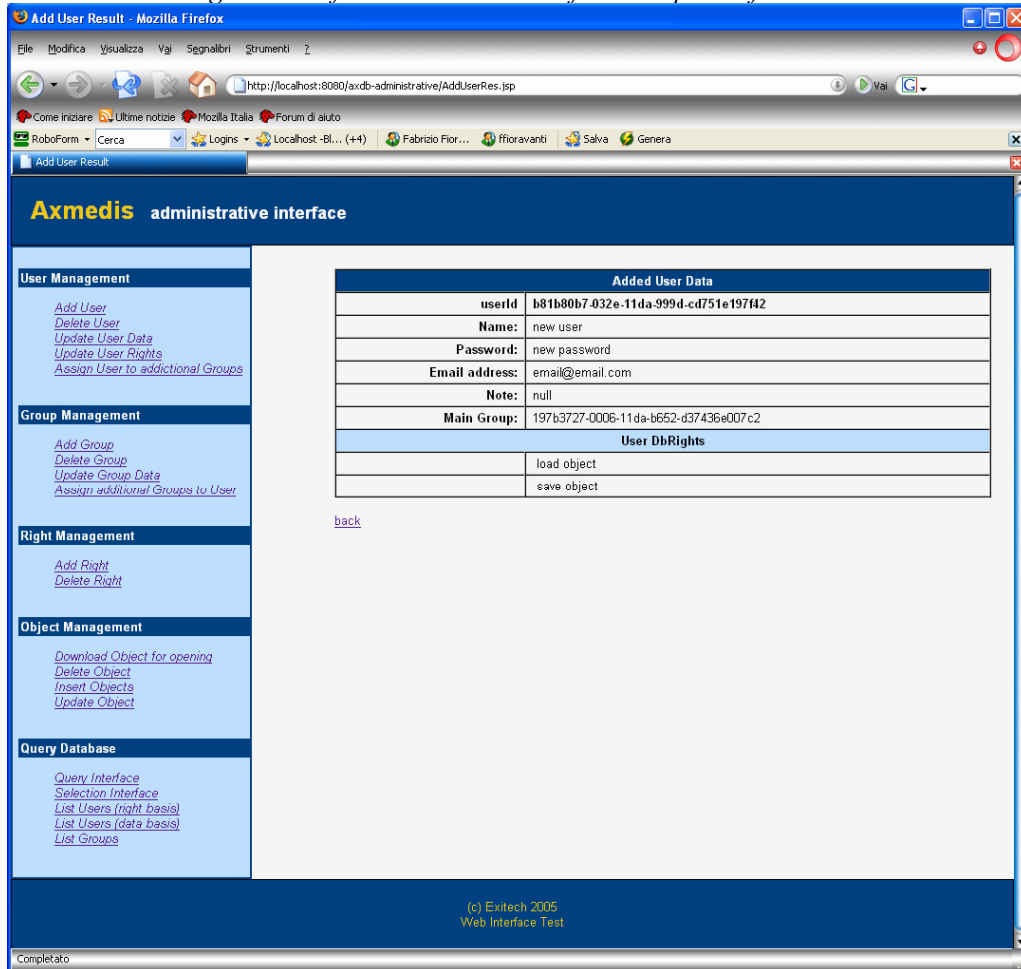
QS enabled ☐

delete object ☐

(c) Exitech 2005
Web Interface Test

Completo

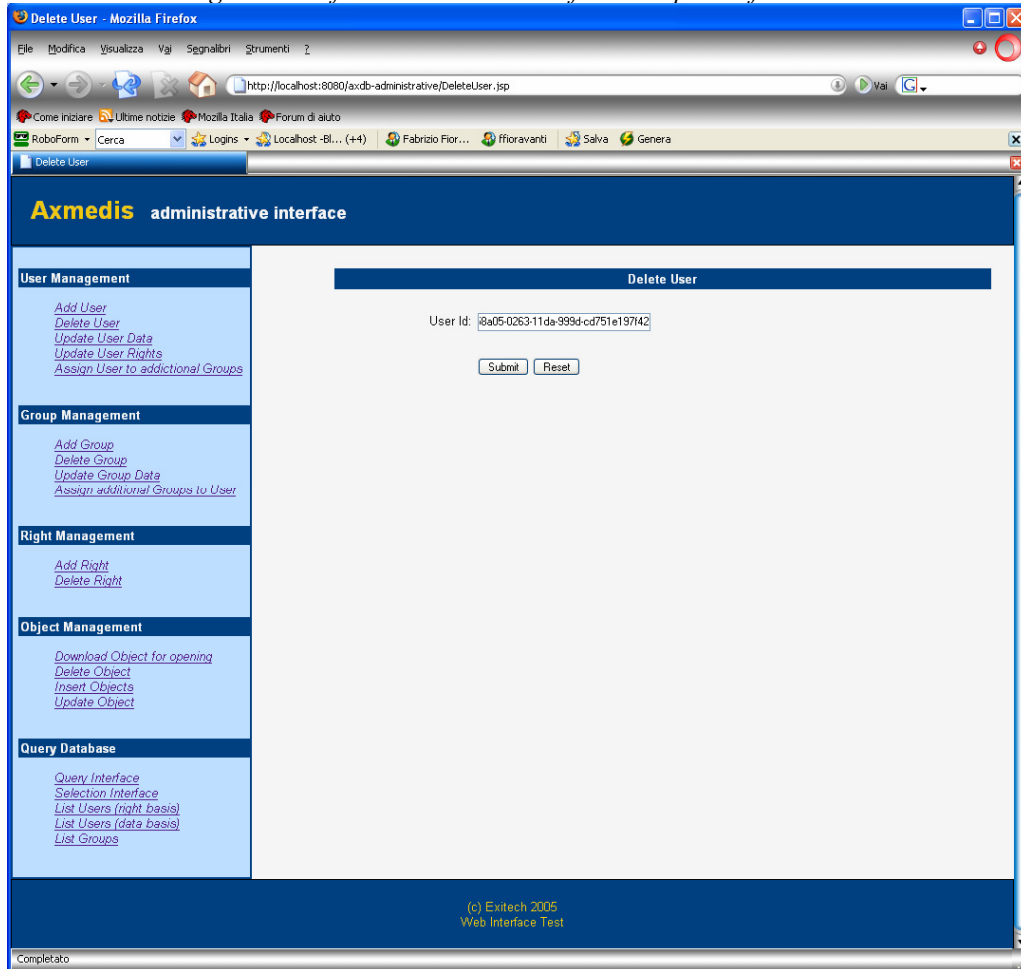
The submission allows to show the inserted parameters and the generated User ID.



Final version will have the ID prefixed by FUS- (that states for Factory User)

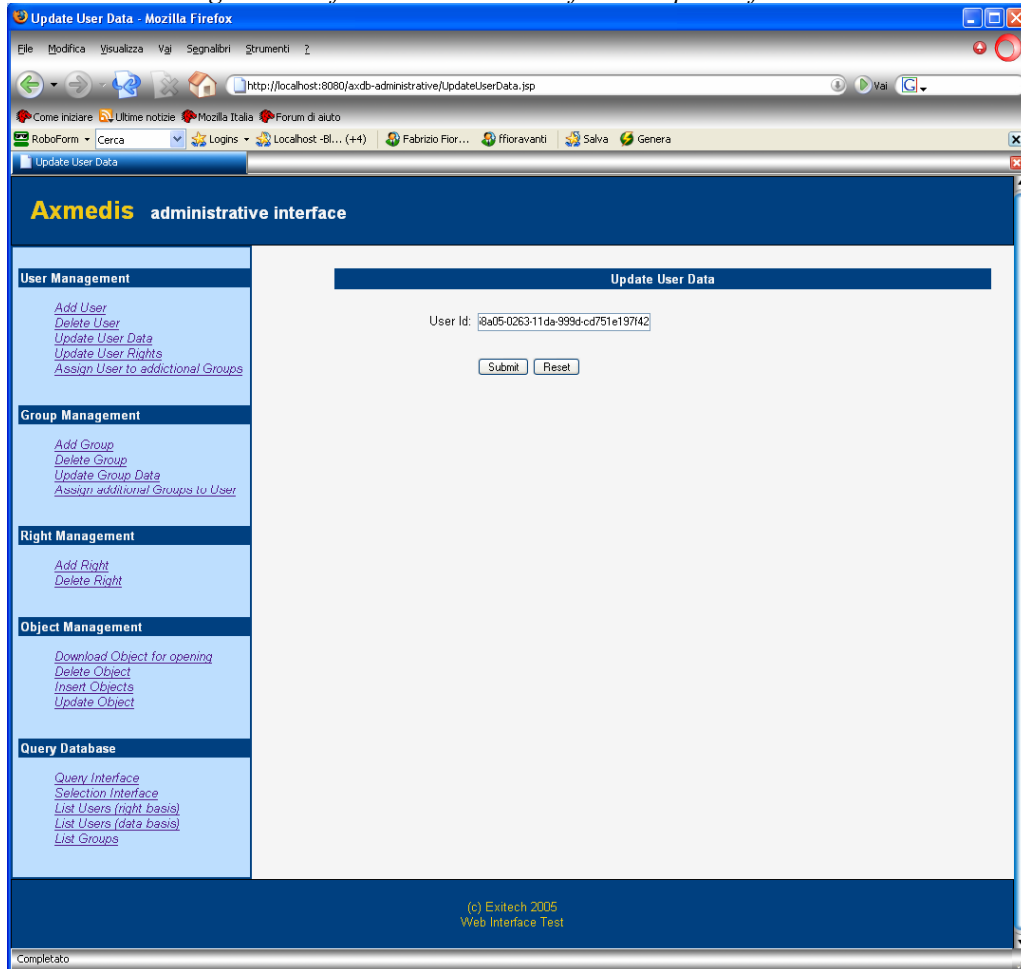
Delete user

This page allows the operator to remove a registered user. In the next version, a list of user will appear instead of the text input to insert the user ID. It is not possible to delete an admin user.

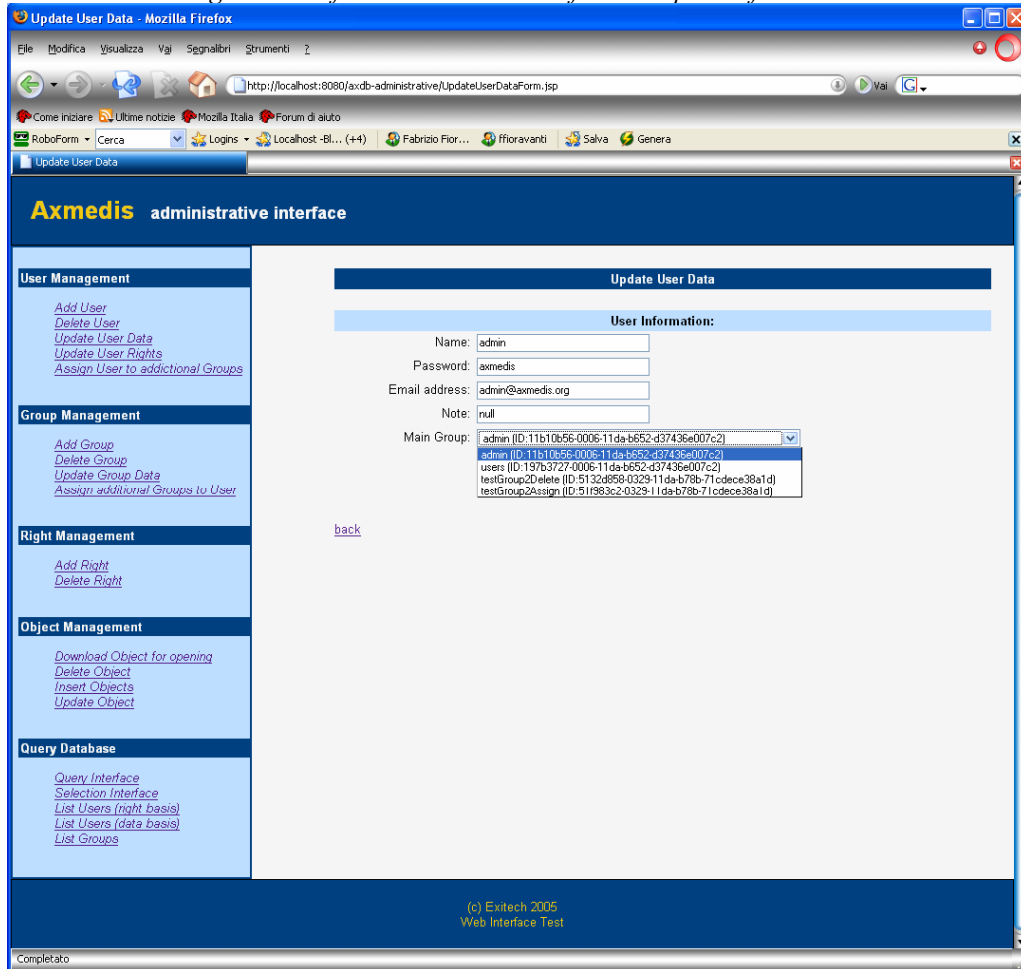


Update user data

This page allows to update the user data. In the next version a list of user will appear instead of the text input to insert the user ID.

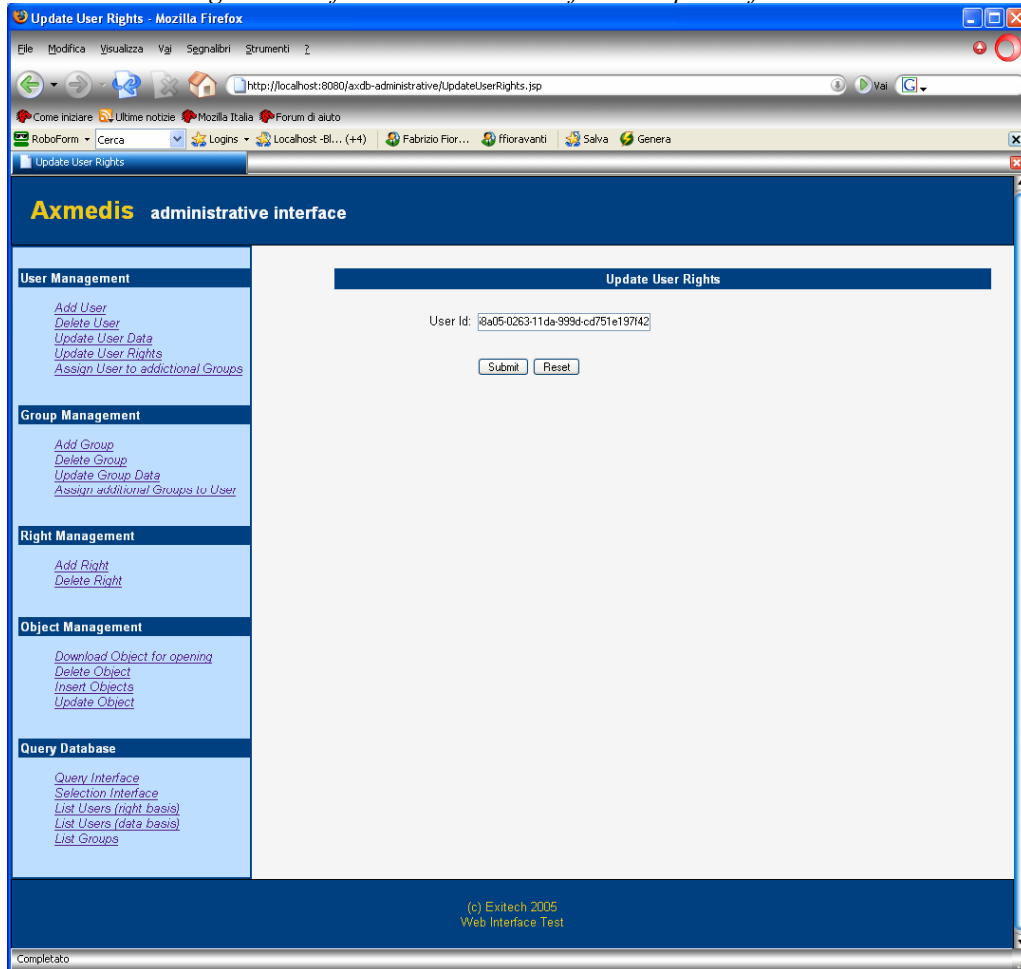


Once the user has been identified, it will be possible to change its name, password, email, note and main group.

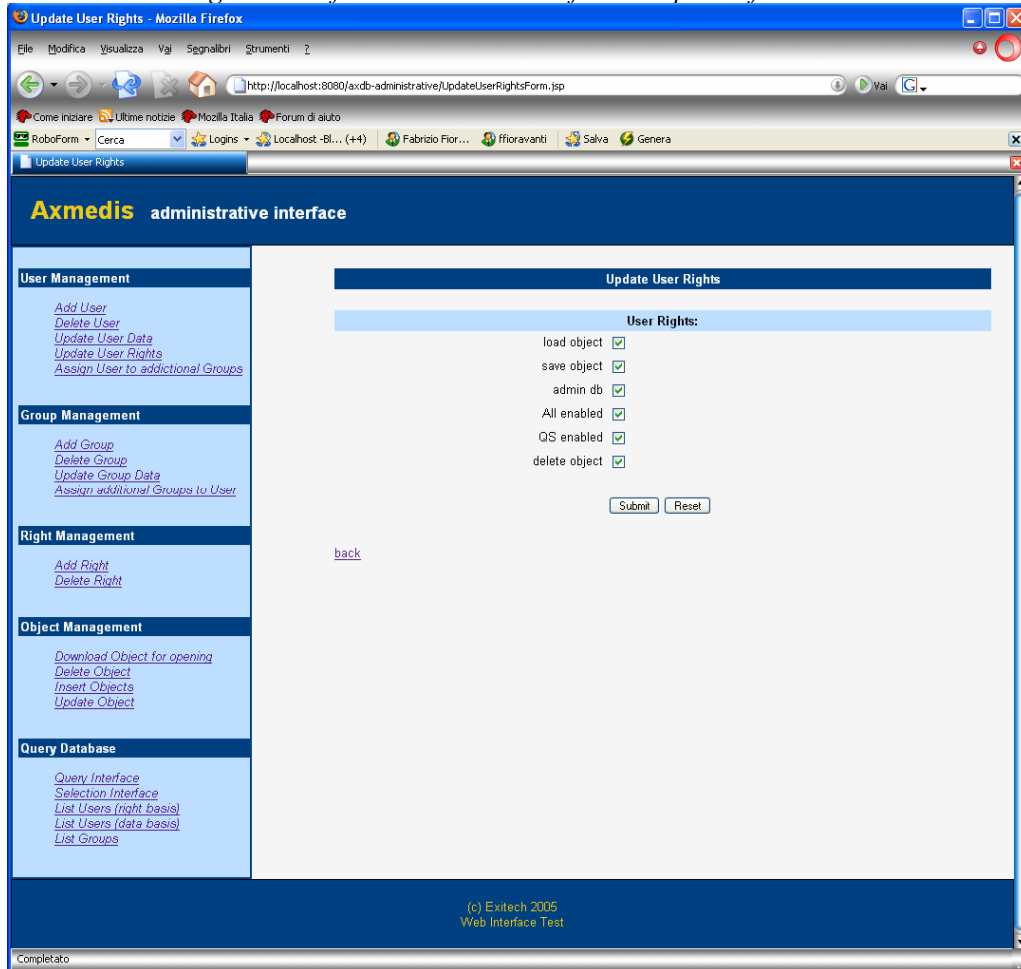


Update user rights

This page allows to update the user rights. In the next version a list of user will appear instead of the text input to insert the user ID.

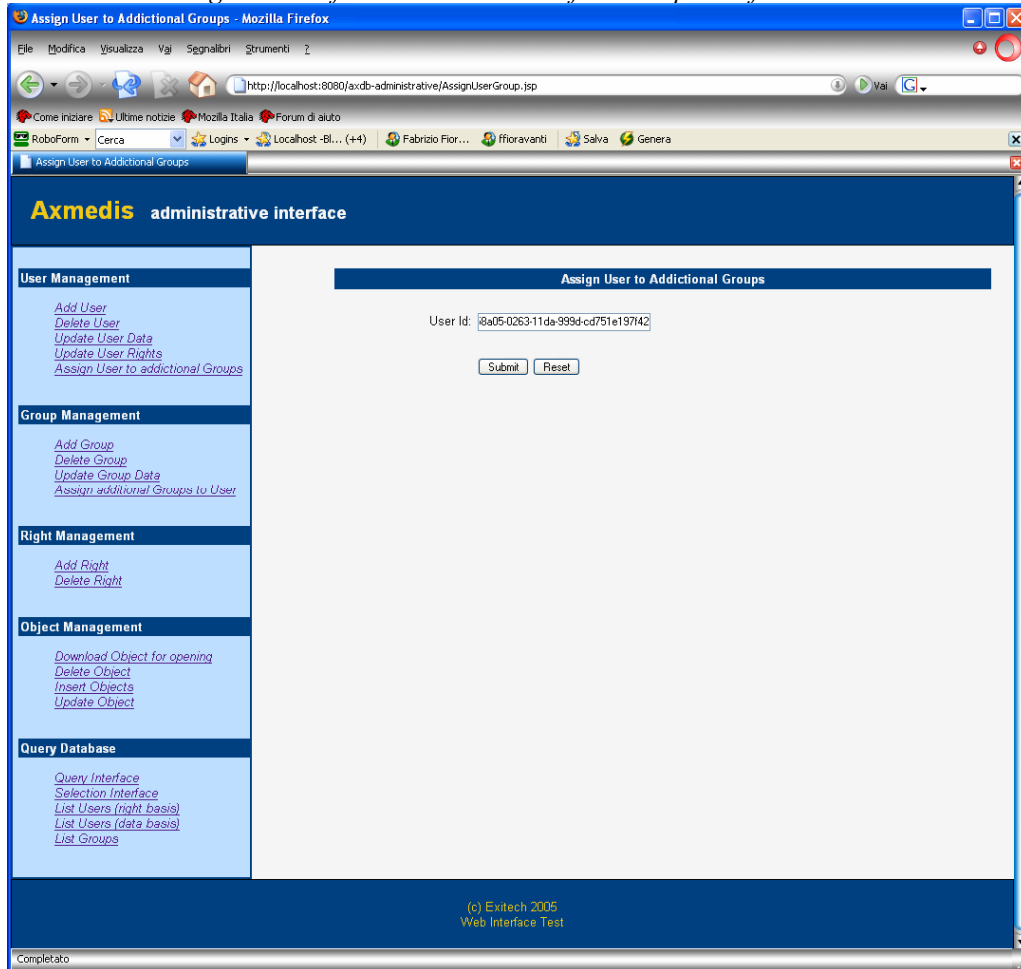


Once the user has been identified, it will be possible to change the rights that the user have. Current rights are checked, while available rights not currently active are not checked.

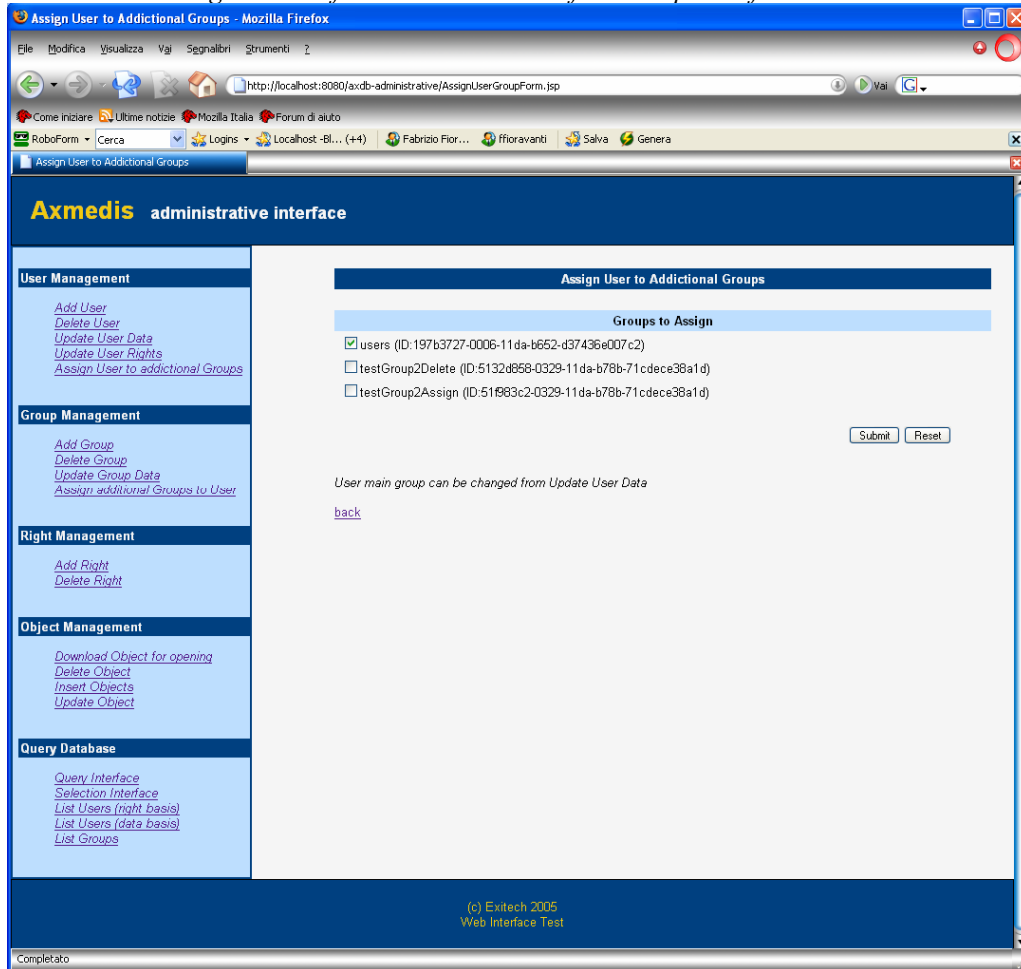


Assign user to additional groups

This page allows a single user to be assigned to different groups. In the next version, a list of user will be shown, instead of the text input to insert the user ID.



Once the user is selected it is possible to add groups to it.



Group Management

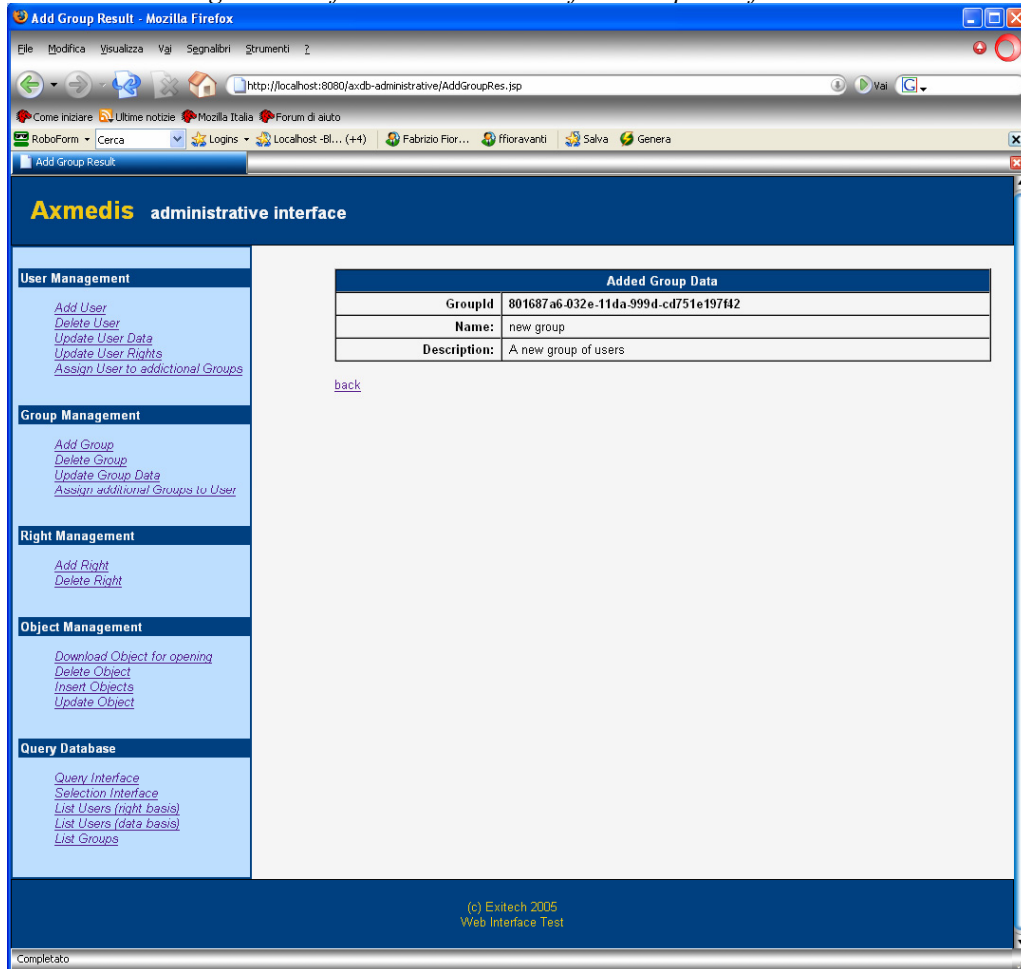
Add Group

This page allows the operator to add a group.

The screenshot shows a web browser window titled "Add Group - Mozilla Firefox" displaying the "Axmedis administrative interface". The browser's address bar shows the URL "http://localhost:8080/axdb-administrative/AddGroup.jsp". The interface has a blue header and a sidebar on the left containing several management categories: "User Management", "Group Management", "Right Management", "Object Management", and "Query Database". Each category lists specific actions like "Add User", "Delete User", "Add Group", etc. The main content area is titled "Add Group Form" and contains a "Group Information:" section with input fields for "Name" (containing "new group") and "Description" (containing "A new group of users"). Below these fields are "Submit" and "Reset" buttons. At the bottom of the page, there is a copyright notice "(c) Exitech 2005 Web Interface Test" and a status bar at the very bottom showing "Completato".

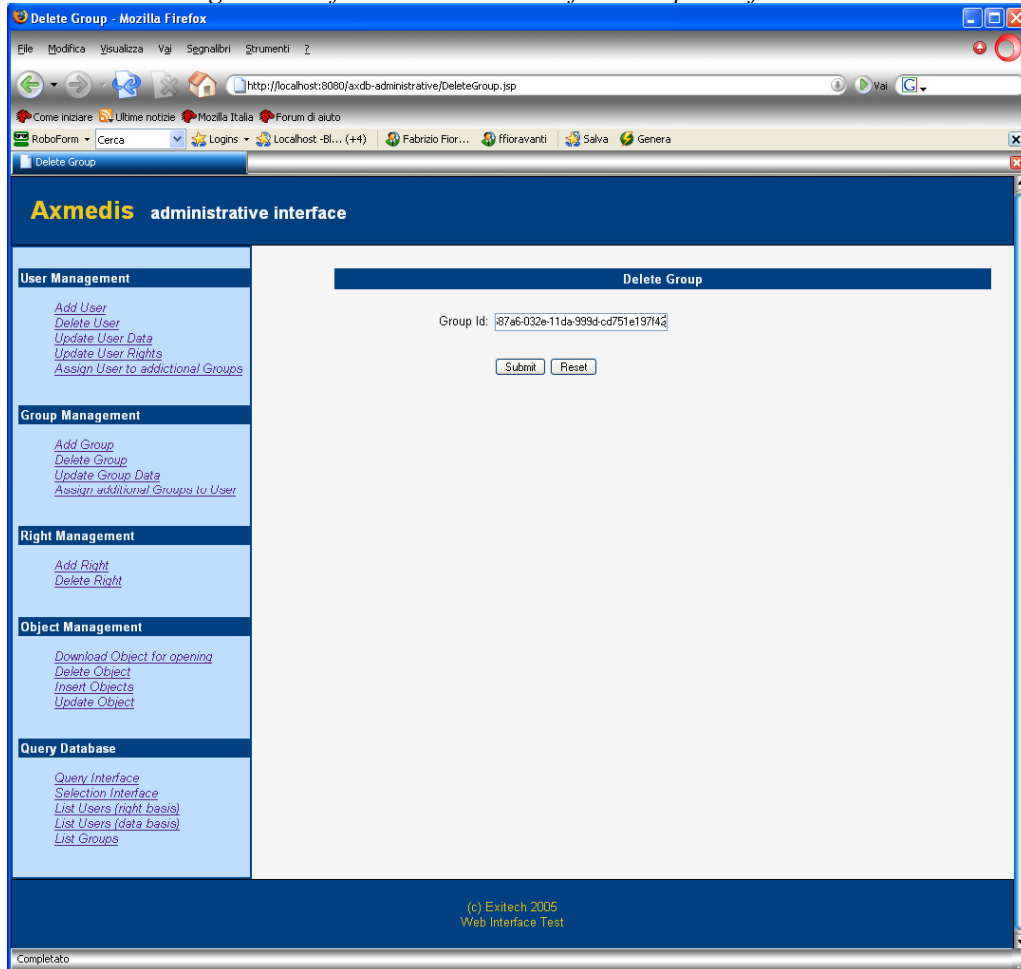
After submit a confirmation page showing the automatically generated group ID will be shown. Final version will have GroupID be prefixed by FGR- (that states for Factory Group). In the final version the groups will be associated also to a set of rights that will be inherited by all the user belonging to a group.

These additional rights will not be modified by the user management but only by the group management.



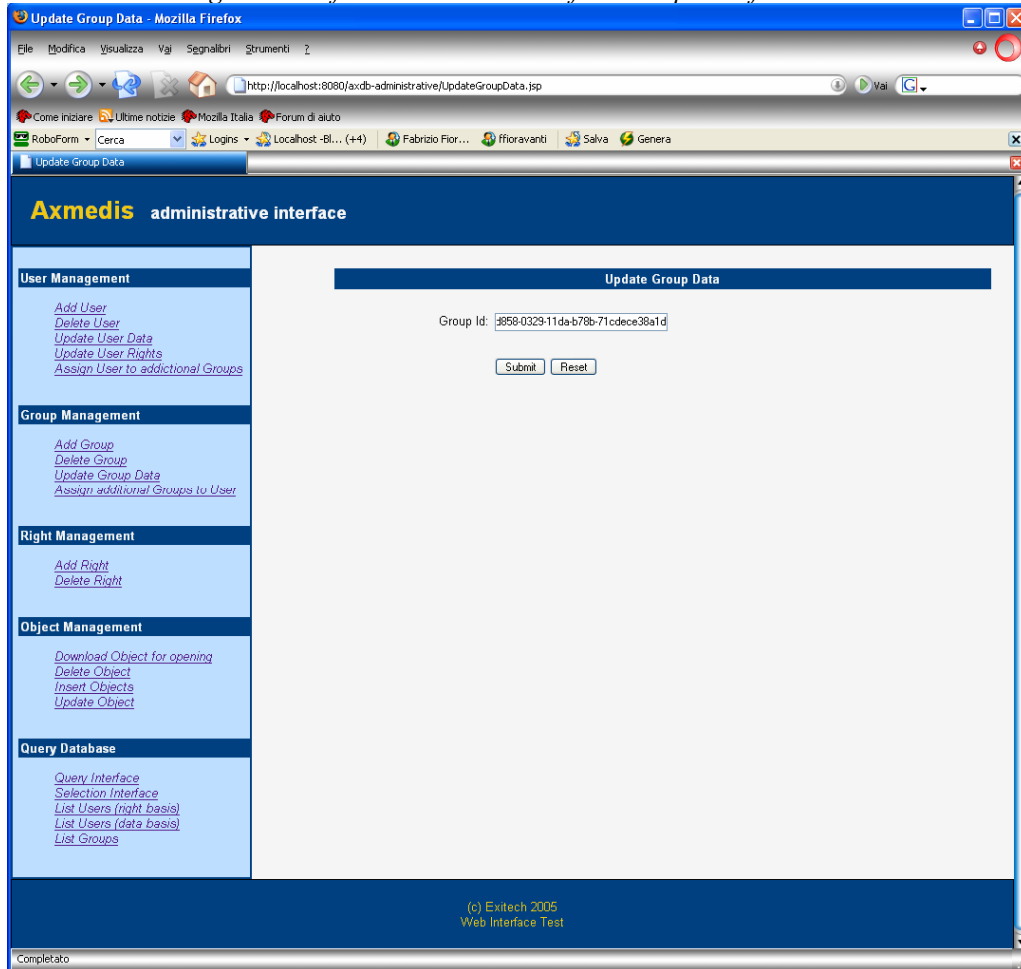
Delete group

This page allows to delete a group. In the next version a list of groups will appear instead of the text input to insert the group ID.

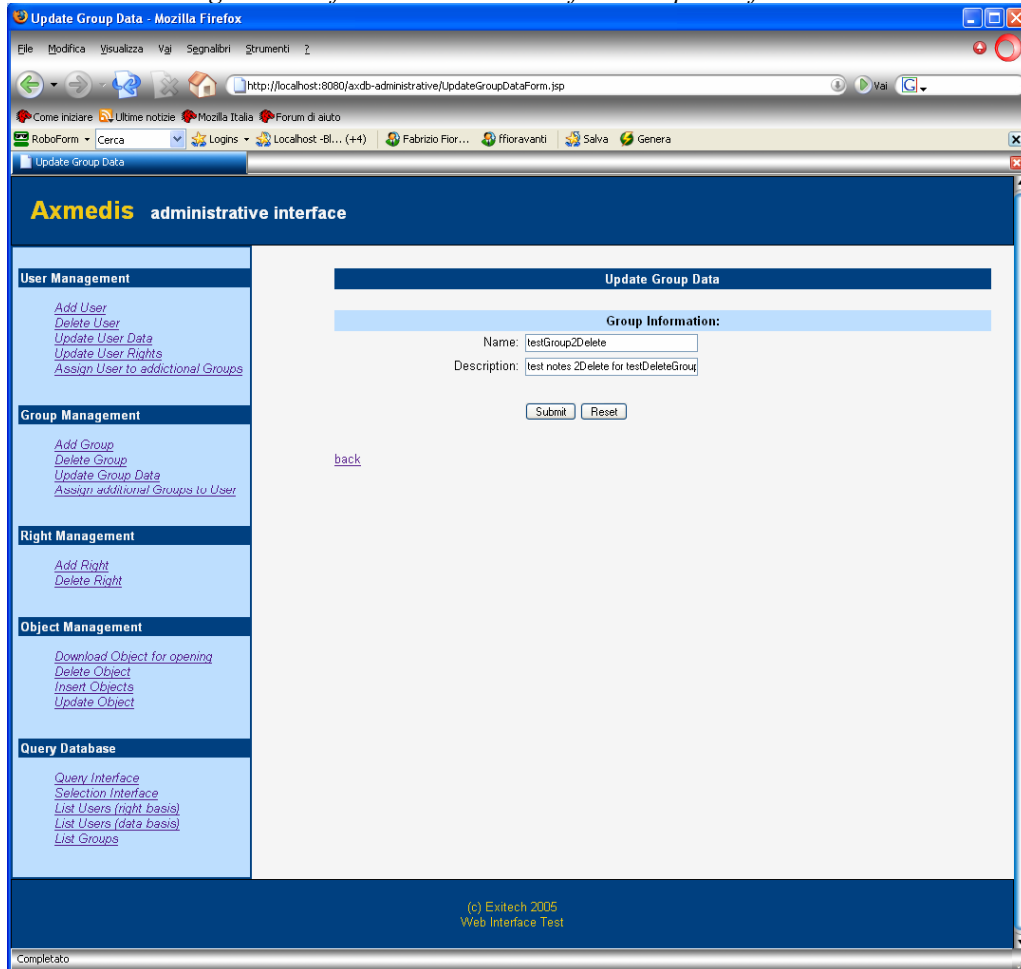


Update group data

This page allows the operator to update a group of related data. In the next version a list of groups will appear instead of the text input to insert the group ID.

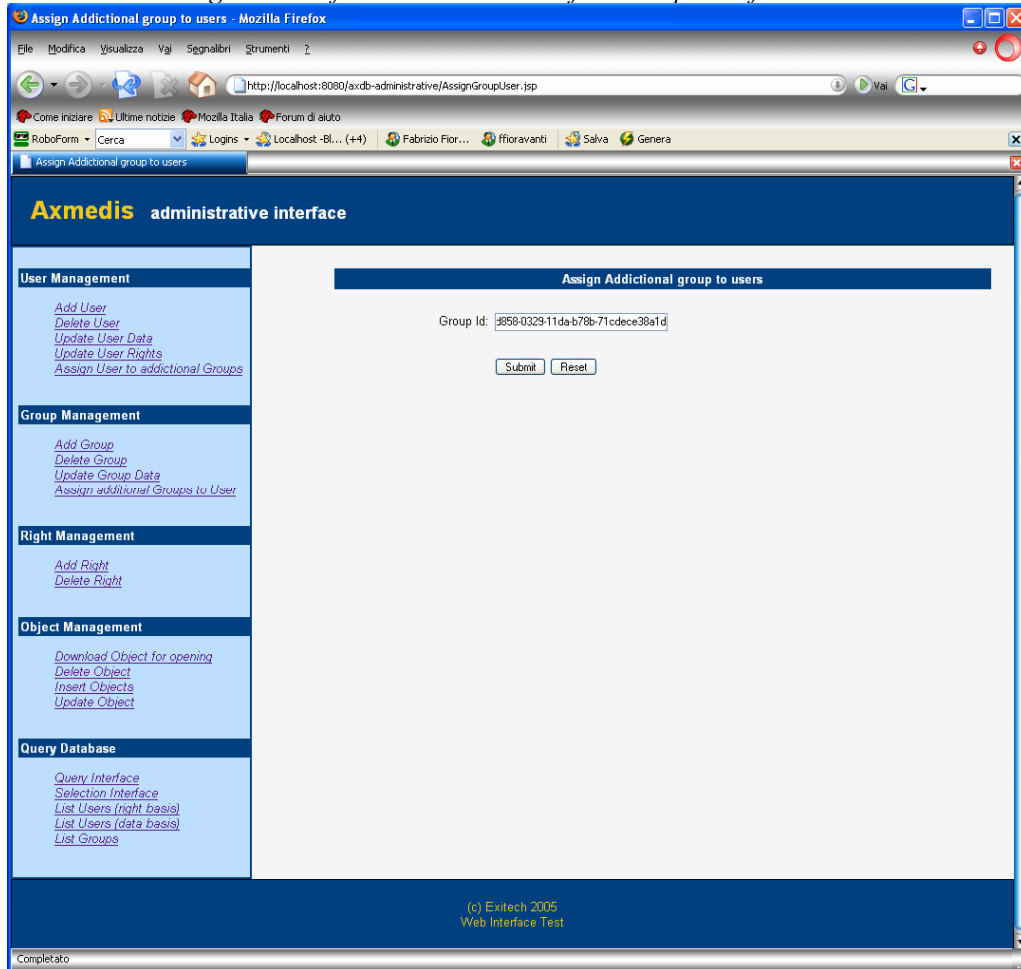


After choosing the group, the modifiable data will appear.

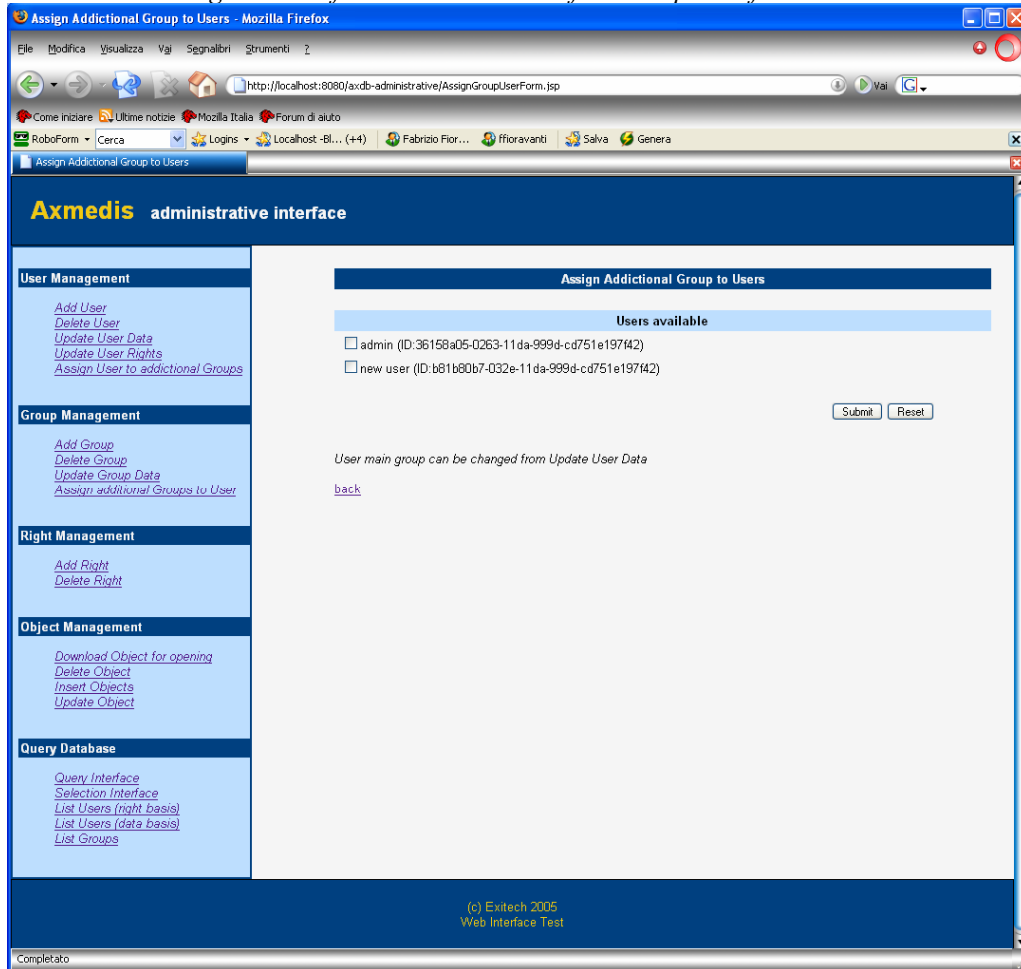


Assign additional users to group

This page allows the operator to assign one or more users to a group. In the next version a list of groups will appear instead of the text input to insert the group ID.



The user that already belongs to the group will be automatically checked.

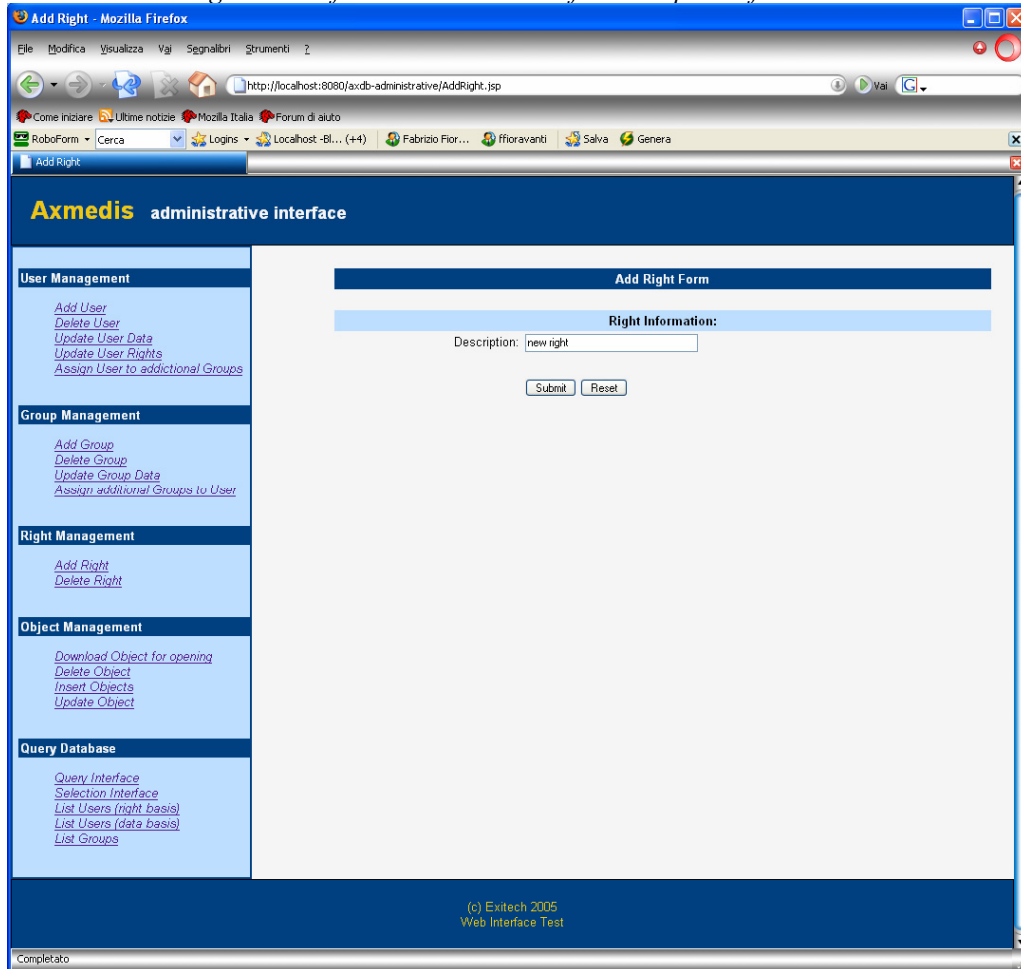


Right Management

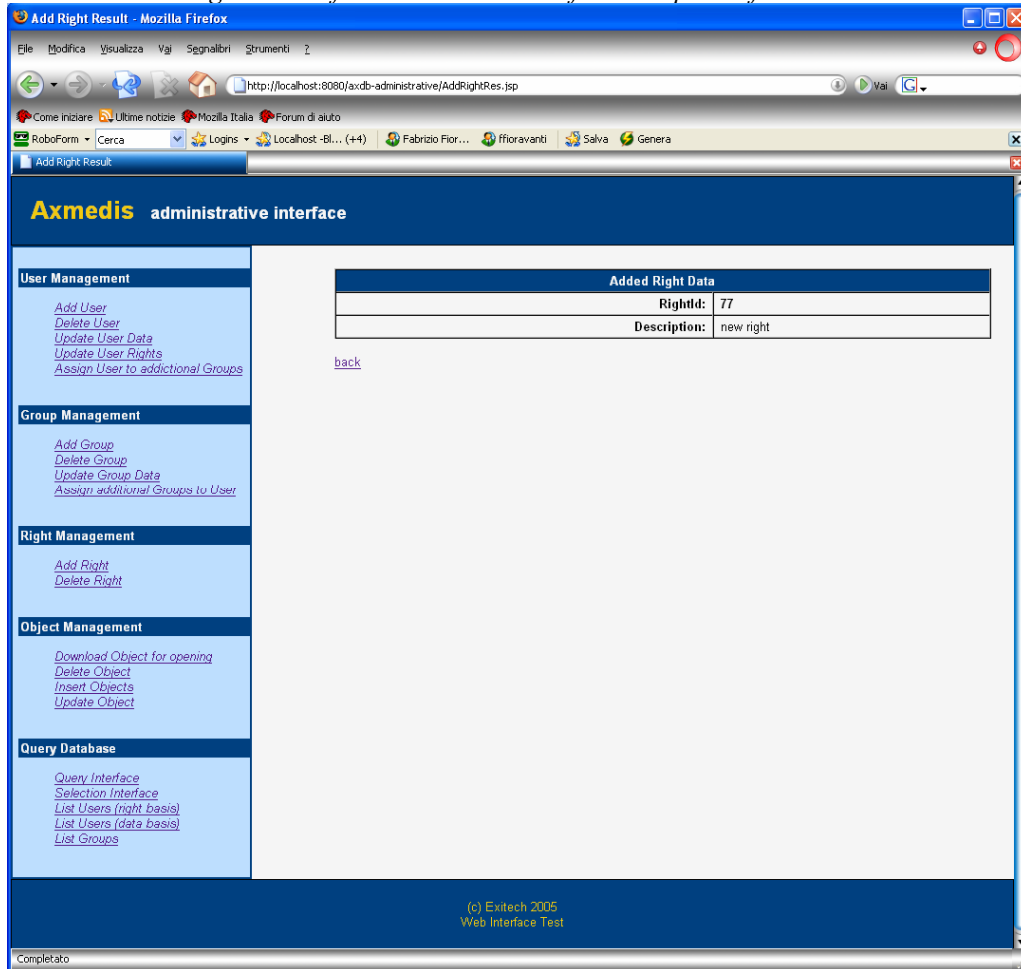
This section allows the operator to add and remove rights that can be assigned to the users.

Add right

This page creates a new right.

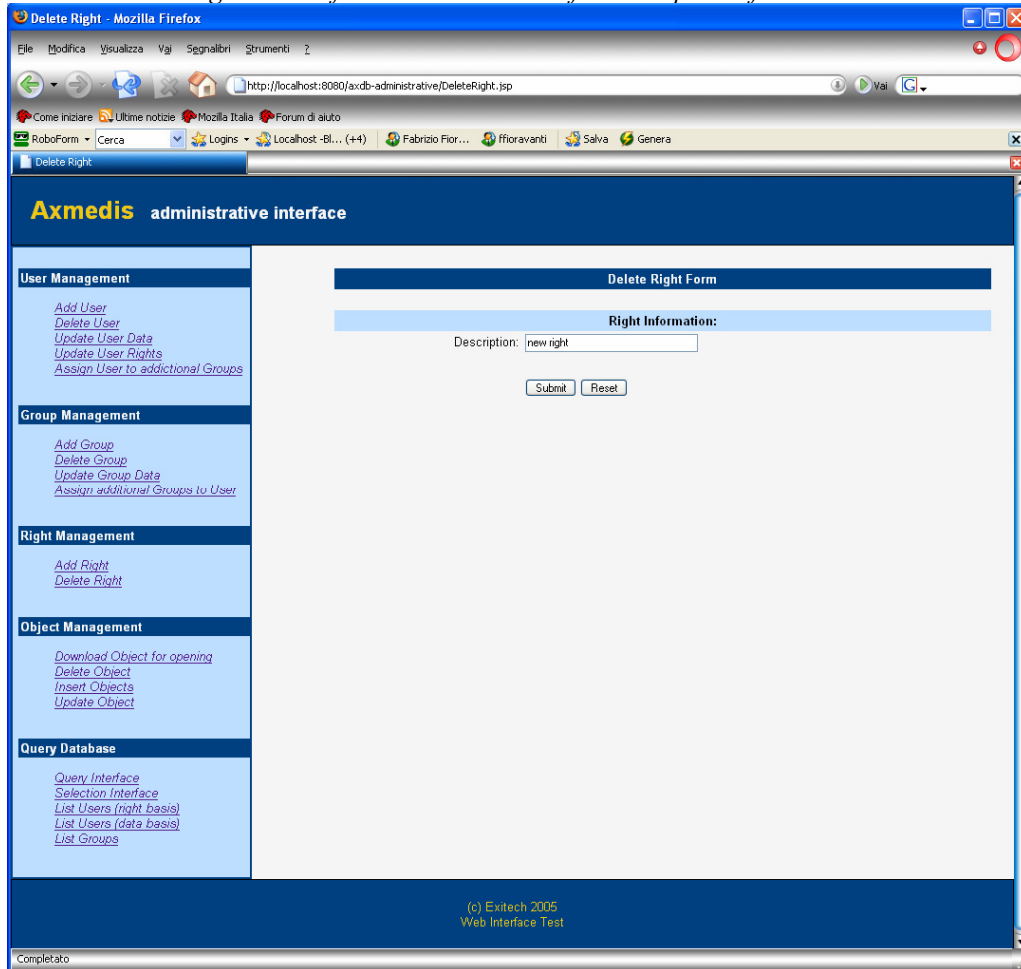


After the creation a summary screen is shown.



Delete right

This page, given the right name, allows the operator to eliminate such right from the DB.



Object Management

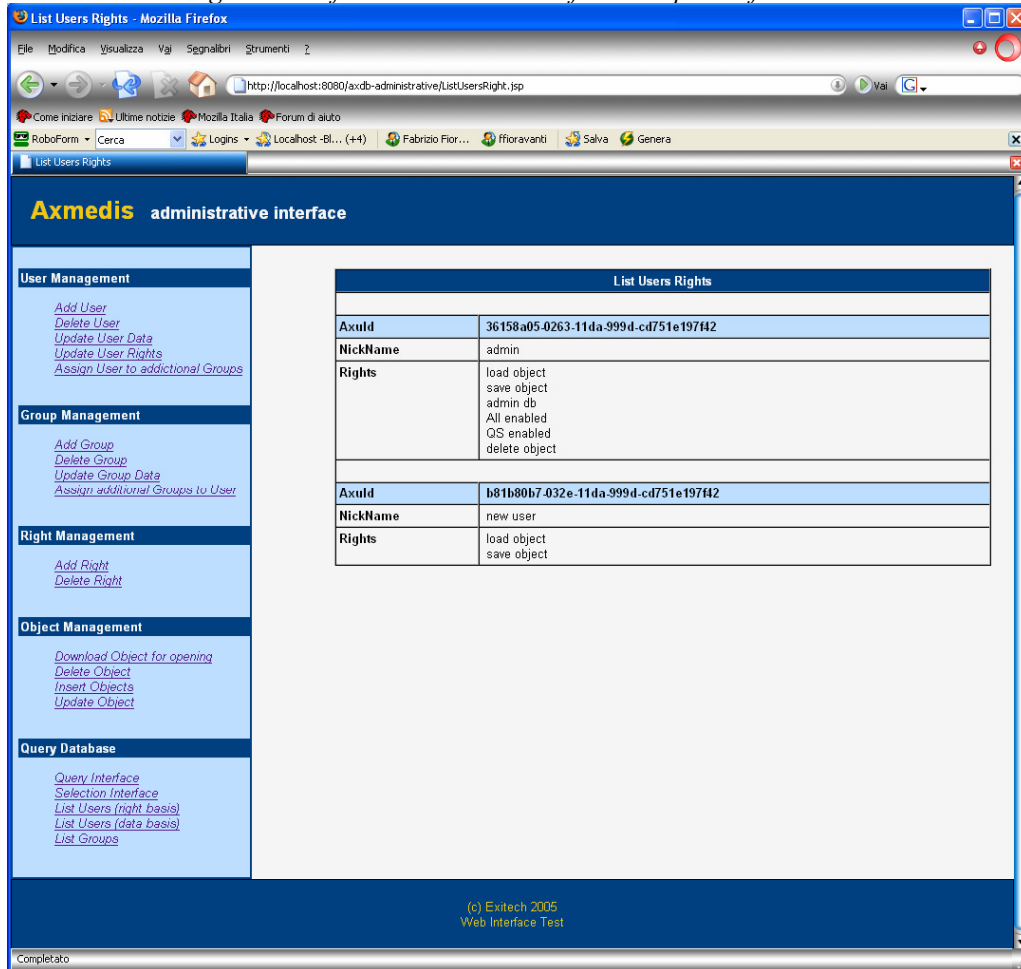
This section allows the operator to manage object (load/save/update/remove). It is mainly based on the Loader/Saver functionalities.

In addition to the menu item represented in the previous screen-snapshots, some other operations will be possible, such as:

- Mark an object as unavailable
- Delete one or more version of an object

List Users (right basis)

This page shows all the users that are present in the DB with their rights.



Axmedis administrative interface

User Management

- [Add User](#)
- [Delete User](#)
- [Update User Data](#)
- [Update User Rights](#)
- [Assign User to additional Groups](#)

Group Management

- [Add Group](#)
- [Delete Group](#)
- [Update Group Data](#)
- [Assign additional Groups to User](#)

Right Management

- [Add Right](#)
- [Delete Right](#)

Object Management

- [Download Object for opening](#)
- [Delete Object](#)
- [Insert Objects](#)
- [Update Object](#)

Query Database

- [Query Interface](#)
- [Selection Interface](#)
- [List Users \(right basis\)](#)
- [List Users \(data basis\)](#)
- [List Groups](#)

List Users Rights

Axuld	36158a05-0263-11da-999d-cd751e197f42
NickName	admin
Rights	load object save object admin db All enabled QS enabled delete object
Axuld	b81b80b7-032e-11da-999d-cd751e197f42
NickName	new user
Rights	load object save object

(c) Exitech 2005
Web Interface Test

List Users (data basis)

This page shows all the users that are present in the DB with their data.

Axmedis administrative interface

User Management

- [Add User](#)
- [Delete User](#)
- [Update User Data](#)
- [Update User Rights](#)
- [Assign User to additional Groups](#)

Group Management

- [Add Group](#)
- [Delete Group](#)
- [Update Group Data](#)
- [Assign additional Groups to User](#)

Right Management

- [Add Right](#)
- [Delete Right](#)

Object Management

- [Download Object for opening](#)
- [Delete Object](#)
- [Insert Objects](#)
- [Update Object](#)

Query Database

- [Query Interface](#)
- [Selection Interface](#)
- [List Users \(right basis\)](#)
- [List Users \(data basis\)](#)
- [List Groups](#)

List Users Data

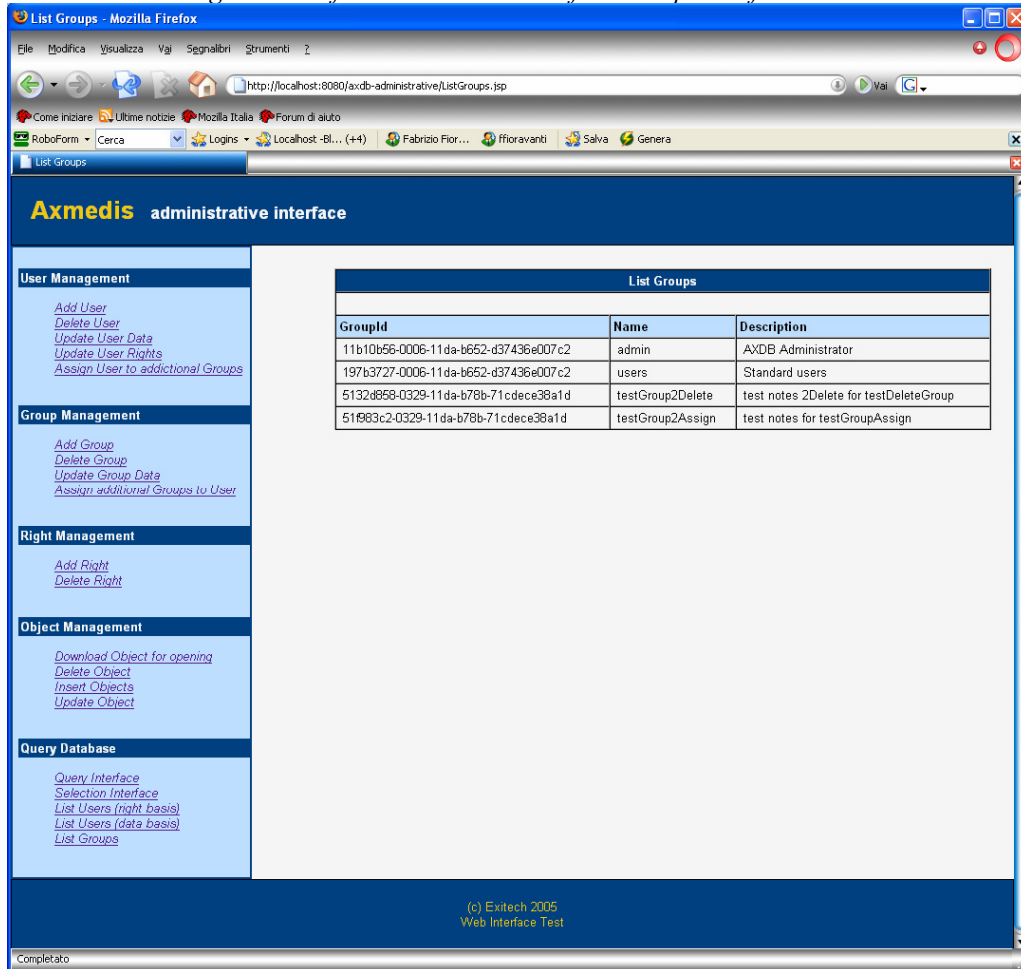
Axuld	36158a05-0263-11da-999d-cd751e197f42
NickName	admin
Password	axmedis
Email	admin@axmedis.org
Note	null
Main Group	admin (ID:11b10b56-0006-11da-b652-d37436e007c2)
Other Groups	users (ID:197b3727-0006-11da-b652-d37436e007c2)

Axuld	b81b80b7-032e-11da-999d-cd751e197f42
NickName	new user
Password	new password
Email	email@email.com
Note	null
Main Group	users (ID:197b3727-0006-11da-b652-d37436e007c2)
Other Groups	-

(c) Exitech 2005
Web Interface Test

List groups

This page shows all the groups that are present in the DB with their data.



Axmedis administrative interface

User Management

- [Add User](#)
- [Delete User](#)
- [Update User Data](#)
- [Update User Rights](#)
- [Assign User to additional Groups](#)

Group Management

- [Add Group](#)
- [Delete Group](#)
- [Update Group Data](#)
- [Assign additional Groups to User](#)

Right Management

- [Add Right](#)
- [Delete Right](#)

Object Management

- [Download Object for opening](#)
- [Delete Object](#)
- [Insert Objects](#)
- [Update Object](#)

Query Database

- [Query Interface](#)
- [Selection Interface](#)
- [List Users \(right basis\)](#)
- [List Users \(data basis\)](#)
- [List Groups](#)

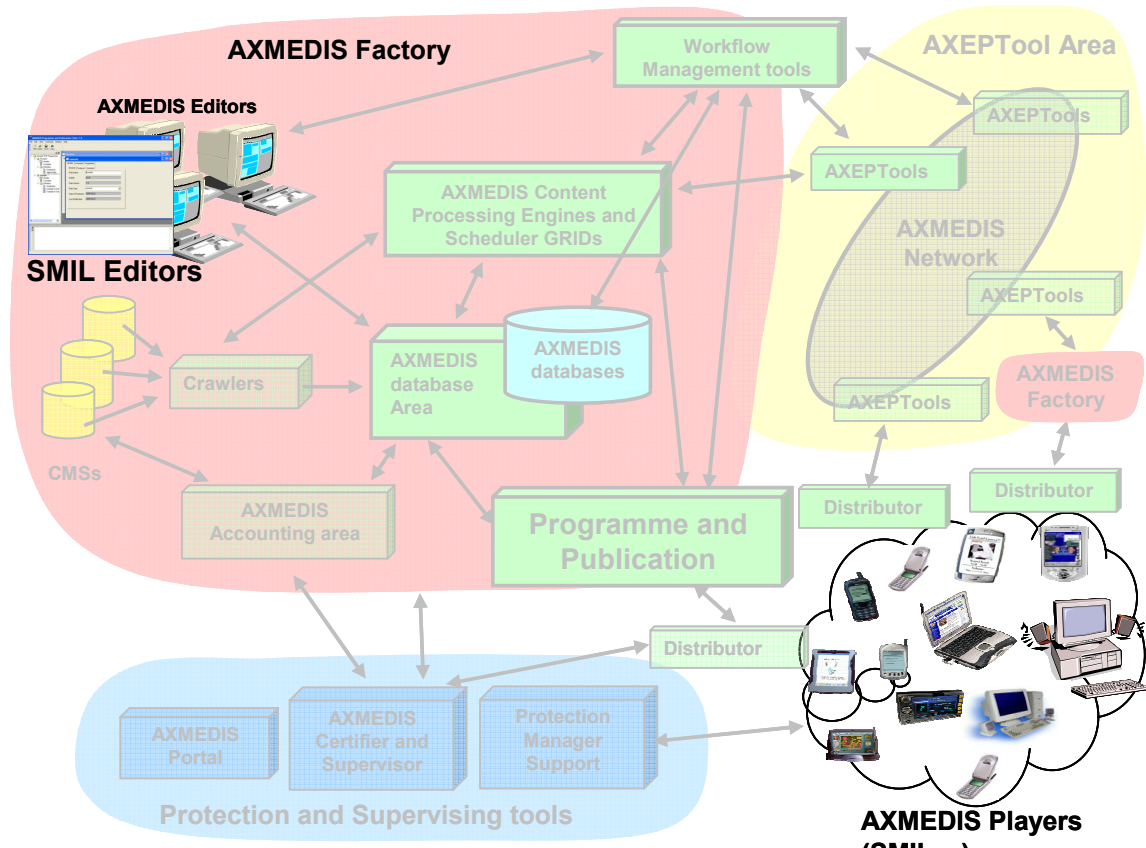
List Groups		
GroupId	Name	Description
11b10b56-0006-11da-b652-d37436e007c2	admin	AXDB Administrator
197b3727-0006-11da-b652-d37436e007c2	users	Standard users
5132d858-0329-11da-b78b-71cdece38a1d	testGroup2Delete	test notes 2Delete for testDeleteGroup
51f983c2-0329-11da-b78b-71cdece38a1d	testGroup2Assign	test notes for testGroupAssign

(c) Exitech 2005
Web Interface Test

Completato

6.2 Advanced Content Production Aspects

6.2.1 SMIL Editor and Player



When AXMEDIS users (content authors) want to create their own multimedia content, they will use the Synchronized Multimedia Integration Language (SMIL). They have to use a SMIL-based authoring environment to integrate streaming audio and video with images, text or any other media types. The following diagram resumes the functionality provided by a typical SMIL authoring environment in which AXMEDIS users can make a production of new media content .

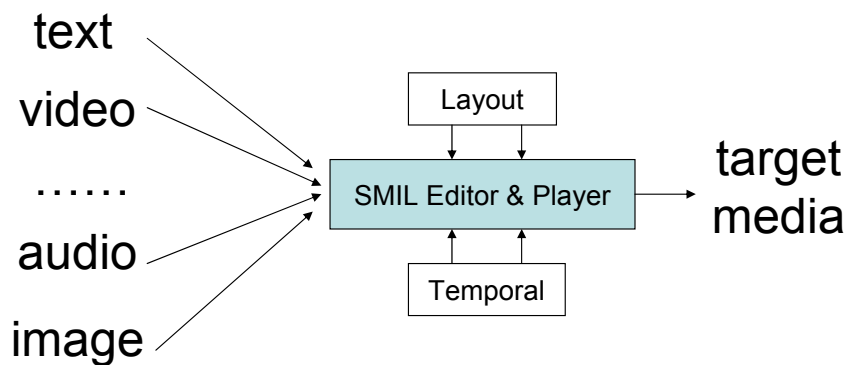


Figure 1 - Functional diagram of a SMIL authoring environment

The SMIL editor and player provide a flexible authoring environment for AXMEDIS users. An AXMEDIS User can use this toolset to create a new complex media scene, starting from media objects of different natures, e.g. video, audio, image, text, etc.

The SMIL editor and player is composed by a Visual editor, a Behavioral editor, a Description editor, and a Player.

AXMEDIS users use the Behaviour and Visual Editors to organize the media objects in space and time by providing an easy “mouse-driven” interface to arrange and manipulate video, pictures, text, etc on the screen. A user can use the toolbox of the visual editor by visually placing rectangular shapes within a visual environment. He can place and resize a rectangular shape by clicking the mouse, holding on and then dropping the shape on the canvas. Therefore, the user can choose the place for the media object. After that, the user can add a media object to the rectangular area by right clicking. There will be a new menu popping up with functions of “Add”, “Open”, etc. The screenshot below shows how a user can draw a shape in a certain area and then associate a media object to this area in a 2-D environment. Moreover, it permits the users to manage (i.e. move, delete, add, etc...) subparts of the target object which has spatial properties or constraints.

When an AXMEDIS user wants to choose the display time for the media object. S/he will use the Behavioural editor to place media objects along a timeline and finely tune events to be triggered during playback. He can tune the period simply by clicking the tag along the timeline and make any modifications along the timeline. The exact moment in time in which a given decoding process has to start can be specified by the user in two ways. One is graphical and permits to drag & drop along the timeline the shapes representing media object; the second one is “numerical” and permits to directly enter the numerical value of the time in which a given event has to be triggered.

When an AXMEDIS user wants to add some description and comment for each media object, he can will use description editor. He can use this editor by right clicking the rectangular area and adding comments and descriptions for the corresponding media object.

After editing, the user can preview (decode) the result by using the SMIL player and also save the whole scene as a SMIL file by clicking “Save” or “Save as...” in the “File” menu of the integrated player. AXMEDIS users can immediately see the result of their work in the very same environment they use for content creation and manipulation. It is “what you see is what you get” authoring environment.

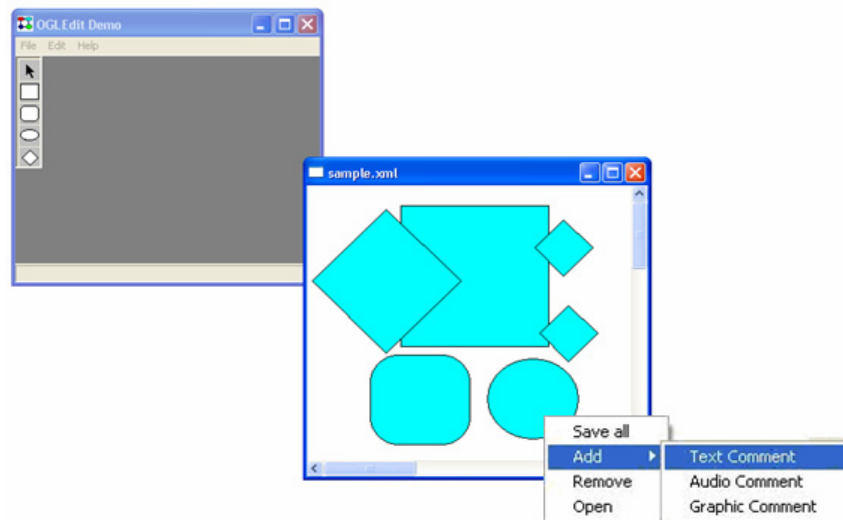


Figure 2 - Visual editor with contextual menu used to add annotations

After each editing stage, an AXMEDIS end-users can will use the SMIL player introduced above and shown below to decode and render media content on their multimedia terminals (be they PCs or portable devices). The player interface is very close to those of available commercial players, end users He can will use this player access AXMEDIS content by clicking “File” and then click “Play”, “Stop”, “Pause”, etc to control the process of display. By clicking the relative options of the menu, the user can also open some media objects

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools and save or save with a different name in some directories. Also the user will have the options of “rewind”, “forward”, etc from the menu. Enhanced versions of the player will provide more functionality like creating a list of favourite media objects for specific user, searching media objects or relative webpage linking.



Figure 3 - SMIL player interface

6.2.2 Adaptation (FHGIGD, DIPITA, EPFL, etc)

Audio Adaptation:

The audio adaptation tool provides a number of functionalities to change audio file formats and resolutions. These functionalities of the audio adaptation tool will be presented by taking a high-quality uncompressed audio file and by generating a number of variations of this file. Of course, all of these functionalities can be used in a single command line but for the sake of simplicity, we demonstrate each functionality independently.

How to down-sample a file: assume that the original file `InputFile` is of CD quality i.e. it has a sampling rate of 44100 Hz, the tool can be used to down-sample the file to other typical sampling rate, for example 11025 Hz so that the resulting file `OutputFile` is 4 times smaller:

```
AudioAdaptation InputFile -r 11025 OutputFile
```

How to mix channels of a file: assume that the original file `InputFile` has multiple channels, the tool can mix down channels to built a mono signal so that the size of the resulting file `OutputFile` is N times smaller where N is the number of channels of the original file:

```
AudioAdaptation InputFile -c 1 OutputFile
```

How to change the format and codec of a file: the original file can be saved in another format (WAV to .AIFF for example), the original file can be saved in the same format but with a different codec (for example a WAV file with 16 bits samples saved .as a WAV file with 8 bits samples) and the original file can be saved in a different format and with a codec allowing for compression (WAV to MP3 with a bandwidth of 128kb/s for example). This first example shows how to save an AIFF input file `InputFile.aif` as a WAV file with unsigned 8 bits PCM encoding of samples:

```
AudioAdaptation InputFile.aif -codec pcm_u8 OutputFile.wav
```

This second example shows how to encode a WAV file `InputFile.wav` as a MP3 file `OutputFile.mp3` with 128kb/s bit rate:

```
AudioAdaptation InputFile.wav -codec mp3 -b 128 OutputFile.wav
```

How to select a specific part of an audio file: apart from these adaptations aiming at reducing the size of the resulting audio file by changing its quality, the tool can create a new file by using only a small part of the original file and add fading to the corresponding excerpt to provide smooth start and end. By combining these selection strategies with change of resolution, it is possible to create reduced bit-rates snapshots of high-quality audio files in a single operation. The following command demonstrates how to take from the `InputFile` a slice of 30 seconds of audio data starting 45 seconds after the beginning file and save it as `OutputFile` :

```
AudioAdaptation InputFile -st 45 -d 30 OutputFile
```

Multimedia adaptation:

The multimedia adaptation tool provides a number of functionalities to adapt so-called rich media content. By rich media content, we mean content with local and client-server interactivity, natural audio and video (ex MPEG-1 and MPEG-2) mixed with synthetic audio (MIDI, MPEG-4 Structured Audio), 2D and 3D graphics, web content, etc...

1. File conversion: we will show how to convert multimedia files from one format to another (for example, convert a Macromedia Flash file into a MP4 file).
2. Extraction of media files: we will demonstrate how to extract single media files from complex multimedia files; for example this becomes crucial when one needs to send a multimedia file to a device only supporting audio output.
3. File concatenation: we will show how to concatenate a number of different files into a single container: for example, how to cat a set of AVI videos into a single MP4 file.

Video Adaptation

The work performed focused on the (integration of) transcoding algorithms. To show AXMEDIS' capabilities in terms of content adaptation as well as extensibility the FFMPEG library has been selected for the implementation/integration of the firstly available video adaptation functionality.

Via command line different video options are available, e.g.:

-b bitrate	set video bitrate (in kbit/s)
-r rate	set frame rate (Hz value, fraction or abbreviation)
-s size	set frame size (WxH or abbreviation)
-aspect aspect	set aspect ratio (4:3, 16:9 or 1.3333, 1.7777)

Besides the basic functionality so called “advanced options” are available:

-pix_fmt format	set pixel format
-g gop_size	set the group of picture size
-intra	use only intra frames
-qscale q	use fixed video quantiser scale (VBR)
-qmin q	min video quantiser scale (VBR)
-qmax q	max video quantiser scale (VBR)
-mbqmin q	min macroblock quantiser scale (VBR)
-mbqmax q	max macroblock quantiser scale (VBR)
-qdiff q	max difference between the quantiser scale (VBR)

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

-qblur blur	video quantiser scale blur (VBR)
-qcomp compression	video quantiser scale compression (VBR)
-rc_init_cplx	initial complexity for 1-pass encoding
-4mv	use four motion vector by macroblock (MPEG4)
-obmc	use overlapped block motion compensation (h263+)
-part	use data partitioning (MPEG4)
-strict strictness	how strictly to follow the standards
-deinterlace	deinterlace pictures
-ildct	force interlaced dct support in encoder (MPEG2/MPEG4)
-ilme	force interlaced me support in encoder MPEG2
-aic	enable Advanced intra coding (h263+)
-aiv	enable Alternative inter vlc (h263+)
-umv	enable Unlimited Motion Vector (h263+)
-alt	enable alternate scantable (mpeg2)
-trell	enable trellis quantisation
-nr	noise reduction

The available functionality addresses two categories of video adaptation:

- Considering the (fixed) needs of the content creator or service providers: The content can be adapted according to the requirements of the content creator or service providers. The simplest case is the adaptation of the spatial resolution as supported to a fixed resolution without changing the aspect ratio.
- Dynamic content creation: The tool allows changing the aspect ratio of the video content. Thus, a basic functionality required to dynamically adapt content according to the receiving device is available.

6.2.3 Fingerprint (FHGIGD, DIPITA, EPFL, etc)

Audio Descriptors Extraction:

To present the functionalities of the audio descriptor extractor within the AXMEDIS framework, three cases will be considered:

1. How to extract textual descriptors from the audio data given an extractor model.
2. How to train the algorithm to solve a new problem, i.e. to allow for extracting automatically new textual descriptors from the audio data.
3. How to use the abstract numerical descriptors extracted from an audio file to retrieve similar items.

Extracting textual descriptors from audio:

The audio descriptor extractor tool is able to associate automatically keywords describing content to the audio data. For example, one may want to associate genre labels to his/her collection of audio files. Given an input model properly trained to recognize a certain set of music genre labels (see next paragraph for the training of a model), the audio descriptor extractor will analyze the audio file and try to characterize it with a genre label known by the model. If the specified audio file does not correspond to a genre label known by the model, it will be marked as unknown. The following command line takes `AudioInputFile` as input (compressed or uncompressed audio file), extract its descriptors and save them in `DescriptionOutputFile`. The model used is specified with the `-m` option.

```
AudioDescription AudioInputFile -m genre.model DescriptionOutputFile
```

Training the algorithm to extract new textual descriptors:

The extraction of textual descriptors from audio data relies on a supervised machine-learning scheme. In other words, one needs to provide a set audio files along with their corresponding labels so that the algorithm will learn automatically how to associate musical characteristics to labels. Taking the example of genre recognition, one needs to provide to the algorithm a set of audio files along with their corresponding musical

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

genres. The algorithm will learn a mapping from the acoustic data to the set of genres specified in the training set. If a new song appears to be too *different* from the examples of the training set, it will be given the label unknown. Notice that the training is a very time consuming task and that the training set must be carefully built to provide the best classification results. The following command line shows how to train the model `genre.model` based on the list of annotated files in `AudioFileInputList`:

```
AudioDescription -training AudioFileInputList genre.model
```

The annotated input file list should be specified with the following syntax:

```
<Audio File Location 1>\t<Associated Keyword>
<Audio File Location 2>\t<Associated Keyword>
<Audio File Location 3>\t<Associated Keyword>
...
```

In the case of genre recognition, the keyword to associate with an audio file will be a genre label (such as Rock or Blues).

Using audio similarity for retrieval:

Apart from textual descriptors that make sense to human users, the audio descriptor extractor algorithm is able to extract sets of abstract numerical features from the audio data. These features are used to compare two different audio files and to characterise their similarities in some musically meaningful space. More precisely, one can use the algorithm to measure the similarity between audio files according to timbre, rhythmic content, harmonic content or everything together (general audio similarity). Audio similarity can be used for music recommendation, automatic playlist generation or intelligent organization of music collections. The following command shows the similarity between the files `AudioFile1` and `AudioFile2`. The more similar the 2 files is, the closer to 1 will be the similarity, while the more different the 2 files is, the closer it will be from 0.

```
AudioDescription -similarity AudioFile1 AudioFile2
```

Descriptors Extractor from Text Documents:

The purpose of Descriptor Extraction from Text Documents is to automatically retrieve high level metadata from open-domain texts, such as keywords (single or multi-word), main language and the semantic domain of the document.

High level metadata will be used to improve indexing of the text documents content.

The Descriptor Extractor from Text Documents plug-in functionalities will be shown using a command line interface executable. It will present three kind of results:

- a set of single keywords extracted using a state of the art algorithm (TF.IDF) settled in a suitable way for translation
- a revised set of keywords (including multi-words and proper nouns) extracted using a new algorithm
- the main argument of the document

The executable can be run from a console and accept 2 parameters:

1. The name of a English plain text file which is the document to be analyzed
2. A positive integer number defining the maximum number of keywords requested

As an example the following command will extract metadata (not more than 10) from the plain text file called `document1.txt`

```
> DescriptorExtractor document1.txt 10
```

Fingerprint Extractor

The automatic fingerprint extraction addresses two different issues within AXMEDIS: content identification and content verification. For the automatic content identification the content (or parts thereof) that has to be identified is given in the query. The content may have undergone typical processing operations like compression. For the content verification, a fingerprint is calculated and stored in a database. During the verification process the fingerprint is calculated for the object that has to be verified. The resulting fingerprint is compared to the one stored in the database.

Fingerprint Extractor for Audio:

The Descriptor Extractor for audio content will be shown with a MatLab prototype implementation. Different functionalities can be experienced:

- Loading, processing and saving audio content
- Playing video content
- Calculation of the fingerprint for the given audio files
- Comparison of two identical or different audio files based on the calculated fingerprints for verification of the content
- Identification if one audio content is part of another to exemplify the search capabilities

For simplicity of the demonstration a GUI was developed.

Fingerprint Extractor for Images:

The Descriptor Extractor for image content will be shown with a MatLab prototype implementation. The functionalities that can be experienced are:

- Loading the images
- Calculation of the fingerprint for the given image files
- Comparison of the resulting fingerprints

For simplicity of the demonstration a simple GUI was developed.

Fingerprint Extractor for Videos:

The Descriptor Extractor for video content will be shown with a MatLab prototype implementation. Different functionalities can be experienced:

- Loading video content
- Rendering the video content
- Calculation of the fingerprint for the given vide content
- Comparison of the resulting fingerprints

For simplicity of the demonstration a simple GUI was developed.

6.2.4 Workflow (IRC, HP, XIM)

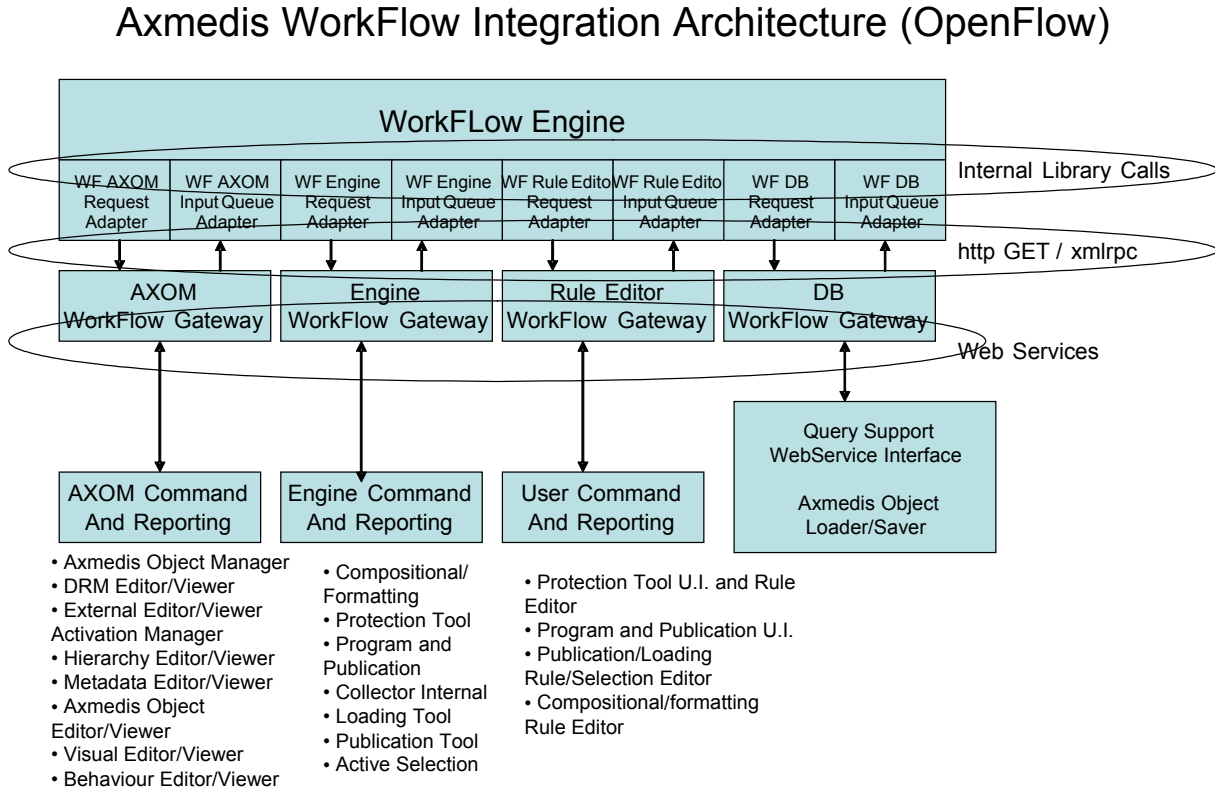
The demonstration of the Workflow activities and control over AXMEDIS tools will be based on a hypothetical production scenario involving two partners. This scenario is based on the scenarios mentioned in DE3.1.2 Part G, however are modified to reflect the current state of the work.

Introduction

Workflow is high level definition of activities, which when carried automatically and/or manually out either by human actor and/or tools in the defined order produces the desired output. Workflow tools provide an interface & glossary to define such workflow and enables users to enact and monitor the progress of all the activities. It thus provides a centralised mechanism for controlling the productions of a factory.

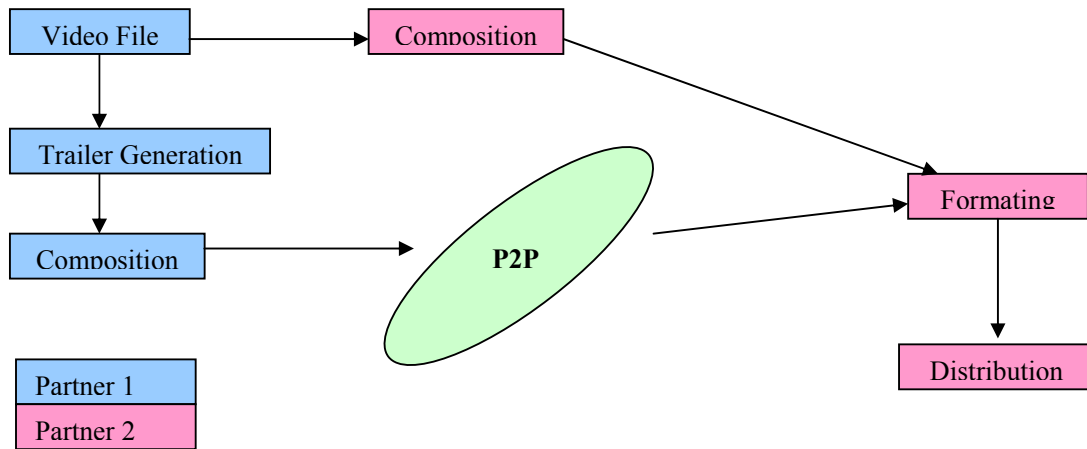
For the case of AXMEDIS, the workflow will form a backbone for the automation of various production and distribution activities by inter-acting with various AXMEDIS tools. For the first phase of the project, we are integrating Openflow (www.openflow.it), an open source activity based workflow engine with various AXMEDIS tools that can be broadly classified into four types namely: Object Editors, Rule Editors, Engines and Databases.

This integration is realised as per the following Integration architecture designed during the analysis and design phase of the project.



Scenario:

In order to demonstrate the functionalities and usefulness of the workflow engine within the AXMEDIS framework the following scenario will be enacted. This scenario depicts a typical AXMEDIS production chain wherein an object produced by one partner is used by another, thus it is a inter-factory workflow.



Sample Workflow involving two partners

The above workflow involves two partners at two different factories. The partner 1 selects one object and produces a modified object from the original using AXMEDIS Editor. As an example, this could be the generation of a trailer from a selected video file. He then publishes it over the P2P network using the publication engine of AXEPTool and thus exposed to other partners.

The partner 2 downloads this trailer video from the P2P network using the loading engine of AXEPTool along with the original object and if required some other object from the database. He then uses these two objects to form a composite object and may apply some further formatting rules using the AXMEDIS composition and formatting engine before the final object is ready to be distributed to the end user.

In order to demonstrate this workflow, specific instructions will be written in the openflow engine, which is the open source workflow management system that has been interfaced with AXMEDIS tools for the first phase of the project. The user will be able to log-in to the openflow engine and will see this workflow as an active workflow along with the list of pre-defined, standard work activities he is supposed to act on. He will be then prompted by the workflow environment to carry out the required operations in order to complete the chain.

Openflow runs on the Zope platform which is managed through the “Zope Management Interface” using industry standard browsers, typically by logging on as the administrator (admin) at URL: <http://localhost:8080/manage>. The screen shot below shows an example of this management interface.

Creation of Workflow

Creating a new process in openflow is a multi-step process which begins with adding an OpenFlow container using the Zope management interface as shown below.

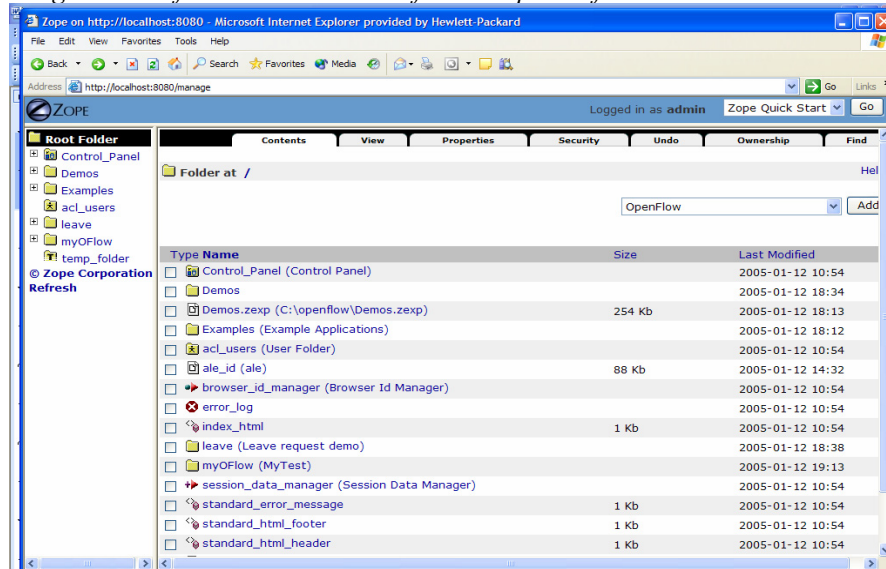


Figure 4: Adding an OpenFlow container through the Zope Management Interface

During the creation of the OpenFlow container, the name of the container must be specified as shown in the next screen-shot.

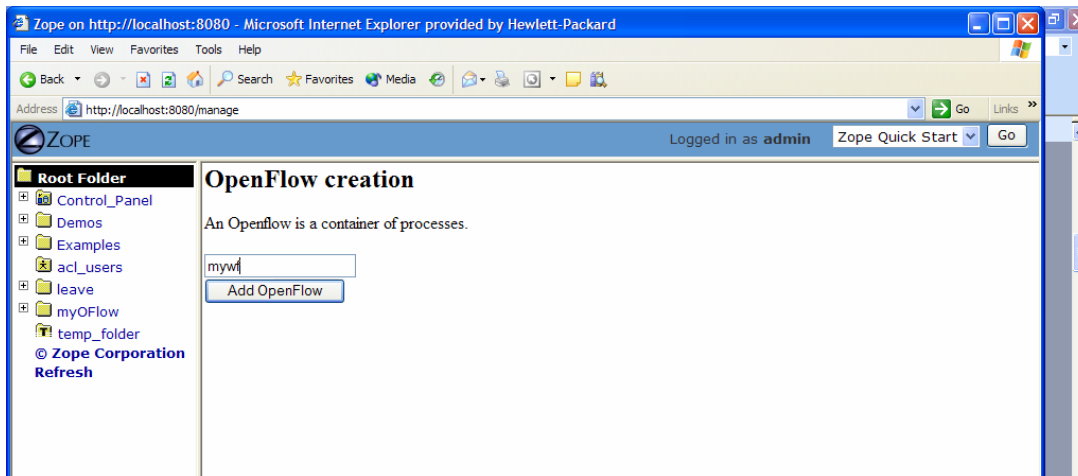


Figure 5: Creating the OpenFlow container

Defining Activities & Transitions

Next it is necessary to define the process and the activities pertaining to the process, together with their transitions (From Activity and To Activity). These operations are performed by accessing the tabs in the Openflow container as shown in the following screen-shots:

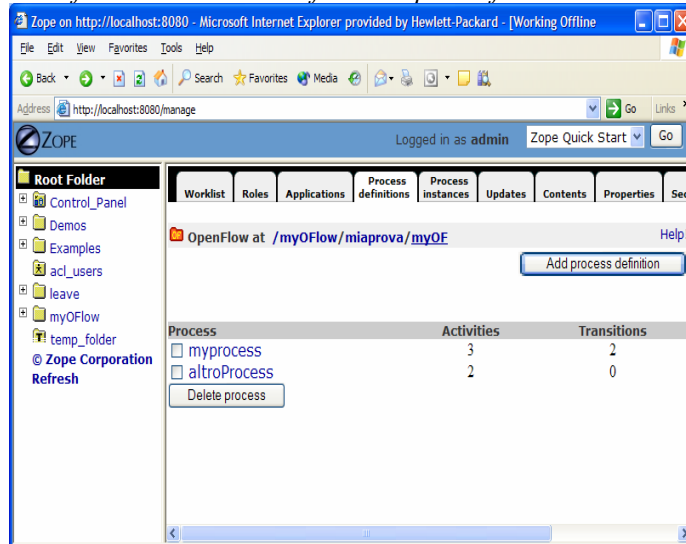


Figure 6: The process definition tab

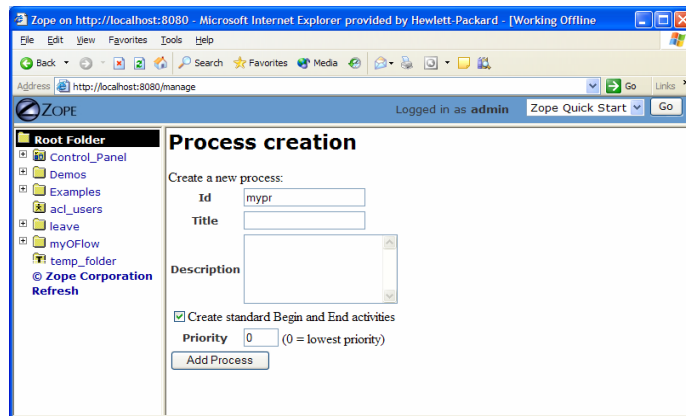


Figure 7: Creating a new Process definition

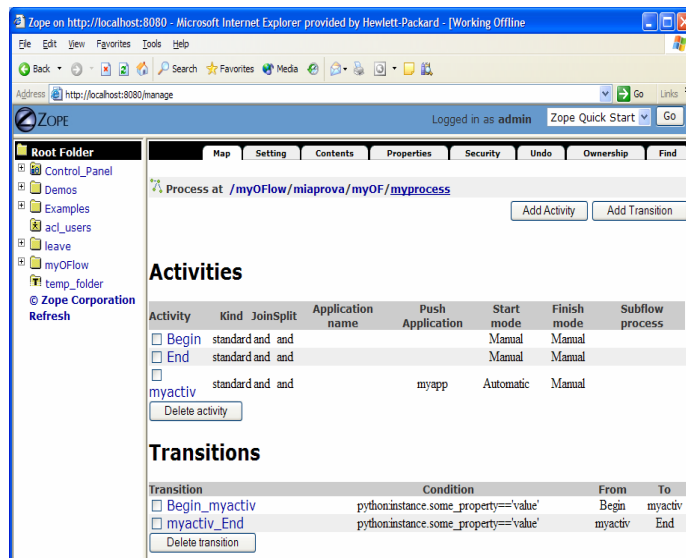


Figure 8: Management of activity and transitions of a process

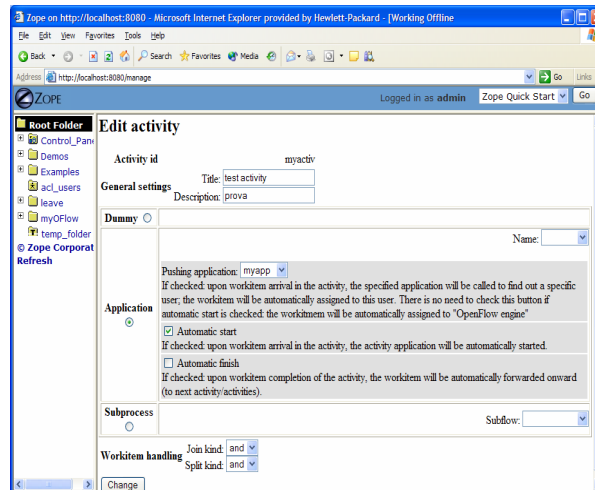


Figure 9: Editing a process activity

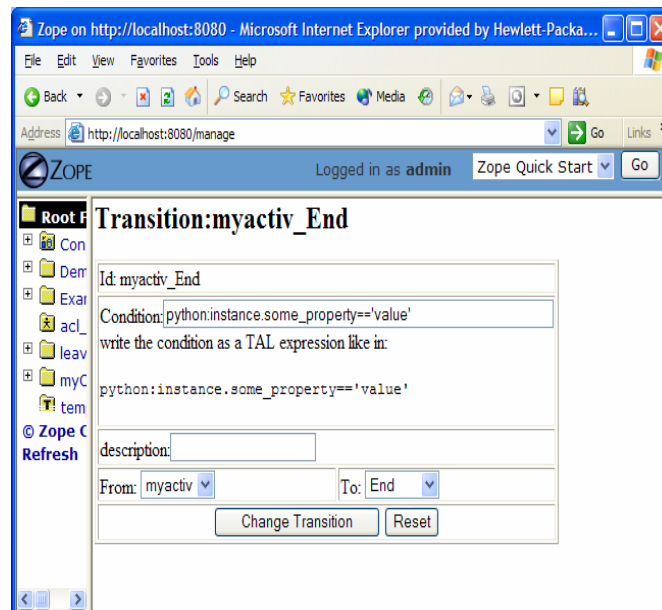


Figure 10: Defining process transition and related conditions

Applications associated to the activities are then specified selecting the Applications Tab.

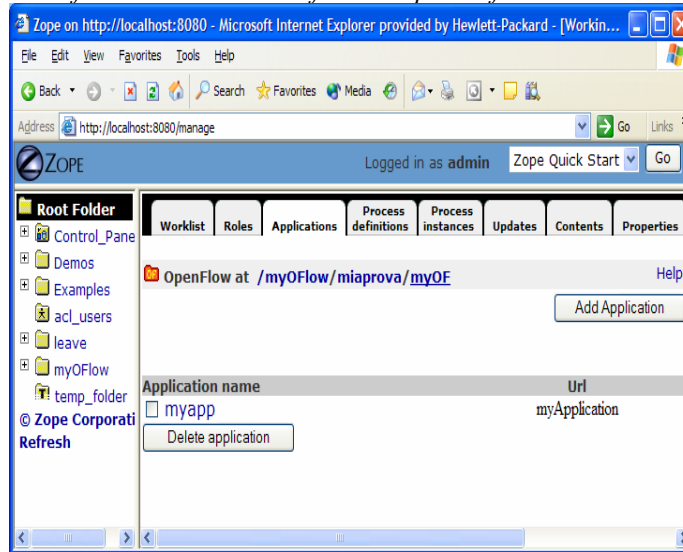


Figure 11: Defining process applications

The users and roles are configured as Zope users and roles as access control list (acl_users).

Activating a Workflow

Once a process has been defined it can be tested. An instance of the process can be created and executed directly in the processflow-instance management tab shown below.

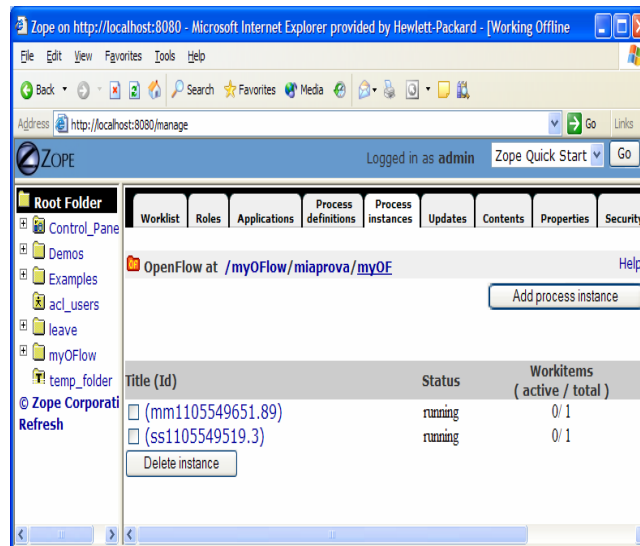


Figure 12: Process instance management tab

The following Figure shows the of the workitems involved in the process instance that has been created.

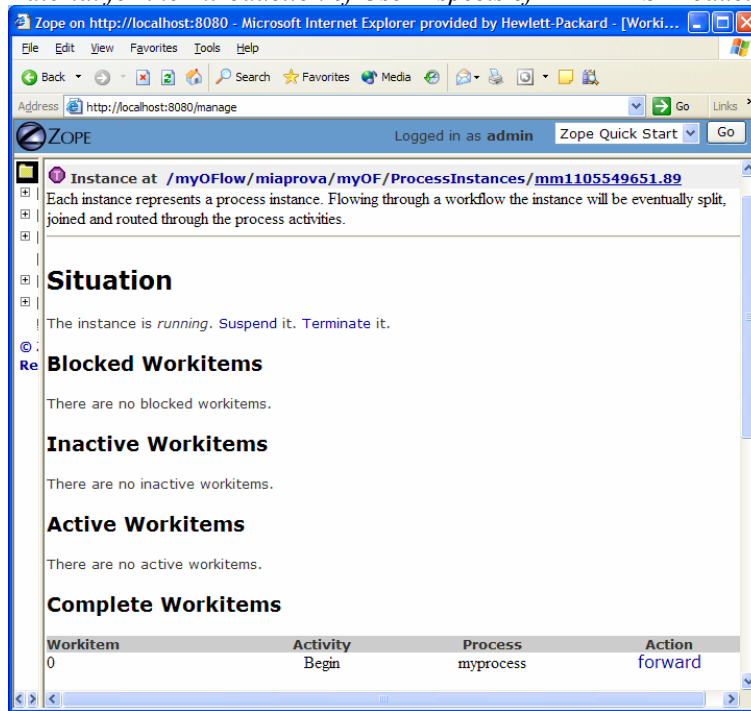


Figure 13: Monitoring and management of a specific process instance

Process Example:

The following simple example illustrates a process to request an AXMEDIS object manipulation (*a mock-up process*). This is an example of explicit forwarding to different actors having different roles. The first actor requests the creation of a new AXMEDIS object by filling out a form. The request goes to the second actor (called Socius) who checks that the request is acceptable. The request is then forwarded to the third actor (called Prefectus) for approval.

The following steps are necessary for the above example process to be enacted:

The first actor (called Tertius) enters an AXMEDIS object manipulation request by filling out the following form as shown in the screen-shot below:

Leave request demo - Microsoft Internet Explorer provided by Hewlett-Packard

File Edit View Favorites Tools Help

Address http://localhost:8080/leave/leave_startform Go Links »

Request for AXMEDIS object manipulation

Start date End date

Type

Reason

Figure 14: Tertius' AXMEDIS object manipulation form

According to the processflow, the request goes to the next actor (called Socius). When Socius logs in, his work list shows that there is a workitem in his worklist as shown in the screen-shot below:

Leave request demo - Microsoft Internet Explorer provided by Hewlett-Packard

File Edit View Favorites Tools Help

Address <http://localhost:8080/leave/mywork> Go Links »

Worklist for socius

Activity	Instance	Status	Actor	Action
Begin	Request of AXMEDIS object manipulation by tertius	inactive	socius	Activate

[To frontpage](#)

Figure 15: Socius' worklist and workitem activation

To execute the workitem, the actor (Socius) has to activate the workitem (Begin) and perform the related activities. Next this actor either forwards the workitem to the next actor, which in this case is the supervisor (called Prefectus), or rejects the request; as illustrated by the screen-shot below:

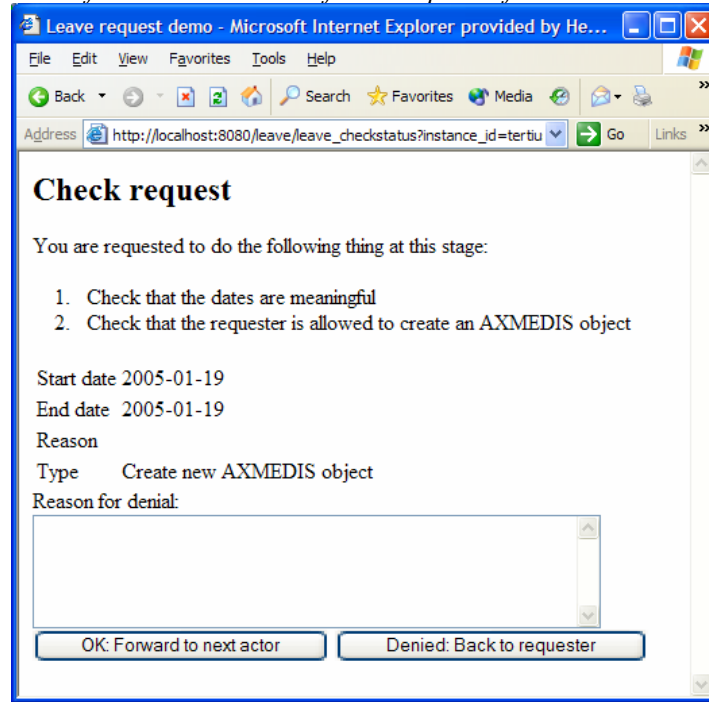


Figure 16: Socius' workitem execution and forwarding

Then the activity is forwarded to the last actor and the process ends.

Conclusion

Using the Workflow Tool, the project manager can define a workflow consisting of various activities and can assign roles/actors responsible for them. These actors are presented with their worklist by the workflow interface. They can carry out these activities as instructed by the workflow engine. The manager can continuously monitor the progress of the workflow and can take corrective actions, if required. The workflow is also able to interact directly with various AXMEDIS tools and hence the activities can be defined as autonomous requiring no human intervention and thus can be started and completed automatically. Thus the workflow tools are able to further increase the automation of various AXMEDIS tasks and thus minimising the costs.

Learning Outcomes

At the end of this tutorial session, the project managers will understand:

- The importance of workflow to streamline activities and reduce the cost of its execution & monitoring.
- How to use the workflow interface to define the workflow, its activities and assign roles to them.
- How to enact the workflow for production
- How the workflow will interact with various AXMEDIS tools and thus can control them.

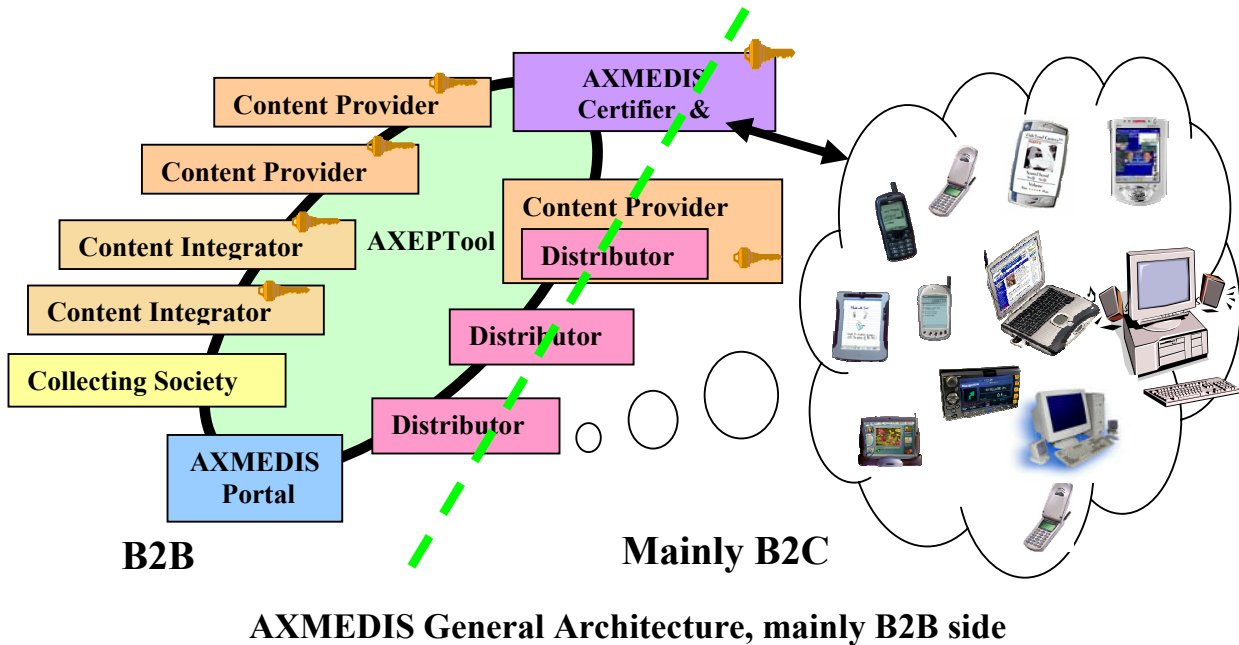
6.3 Content Distribution and Supporting Tools

6.3.1 AXEPTool (CRS4)

The purpose of this demonstrator is to show the shaped P2P Network and its main functionalities, and to describes the features of the AXEPTool prototype currently being developed,

6.3.1.1 Contextualisation of the AXEPTool area

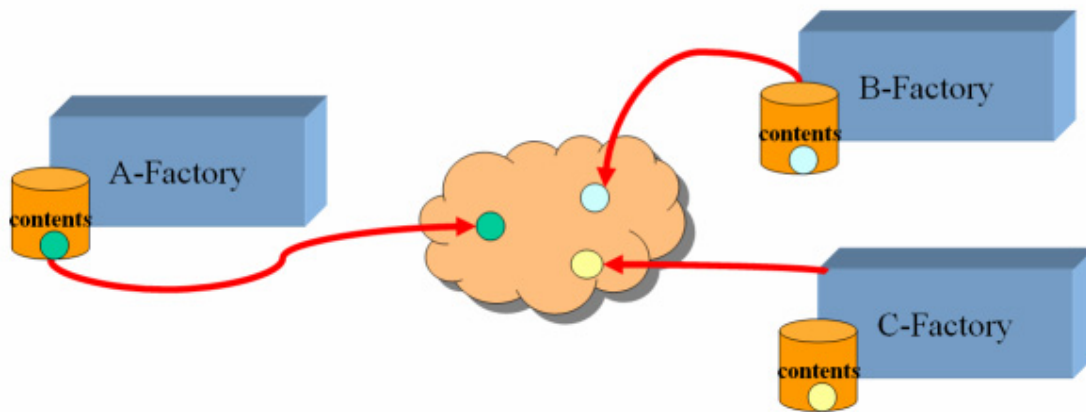
The AXEPTool is a P2P application for content sharing and distribution among content producers and distributors. In the next figure is represented the General Architecture of AXMEDIS, it highlights both the production and in the distribution of AXMEDIS digital content. The role of the AXEPTool is to allow and stimulate the sharing via mechanisms of a P2P network.



6.3.1.2 Demonstration of Scenario: Sharing content in more than one factory with AXEPTool

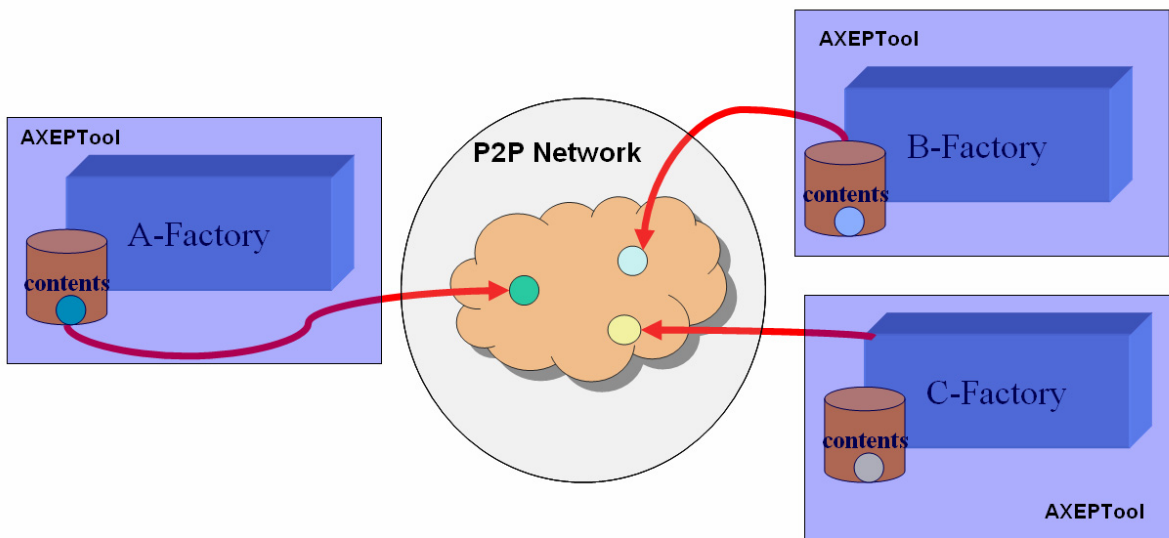
Requirement:

Participant Factories want to share contents in the distributed production environment.



Solution:

A P2P Network is under development and a tool to interact with it is provided: the AXEPTool for content sharing on a B2B environment



The P2P functionalities implemented by AXEPTool allow to:

- connect to the P2P Network: the factory joins the network and become a participant of the B2B process
- trace connections and all connected peers: mainly implemented for network “maintenance”, allows to show all connected participants in the network
- publish a content: allows a factory to make a content public and available to other factories connected to P2P
- query the network for published contents: query facilities for published objects searching task
- download a published content: using query results the factory can download the selected object from the network
- monitor downloads status: allows to monitor the progress and the status of scheduled downloads

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- monitor the network for updates of downloaded objects: For each completed download, the tool will be notified if the same downloaded object is updated and a new version is available in the network

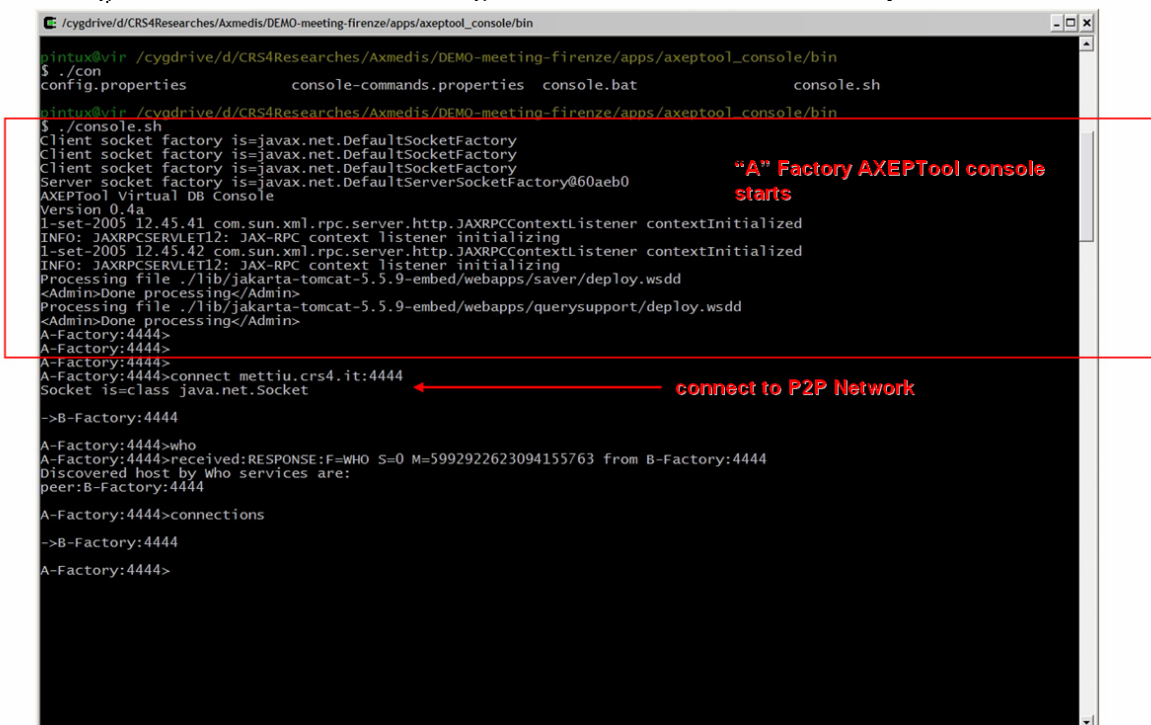
It is important to note that the AXEPTool is not an end-user application. It is mainly a daemon and a library which will be linked and contacted to produce new applications like:

- AXMEDIA, a graphical front-end aimed to B2C and C2C networks
- Content Distribution via Internet, an server side application in which the AXEPTool is linked as a library
- Workflow and content production, the AXEPTool is commanded via JavaScript by the content production engine in order to publish and download objects from the network.
- Query Support, the AXEPTool acts as a Web Service providing query results from other peers.

6.3.1.3 Demonstration of Tools

The **AXEPTool Console** provides a command line based user interface to start a node, connect to the network, sending and processing queries and downloading remote objects. The console is intended as a basic UI to manage a node but, in future, user friendly graphical user interfaces will be developed to access all the functionalities.

Connecting to the P2P network and tracing connections and other connected peers



```
cygdrive/d/CRS4Researches/Axmedis/DEMO-meeting-firenze/apps/axeptool_console/bin
pintux@vir /cygdrive/d/CRS4Researches/Axmedis/DEMO-meeting-firenze/apps/axeptool_console/bin
$ ./con
config.properties          console-commands.properties  console.bat          console.sh
pintux@vir /cygdrive/d/CRS4Researches/Axmedis/DEMO-meeting-firenze/apps/axeptool_console/bin
$ ./console.sh
Client socket factory is=javax.net.DefaultSocketFactory
Client socket factory is=javax.net.DefaultSocketFactory
Client socket factory is=javax.net.DefaultSocketFactory
Server socket factory is=javax.net.DefaultServerSocketFactory@60aeb0
AXEPTool Virtual DB Console
Version 0.4a
l-set-2005 12.45.41 com.sun.xml.rpc.server.http.JAXRPCContextListener contextInitialized
INFO: JAXRPCServlet12: JAX-RPC context listener initializing
l-set-2005 12.45.42 com.sun.xml.rpc.server.http.JAXRPCContextListener contextInitialized
INFO: JAXRPCServlet12: JAX-RPC context listener initializing
Processing file ./lib/jakarta-tomcat-5.5.9-embed/webapps/saver/deploy.wsdd
<Admin>Done processing</Admin>
Processing file ./lib/jakarta-tomcat-5.5.9-embed/webapps/querysupport/deploy.wsdd
<Admin>Done processing</Admin>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>connect mettui.crs4.it:4444
Socket is=class java.net.Socket
->B-Factory:4444
A-Factory:4444>who
A-Factory:4444>received:RESPONSE:F=WHO S=0 M=5992922623094155763 from B-Factory:4444
Discovered host by Who services are:
peer:B-Factory:4444
A-Factory:4444>connections
->B-Factory:4444
A-Factory:4444>
```

"A" Factory AXEPTool console starts

connect to P2P Network

AXMEDIS Content publication

A Factory named "B" has made a connection to the network , B-Factory performs a publication of an AXMEDIS object named objA.axm.

```

B-Factory:4444>
B-Factory:4444>
B-Factory:4444>
B-Factory:4444>publish file:///AXMEDIS/Axmedis-demo/files/objA.axm ← Object publication
publishing file:file:///AXMEDIS/Axmedis-demo/files/objA.axm
file:AXMEDIS/Axmedis-demo/files/objA.axm committed to AX8OUT
file \\AXMEDIS\\Axmedis-demo\\files\\objA.axm copied to .\\lib\\jakarta-tomcat-5.5.9-embed\\webapps\\ROOT\\objA.axm
B-Factory:4444>
B-Factory:4444>

```

After publication, the object is available in the P2P network to other factories, so a query can match it and can be included in results

Querying the network:

“A” Factory queries the network for objects of any type;
Query results are received from responding peers

```

C:/cygdrive/d/CRS4Researches/Axmedis/DEMO-meeting-firenze/apps/axeptool_console/bin
A-Factory:4444>who
A-Factory:4444>received:RESPONSE:F=WHO S=0 M=4560226011319738854 from B-Factory:4444
Discovered host by Who services are:
peer:B-Factory:4444

A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>query query.xml ← Querying for contents
file containing the query is:query.xml
query is:<?xml version="1.0" encoding="UTF-8"?>
<query xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="QUERY-v1-6.xsd">
  <source>
    <location>AXEPTOOL</location>
  </source>
  <AXinfoQuery>
    <querycondition>
      <nesting>
        <test>
          <field>MEDIA</field>
          <operator>EQ</operator>
          <value>VIDEO</value>
        </test>
        <or/>
        <test>
          <field>MEDIA</field>
          <operator>EQ</operator>
          <value>AUDIO</value>
        </test>
        <or/>
        <test>
          <field>MEDIA</field>
          <operator>EQ</operator>
          <value>TEXT</value>
        </test>
      </nesting>
    </querycondition>
  </AXinfoQuery>
</query>
A-Factory:4444>received:RESPONSE:F=UNKNOWN S=0 M=7579121614270882479 from B-Factory:4444
sysout -- received query results:<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>

```

Contents Downloading and downloads monitoring:

“A” Factory, using query results, performs an object download and monitors its status


```

<field>MEDIA</field>
<operator>EQ</operator>
<value>AUDIO</value>
</test>
<or/>
<test>
  <field>MEDIA</field>
  <operator>EQ</operator>
  <value>TEXT</value>
</test>
</nesting>
</querycondition>
</AXinfoQuery>
</query>
A-Factory:4444>received:RESPONSE:F=UNKNOWN S=0 M=378975775166827362 from B-Factory:4444
sysout -- received query results:<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>get http://mettiu.crs4.it:8080/objA.axm
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 13% 14488109/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 32% 35158189/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 45% 49490809/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 60% 65632189/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 77% 84013109/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 96% 104666649/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 100% 108006934/108006934 COMPLETE

```

request for an object download

Monitoring of downloads status

Monitor the network for updates of downloaded objects

“B” Factory performs a new publication of an updated version of the AXMEDIS object named “objA.axm”, already downloaded by “A” Factory.

Result: “A” Factory is notified about the update action on “objA.axm”

```

<value>AUDIO</value>
</test>
<or/>
<test>
  <field>MEDIA</field>
  <operator>EQ</operator>
  <value>TEXT</value>
</test>
</nesting>
</querycondition>
</AXinfoQuery>
</query>
A-Factory:4444>received:RESPONSE:F=UNKNOWN S=0 M=378975775166827362 from B-Factory:4444
sysout -- received query results:<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
<queryresults xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="/AXQS-Result-v-1-1.xsd"><AXObject><AXEPTOOL><peers><peerID>http://mettiu.crs4.it:8080</peerID></peers><AXOID>objA.axm</AXOID></AXEPTOOL></AXObject></queryresults>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>
A-Factory:4444>get http://mettiu.crs4.it:8080/objA.axm
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 13% 14488109/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 32% 35158189/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 45% 49490809/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 60% 65632189/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 77% 84013109/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 96% 104666649/108006934 RUNNING
A-Factory:4444>monitor
Local download directory is:./downloaded
[0] http://mettiu.crs4.it:8080/objA.axm 100% 108006934/108006934 COMPLETE
A-Factory:4444>new item on channel B-Factory:4444 : objA.axm, version=8

```

Update occurred, notification performed

6.3.2 Programme and Publication (UNIVLEEDS)

The demonstration of the P&P Area involves two sections, the first demonstration is of the P&P Editor to create, edit save and activate the P&P Programmes; and the second is of the P&P Engine to execute P&P programmes and perform “On-demand” processing.

The demonstration of the Programme and Publication scenarios were derived from the Use Cases in DE2.1.1b.

6.3.2.1 Introduction

Programme and Publication utilises AXMEDIS tools and functionalities to deliver correctly sized and formatted objects for each distribution channel. P&P Programmes define when and where an object or list of objects are to be delivered and can include when to start and terminate the actual P&P programme.

6.3.2.2 Contextualisation of P&P Area

The P&P Area sits between the AXMEDIS Factory and the Distributors (as shown and highlighted in the following figure). The creation of P&P Programmes to specify the distribution of Objects can be created using the P&P Editor. This editor creates a programme that when activated specifies when, how and where to deliver an AXMEDIS Object to the P&P Engine using the processing engines, workflow and the database area in the AXMEDIS factory. The same functionalities in the P&P Area can be used when requesting an Object On-Demand using an AXMEDIS Player through a distributor.

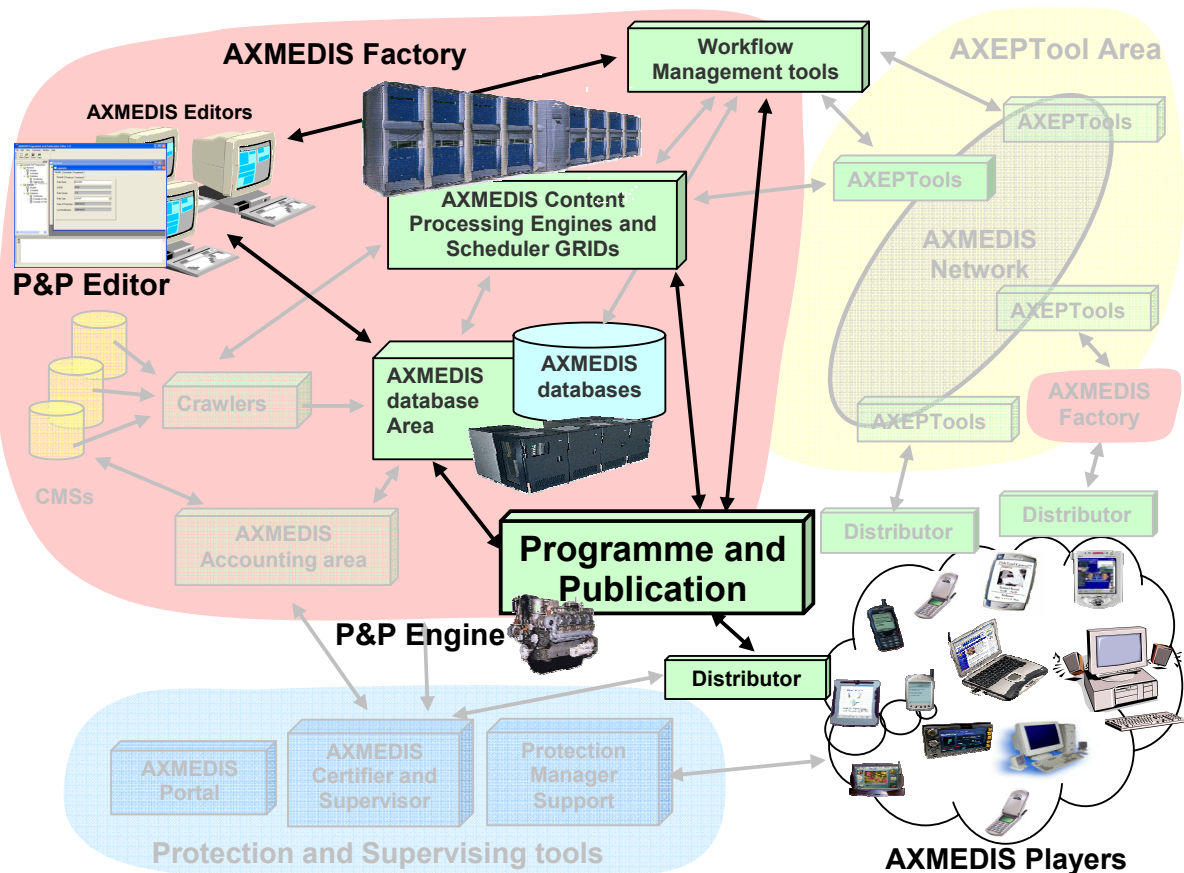


Figure:AXMEDIS Architecture highlighting the P&P Area where the P&P Engine is continuously running and the P&P Editor for creating and editing P&P Programmes

6.3.2.3 Demonstration of P&P Scenarios

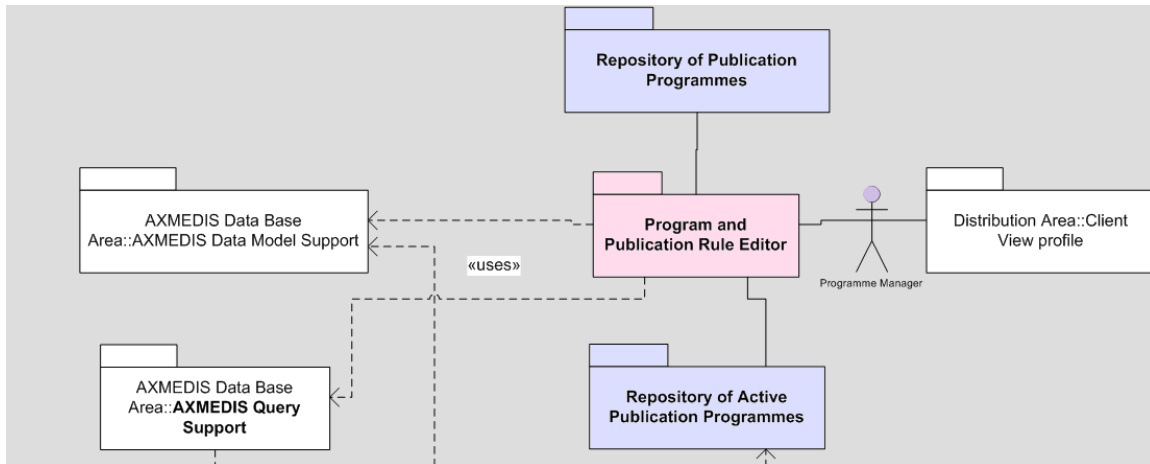


Figure: Scenario 1 – Distribution from a P&P Programme

Scenario 1a: New P&P programme

1. The actor opens the GUI (either through WF or directly) and selects a new P&P Programme
2. The programme manager uses query support to add AXObjects and/or Selections to the P&P Programme
3. The programme manager edits P&P Programme
4. The programme manager saves the P&P Programme to the P&P Repository

Scenario 1b: Loading and editing existing P&P Programme:

1. The programme manager opens the GUI and loads a P&P Programme from the P&P Programme Repository
2. The programme manager uses query support to add AXObjects and/or Selections to edit the P&P Programme
3. The programme manager saves the rule to the P&P Repository

Scenario 1c: Activating or testing an existing P&P programme:

1. The programme manager opens the GUI and loads a P&P Programme from the P&P Programme Repository
2. The programme manager activates/full trials/quick trials the P&P Programme
3. The programme is sent to the P&P Engine using the Active P&P Programme Repository

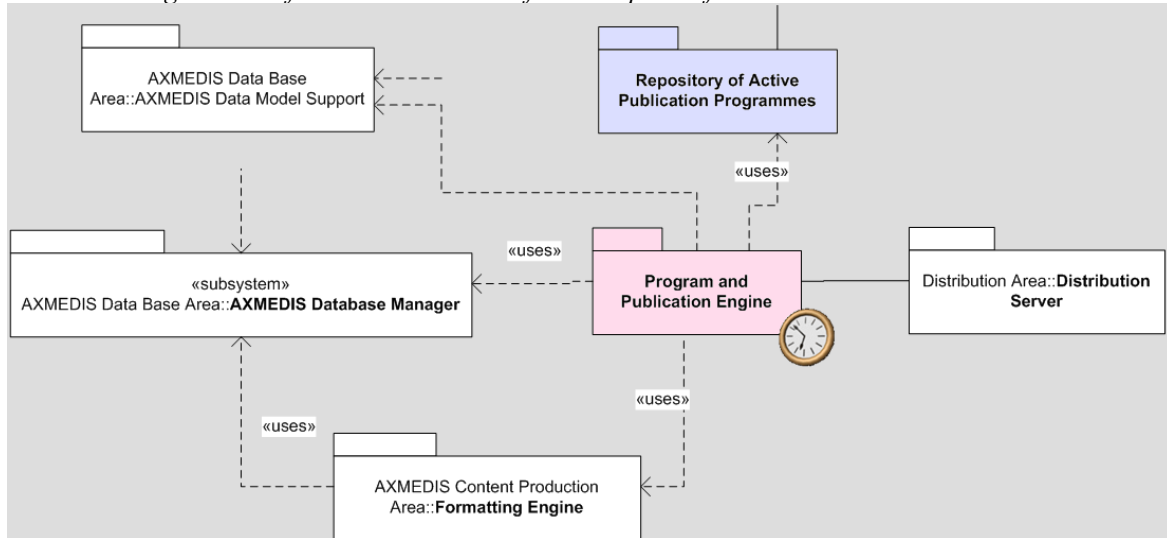


Figure: Scenario 2 – On Demand Request

Scenario 2: Requesting an Object On-Demand:

1. The AXMEDIS user selects and Object to play on the AXMEDIS player using Query support
2. The distribution area the AXMEDIS player is accessing sends a request to the P&P Engine
3. The P&P Engine checks to see if Object is valid for distribution on this channel,
 - a. If not valid, request formatting from the formatting engine and receive a new object or message stating object can not be formatted
4. Receive valid object from the database and deliver to the player using the distribution area or return message stipulating object is unavailable

6.3.2.4 Demonstration of P&P Tools

To deliver AXMEDIS objects using P&P, there are two scenarios:

- Scenario 1. In the first scenario a programme manager wants to distribute content on a particular channel (e.g. satellite). The programme manager needs to select a list of objects and specify when to deliver each object and to whom (i.e. the distribution channel).
- Scenario 2. In the second (On-Demand) scenario, a user selects a multimedia object in the AXMEDIS Distribution Area. The request from the distribution area starts a series of events behind the scenes where the user sees the object requested.

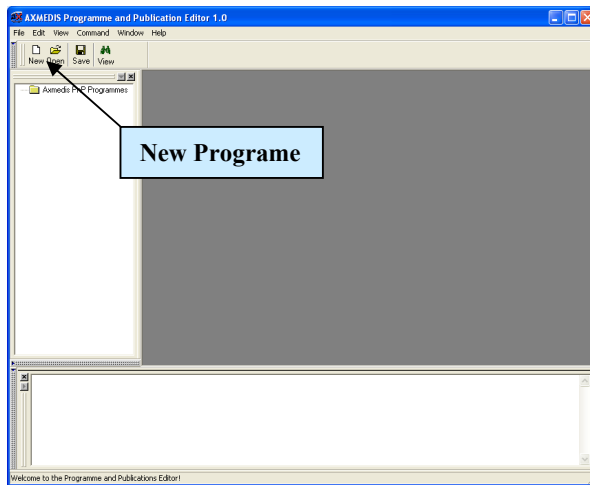
Scenario 1: Programme and Publication Programme Production

A programme producer or manager wishes to create a programme of multimedia objects for viewing on a selected distribution channel. This can be achieved by using the P&P Editor to create a distribution programme which when complete is activated.

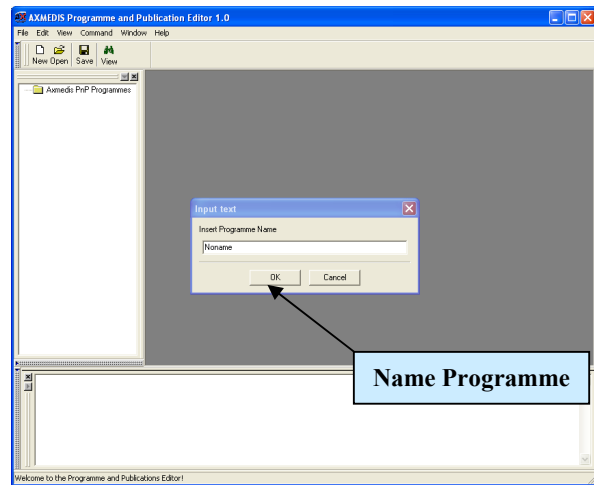
By opening the P&P Editor the programme manager can create a programme to specify the time and destination channel of the objects to be distributed. Once a programme has been created, the programme manager can save it for re-editing at a future time or test the programme. Testing the programme enables the programme manager to check if there are going to be any problems in sending each object and allow the manager within the editor to make changes to successfully distribute a programme. On completion of a complete programme, the programme manager activates the programme and during the programme life cycle the objects will be distributed until the programme has completed or the programme is removed from being activated.

Creating a programme

To create a programme, the programme manager starts the P&P Editor and from the opening screen creates a new programme by selecting “New” from the tool bar or using the File menu (File → New) or using the keyboard short cut “Ctrl-N” (see Screenshot 1). On requesting a new programme, the dialog box pops up and the programme managers enters the name of the draft programme and selects “OK” (see Screenshot 2).

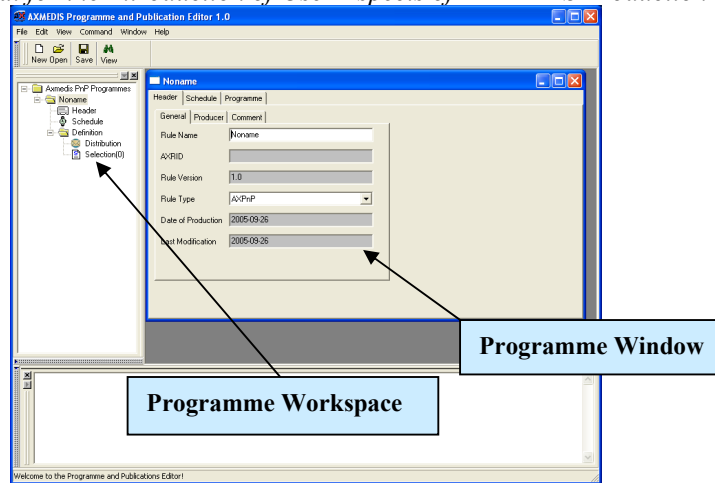


Screenshot 1



Screenshot 2

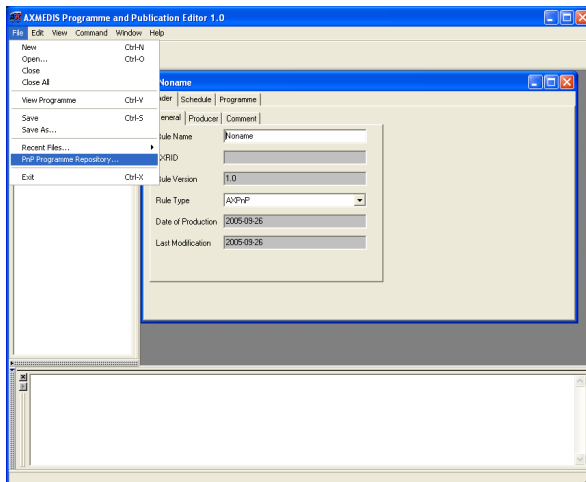
After the programme name has been OK'd, the new programme is ready for editing with the 'tree view' used as a workspace and the programme window for editing the programme details and making a programme schedule (see Screenshot 3).



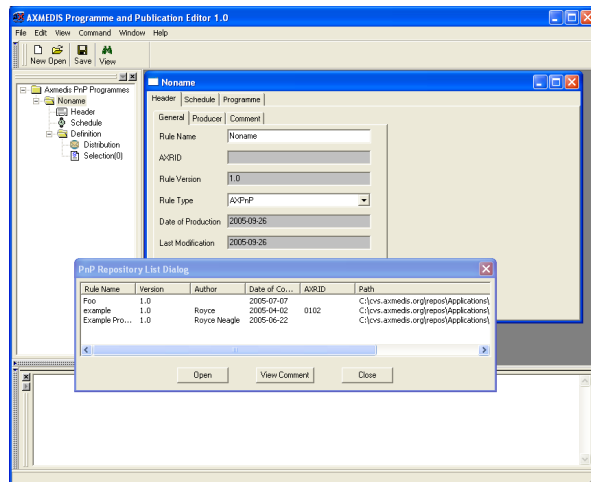
Screenshot 3

Loading and existing programme

Instead of creating a new programme, the programme manager may wish to edit an existing draft programme. The programme manager may have saved it as a file somewhere or saved it in the P&P Repository. By selecting 'open' or 'P&P Programme Repository', as seen in screenshot 4 and 5, the programme manager can load an existing draft programme from the Repository or elsewhere on the system.



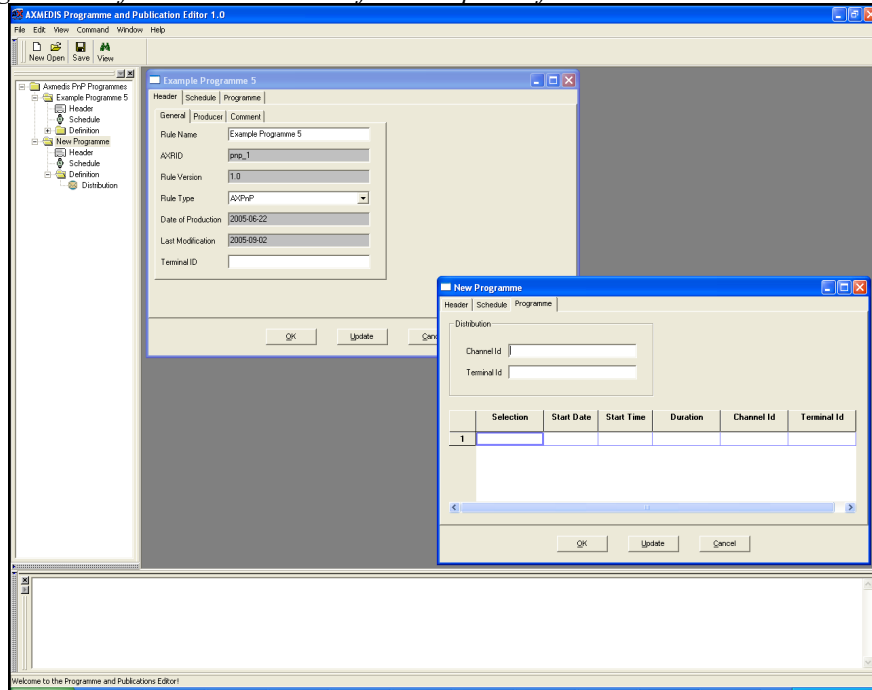
Screenshot 4



Screenshot 5

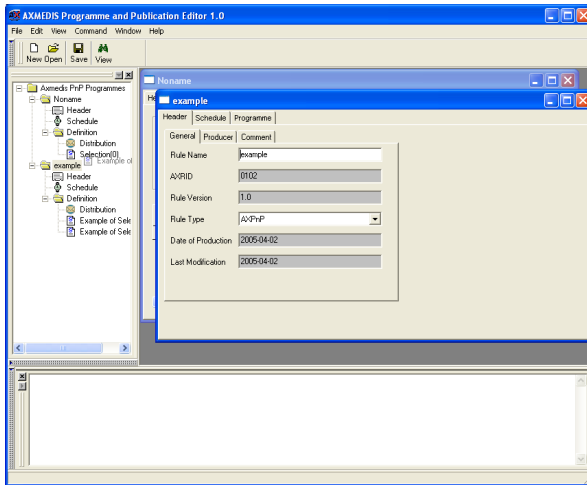
Editing a programme

The programme manager can use the editor to edit multiple programmes (as shown in screenshot 6), and edit programme data such as general, producer and content information; set the schedule for activating the programme; and set the specifications for delivering a multimedia object such as channel and terminal as well as distribution date and time, duration, channel if a different channel for a particular multimedia object is required, and terminal if a different terminal is required for a particular multimedia object.

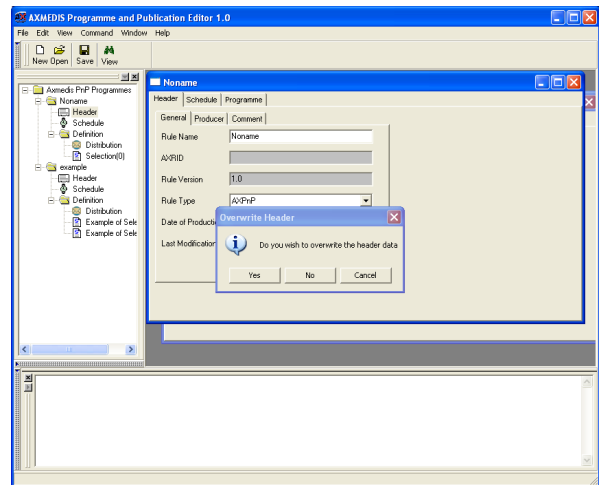


Screenshot 6

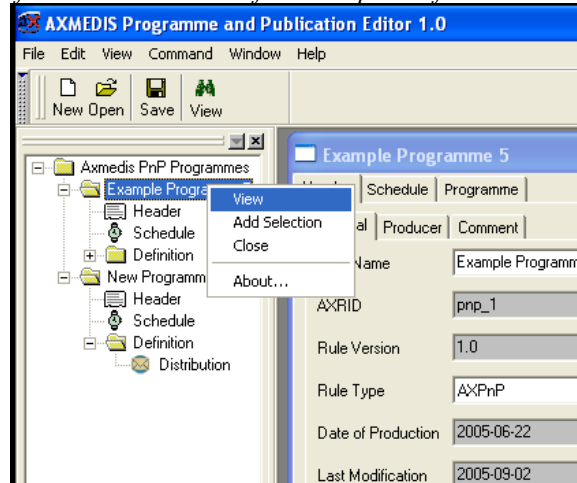
Editing functionalities include being able to drag and drop information from one programme to another using the workspace tree area as shown in the tree view in Screenshot 7 and the popup box in Screenshot 8 or using drop down menus by right clicking within the tree area as shown in Screenshot 9.



Screenshot 7



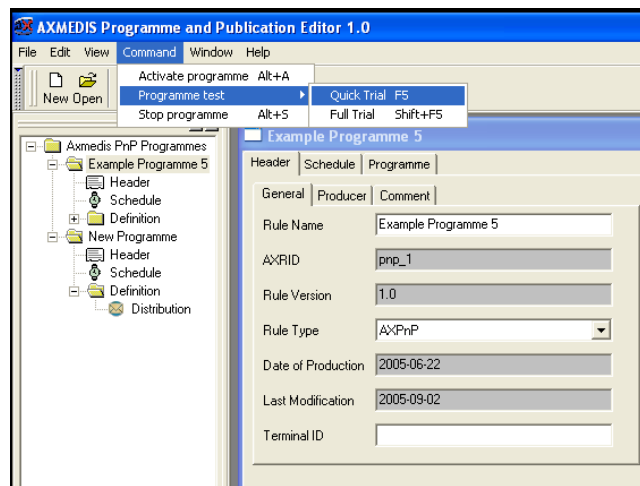
Screenshot 8



Screenshot 9

Testing a completed programme

Now the programme manager has created and edited his programme and believes it is ready for distribution, the programme manager has two options: the first is to run a quick trial; this ensures the programme devised is valid without any of the processing taking place (see Screenshot 10). This means request are made to AXMEDIS tools to make sure tasks can be accomplished for processing the programme. A message is return to the P&P Editor informing the user if the trial was successful or not. The second option is to run a full trial which enforces the P&P engine to request all the processing to be run if required with the exception of the final distribution to the distribution area. The processing jobs are requested and executed and on completion the results are returned. The final action of distribution is the only processing not requested in the full trial.



Screenshot 10

Activating and stopping a programme

When the programme manager is happy with his programme and validated it with a quick and/or full trial, the final option is to activate the programme. This sends the programme to the P&P Engine which processes the programme information and at the specified times distributes the correctly adapted multimedia object to the specified channel. During the lifespan of the active programme, the programme manager can also stop the programme from the P&P Editor by selecting “*Stop Programme*” in the Command menu. This can be seen in Screenshot 10.

Behind the scenes

On a programme activation, selected objects to be sent to a particular channel will be checked to see if the channel can use the object. If the object is fine, then nothing is done and the object is ready to be sent via the specified distribution channel. However, if there is a mismatch, the AXMEDIS tools in the AXMEDIS Factory are used to resize or format the object so that it can be distributed – providing the rights allow the object to be adapted of course. With the object (new or reformatted) that can now be sent, the P&P Engine uses an internal clock to distribute the object at the correct time, as specified in the programme by the programme manager.

Conclusion

Using the P&P Editor, the programme manager can setup a programme that sends objects through a selected distribution channel. The programme manager can create a programme and edit it to set when and where the objects are to be delivered and also test the programme to check that everything is valid for distribution. Once checked, the activated programme is sent to the P&P Engine which uses AXMEDIS tools to automatically adapt the objects to be valid for the distribution channel and at the specified time sends the correctly adapted object to the distribution servers.

Learning Outcomes

At the end of this tutorial session, the programme managers will understand:

- How the P&P Area fits within the AXMEDIS architecture
- How to use the P&P Editor to create a P&P programme to distribute objects on selected channels.
- A programme manager will be able to make a programme and edit existing programme schedules;
- Test a programme
- Activate the programme to deliver objects.
- Stop a programme that has been activated through the P&P Editor.

Scenario 2: Programme and Publication On-Demand (DE2.1.1a Section 12)

The “on-demand” production scenario provides the functionality whereby a user can enter a query to the AXMEDIS database using a Client GUI. A list of available and relevant contents is then returned to the user. From the list of available contents the user selects relevant content to purchase. If the selected content is not yet available (in the appropriate format) it can be produced (automatically processed) for and transmitted to the requesting user immediately (see figure below).

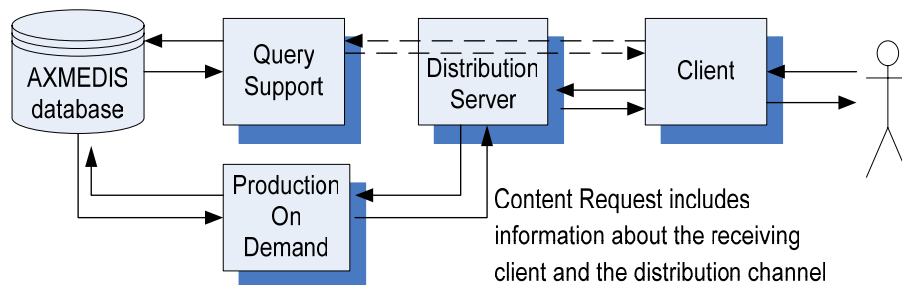


Figure: A user submits query for relevant contents. From the result list s/he selects the object(s). If the selected content is not in the appropriate format for the distribution server the content is automatically produced.

Querying the AXMEDIS Database

Imagine a user with a PDA interested to obtain a specific piece of music, e.g., a song, and the user may have a rough idea about the title and/or the year of this item. The user enters this information in a simple form on a web-page that is displayed on the PDA. Based on this information, a query message is sent to the AXMEDIS Distribution Server.

The results of the query to the AXMEDIS database are passed back to the client via the Query Support and the Distribution Server. The results are then presented to the user.

Requesting An AXMEDIS Multimedia Object

Now the user wishes to access one of the listed multimedia objects on-demand for viewing on a selected distribution channel. The user may browse through the list of available contents and select one or more specific objects according to the interest. This selection is then passed on to the Distribution Server.

The distributor connects to the P&P engine, requests a multimedia object and states the distribution channel, e.g., PDA. This allows the P&P engine to determine whether the object is available in the correct format for the target.

If the object is already available in the correct format it is sent straight back to the distributor. If the object requires some reformatting the P&P engine deals with this and then sends a new correctly formatted object to the distributor.

Behind the scenes

The P&P engine looks up the requested multimedia object and determines whether it is appropriately formatted for that distribution target.

If no reformatting is required, the engine requests the object from the AXMEDIS database and sends it on to the distributor. If the object is not in the correct format for the distribution target, the engine requests a new object in the correct format. Once this object has been created, the P&P engine is notified by being sent the object identifier for the newly formatted multimedia object. This object can now be retrieved from the AXMEDIS database and delivered to the distributor.

What happens if the object is unavailable?

If the requested object is unavailable or cannot be re-processed into a suitable format, a fail message is sent back to the distributor.

Conclusion

The P&P Engine handles all on-demand requests for AXMEDIS objects from the distributors. Requests are sent to the engine along with a target channel and the P&P engine does any work necessary to get a correctly formatted object returned to the distributor.

Learning Outcomes

At the end of this tutorial session, the distributors will understand:

- How the P&P Area fits within the AXMEDIS architecture
- How AXMEDIS objects can be requested by distributors.
- How the P&P Engine can reformat AXMEDIS objects into the appropriate format for the selected channel
- How the P&P interacts with Query on Demand to allow users to retrieve and view selected AXMEDIS objects selected from a query processed by Query on Demand
- How AXMEDIS integrated different distribution channels into a common framework for Querying and Production on Demand

6.3.3 Query on Demand (FHGIGD)

The “On-Demand” module allows interactive queries of content by consumers. If the customer is interested in a content that is not yet available in a compatible format, for the particular channel, the content can be automatically generated and delivered for immediate consumption in the appropriate format for a specific distribution channel and client device.

The overall scenario can be seen in the figure above (in the last section).

Introduction

The web interface of the demonstrator provides the possibility to compose a query, which is send to the distribution server. The server evaluates this query and returns a list of matching content, which is shown to the user by the web interface.

Composing a Query

After the user has entered his account information the web interface forwards to the query page. This page allows to compose a query, which is finally send to the server by clicking the submit button.

A query can contain one or more terms which are combined using an 'AND' or 'OR' conjunction. Each term consists of four input fields: Field, Negation, Operator and Value.

- Field represents a specific database field.
- Negation negates the following operator.
- Operator represents the operation which is applied.
- Value represents the value against which the database field has to match.

The buttons on the left are used to add or remove terms.

Figure: Web Interface for Querying of AXMEDIS Database

Based on this information, a query message in XML is created and sent to the AXMEDIS Distribution Server. This query is transcoded and sent to the web service interface of the AXMEDIS database. It is worth noting that Queries are not limited to local databases. The framework allows query to be forwarded and submitted to all the content available and connected to the AXMEDIS P2P (peer-to-peer) network.

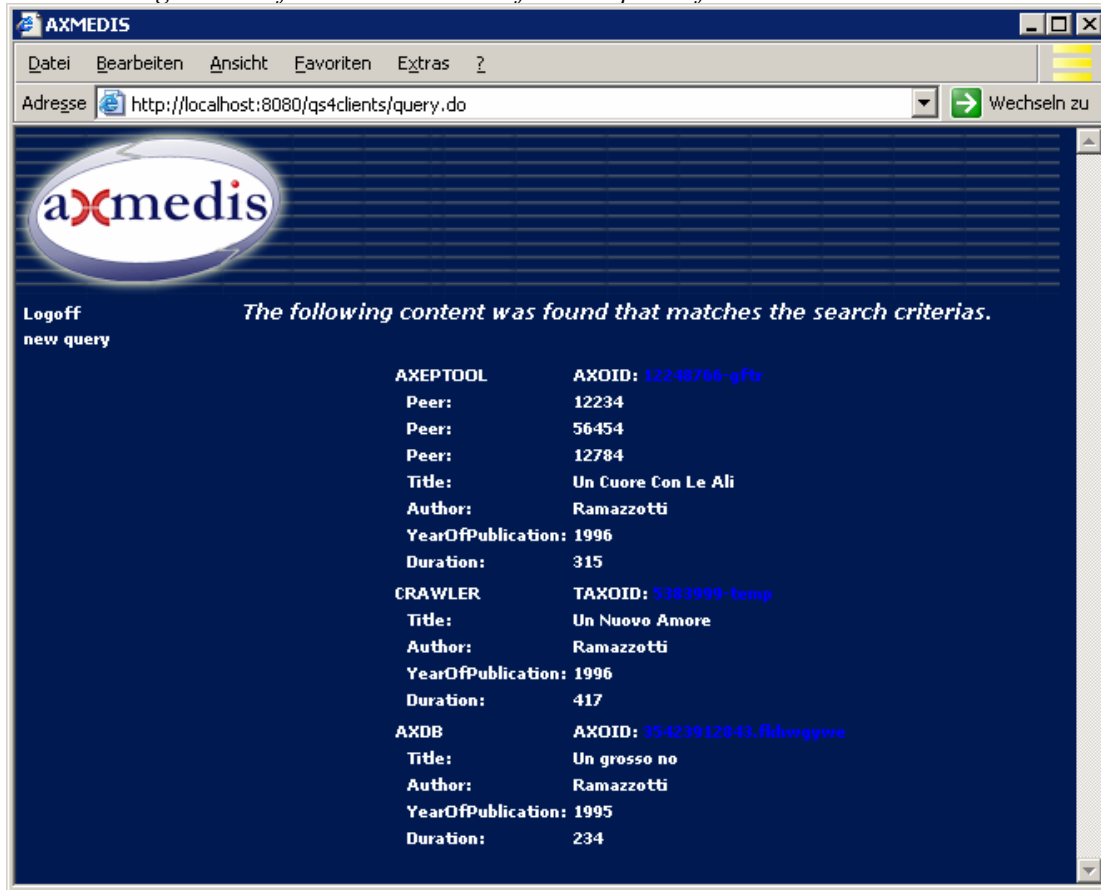


Figure: Presentation of results found in AXMEDIS Database

The results of the query to the AXMEDIS database are passed back to the client via the Query Support and the Distribution Server. The results are then presented to the user on a web-page, and the user can now browse through the list of available contents and select one or more specific objects according to the interest. This selection is then passed on to the Distribution Server.

Behind the scenes

The demonstrator web interface provides the possibility to query the AXDB-Server for information about the available multi-media content. The AXDB server provides a powerful query language that is too complex to be handled by an end user in this case. For this purpose the AXMEDIS specification defines the simplified query language schema. The user/web interface therefore creates a request based on this language schema, which is processed by the Query Support for Distribution Channels module as described in the next section.

The query message includes a Client-Profile which specifies the capabilities of the client device, e.g. screen size or CPU (e.g., Composite Capabilities Preference Profiles). At the Distribution Server a Distribution-Profile is added and the query is passed on to the Query Support (for Distribution Channels) within the AXMEDIS framework. The query is verified and a query to the AXMEDIS database for the requested content is created that takes the Client- and Distribution-Profiles into account. Further parameters are added and adjusted according to the Client- and the Distribution-Profiles.

After the server has evaluated the user request and returned the results, the demo client presents a list of matching content to the user. The user then chooses the content she/he wants to see or goes back to enter another query. If the user requests content, this request is delegated to the Programme and Publication Tools module, which prepares the content, based on the determined AXOID.

Conclusion

The Query on Demand allows final users to pass queries into the AXMEDIS database. It exposes an easy to use interface for different distribution channels with a simplified query language. It also integrates with the P&P tool to produce a correctly formatted object for a specific client and distribution channel.

Learning Outcomes

At the end of this tutorial session, the distributors will understand:

- How the Query on Demand fits within the AXMEDIS architecture
- Querying for AXMEDIS objects
- How the P&P interacts with Query on Demand to allow users to retrieve and view selected AXMEDIS objects using the Query on Demand web interface
- How the queried objects are automatically reformatted by the P&P Engine into the appropriate format for the selected client device and distribution channel

6.4 Content Protection Tools

6.4.1 DRM Tools (FUPF)

6.4.1.1 Edition and visualisation of DRM information

The demonstration of the edition and visualisation of DRM information is derived from the tools implemented in WP4.5 and the Use Cases defined in DE2.1.1b.

Introduction

In order to provide DRM capabilities inside the AXMEDIS system, it is needed to create DRM information. To describe DRM information governing AXMEDIS objects, it has been decided to use the rights expression language defined by MPEG-21 standardisation group, MPEG-21 REL. This is not the only rights expression language considered inside AXMEDIS, as we will also work with OMA DRM REL, a rights expression language proposed by Open Mobile Alliance (OMA), derived from ODRL (Open Digital Rights Language).

At the moment, tools for edition and visualisation of DRM information only work with MPEG-21 REL. Other tools, like Rights Expression Translator, are being developed to provide interoperability of DRM systems inside AXMEDIS.

DRM information will support different business models, from the creation of content to its distribution to the final user by means of licences. These licences will be organised as a chain and, in order to authorise user actions (not only final user actions, but also distributor actions over content), all licences in the chain will be considered.

Edition and visualisation of DRM information involves several modules inside the AXMEDIS system, from the creator, distributor and final user side to the Protection and Supervision part of the system.

To edit and visualise DRM information, there are several scenarios:

Scenario 1. In the first scenario a distributor creates a final user licence based on the distribution licence he received from a content creator.

Scenario 2. In the second, a final user visualises the rights he has purchased over a digital content from a distributor.

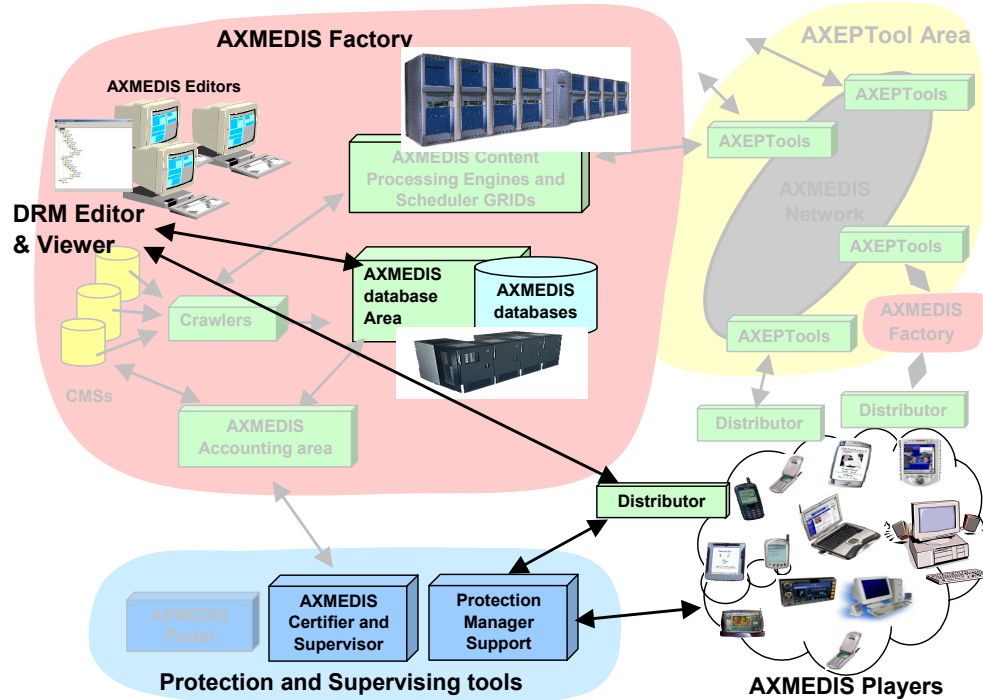


Figure: AXMEDIS Architecture highlighting the DRM Tools area

The general structure of an AXMEDIS licence is shown in the next figure:

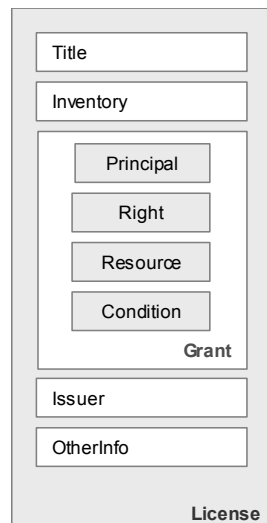


Figure: REL Licence Structure

The fields represented in the above licence are:

- Title: this element provides a descriptive phrase about the Licence that is intended for human consumption in user interfaces. Automated processors must not interpret semantically the contents of such title elements.
- Inventory: this element is used for defining variables within a Licence. In the Inventory element of a licence can be defined LicencePart elements that in turn can have licencePartId attributes that can be referenced from elsewhere in the licence. Therefore, REL provides a syntactic mechanism for reducing redundancy and verbosity in Licences that can be used throughout a Licence.

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Grant or GrantGroup: The Grants and GrantGroups contained in a licence are the means by which authorisation policies are conveyed in the REL architecture. Each Grant or GrantGroup that is an immediate child of a licence exists independently within that licence, no collective semantic (having to do with their particular ordering or otherwise) is intrinsically associated with the presence of two or more of them within a certain licence. Inside an REL licence, the most important element is the Grant. A Grant is an XML structure that is formed by four elements:
 - o Principal represents the unique identification of an entity involved in the granting or exercising of Rights.
 - o Right specifies an action or activity that a Principal may perform on, or using, some associated Resource.
 - o Resource represents the object against which the Principal of a Grant has the Right to perform.
 - o Condition represents grammatical terms, conditions and obligations that a Principal must satisfy before it may take advantage of an authorisation conveyed to it in a Grant.
- Other information: Using the wildcard construct from XML Schema, a Licence provides an extensibility hook within which licence issuers may place additional content as they find appropriate and convenient. This can be useful for conveying information that is peripherally related to, for example, authentication and authorisation, but is not part of the REL core infrastructure. It should, however, be carefully understood that not all processors of REL licences will understand the semantics intended by any particular use of this extensibility hook. Processors of the licence may choose wholly at their own discretion to completely ignore any such content that might be present therein.

Scenario 1: A distributor creates a final user licence

A distributor is able to create final user licences for the contents he/she distributes. The licence issued by this distributor has to conform to the conditions imposed on him/her by the creator or other distributor of the content.

Creating a licence

A new licence can be created in two ways, from scratch, by using the option New or by opening an existing one and modifying it, with the File menu option Open.

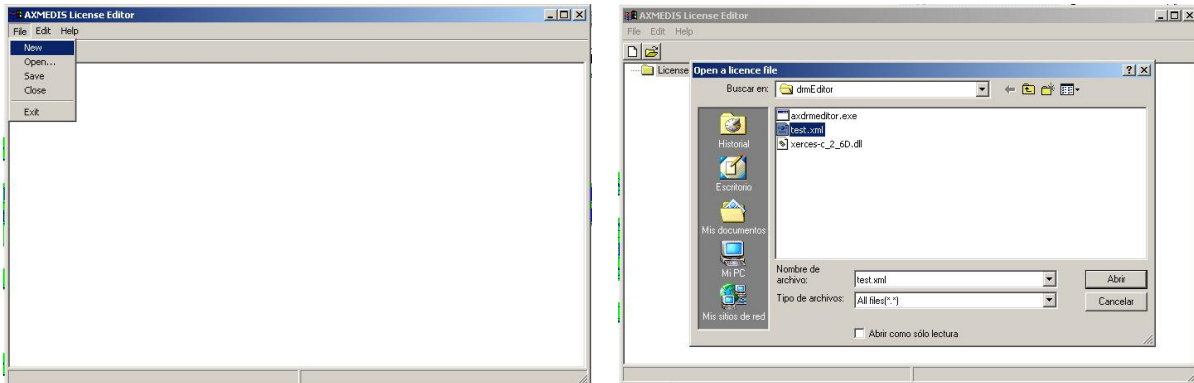


Figure: New and Open menu options on DRM Editor and Viewer

If the New option is selected, then an empty licence is created, as shown in the next figure. In the empty licence, a grantgroup and an issuer for the licence can be added.

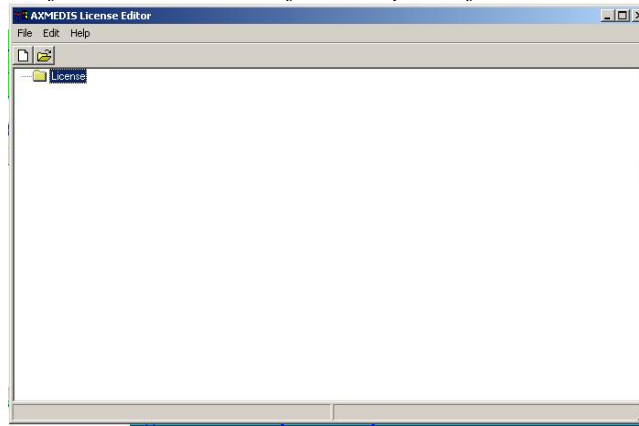


Figure: Empty licence

The issuer should be the distributor, which has to sign the licence in order to guarantee that it has not been modified and that he/she was the issuer of it.

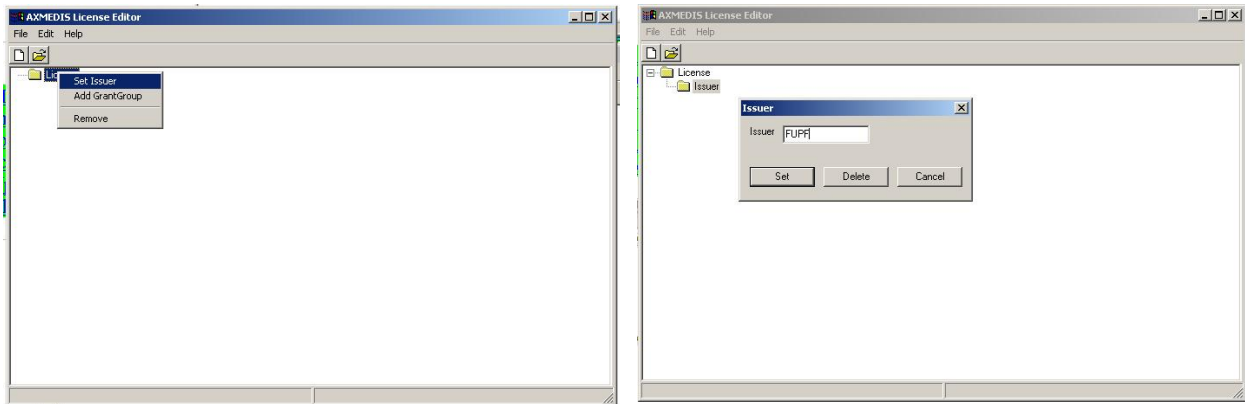


Figure: Adding issuer to the licence

In the grantgroup element, the final user who this licence is for can be defined. Also the digital content, the right or rights granted to the user and the conditions.

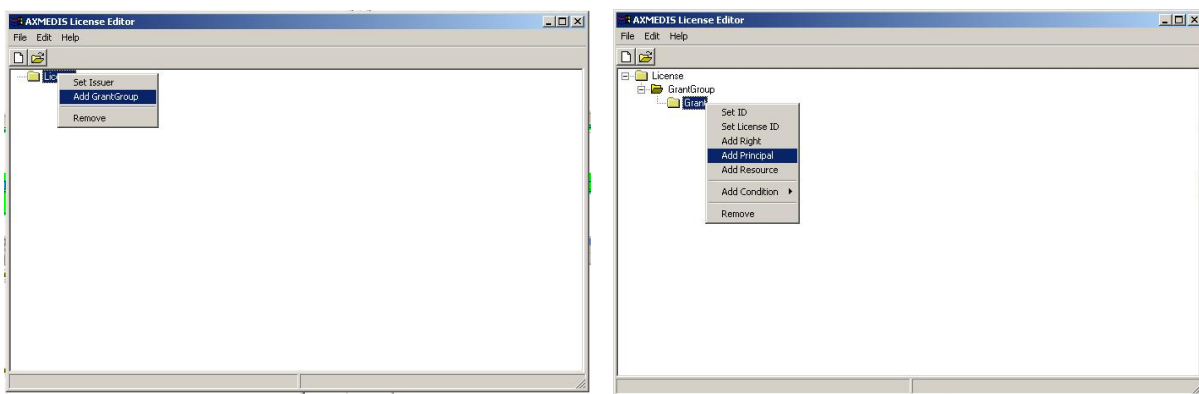


Figure: Adding GrantGroup and Principal (final user) to the licence

Next figures show how the right can be selected from a list of rights (defined in the multimedia extension of MPEG-21 REL standard) and how the resource can be added.

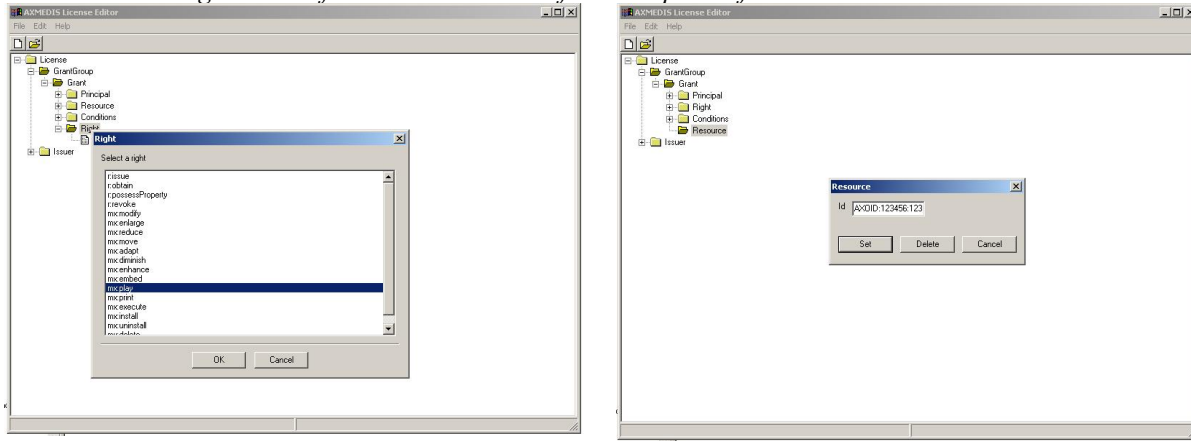


Figure: Adding right and resource

These conditions can be related to location (country where the licence has effect), how the payment has to be processed (prepay, flat fee, pay per use) or the validity interval of the licence (it can be defined the starting date and / or the end date of the validity interval).

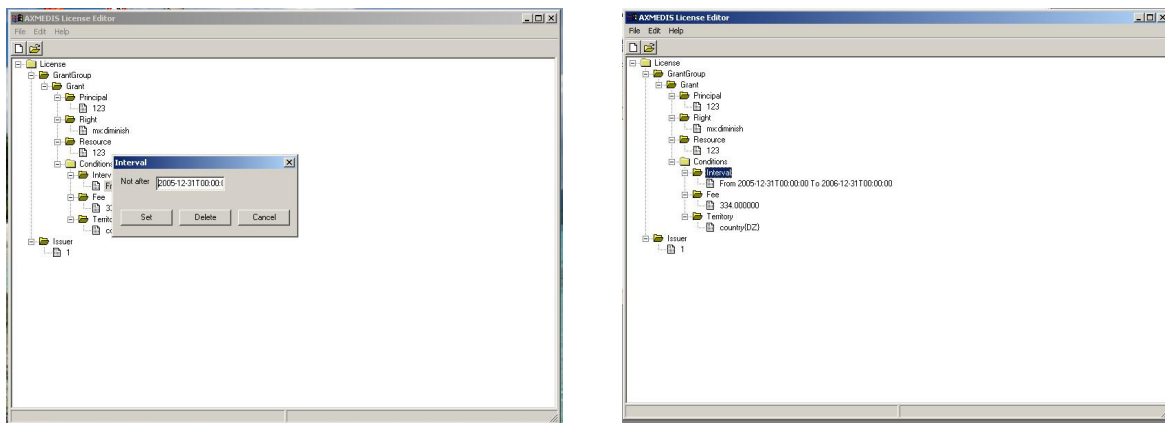


Figure: Modifying validity interval

Conclusion

Using the DRM Editor and Viewer, a distributor can create a licence for the final users. This licence will give permission to a user to exercise a right over an AXMEDIS object. Permission over several objects can be given, if the distributor has the right to do so.

Learning Outcomes

At the end of this tutorial session, the attendants will understand:

- How a distributor can create a licence for a final user
- How to add rights and conditions of usage of a digital content inside a licence
- How the licence is structured

Scenario 2: A final user visualises his licences

A final user is able to visualise the licences he/she has purchased over content. This is also applicable to distributors or content creators who want to visualise the licences they have created or they have purchased.

Opening an existing licence

A licence can be opened with the File menu option Open.

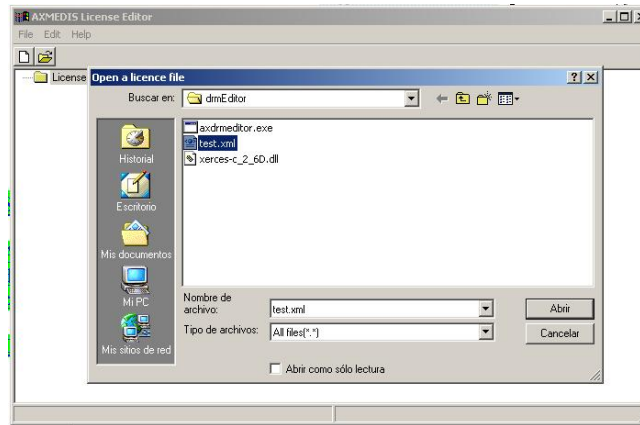


Figure: Open menu option on DRM Editor and Viewer

When the licence is opened, it is shown as a tree in the DRM Editor and Viewer. The different elements are shown as directories and the contents (leafs of the tree) are shown as files. Directories are opened and closed when the user clicks on the corresponding values. When the user opens and closes directories, contents like rights issued, conditions and receiver of the licence (Principal) are shown.

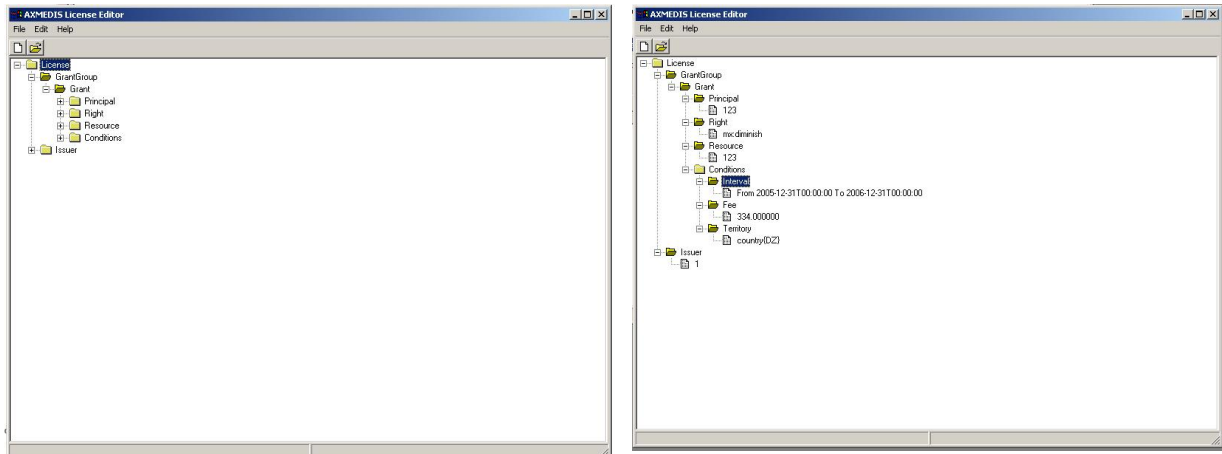


Figure: Licence visualisation with directories closed and opened

The user can perform searches over the licence conditions in order to check if he has a specific right over a digital content.

The search fields available are:

- AXOID: Identifier of the AXMEDIS Object (digital content) over which a right has been issued. This is identified as Resource in the licence tree.
- Person: This is the “receiver” of the licence. This is identified as Principal in the licence tree.
- Right: The actions a user can perform over a licence. Only the rights inside the licence are eligible to be selected.
- Condition: These are the usage conditions of the digital content. There are three kinds of conditions: date condition that is checked against the interval defined in the licence, territory condition and fee condition.

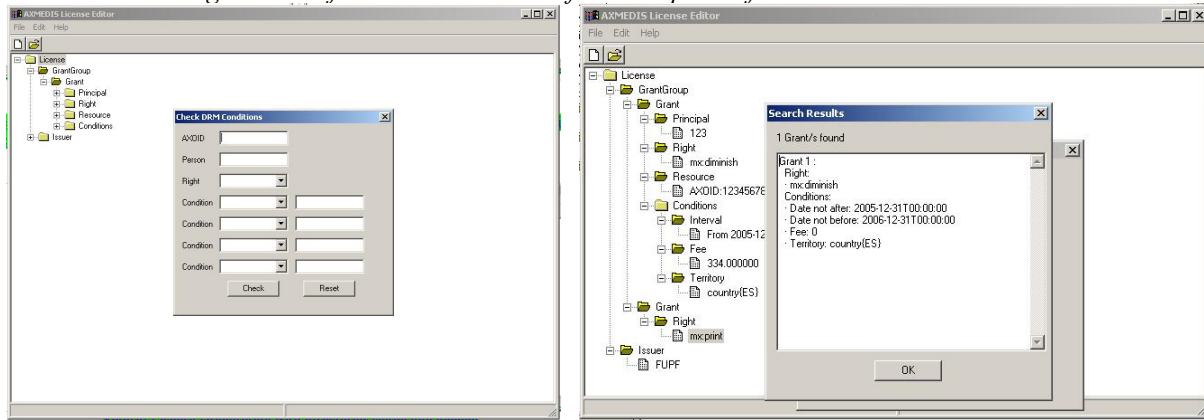


Figure: Searching grants inside the licence

The results of the search are shown as a list, where the accomplishing grants are shown.

Conclusion

Using the visualisation functionalities of DRM Editor and Viewer, a user can visualise a licence he/she has purchased, performing basic searches over it. For the distributors, this searching functionality could be useful to check the rights and conditions of the newly created licences.

Learning Outcomes

At the end of this tutorial session, the attendants will understand:

- How a licence can be opened
- How it is visualised and the elements it may contain
- How to perform searches in order to check rights and usage conditions

6.4.2 Content consumption scenario

Apart from creating licences for distribution and for final users, it would be very useful to show the relationship between the different modules providing protection in the system.

To demonstrate content consumption, there are several scenarios:

Scenario 1. In the first scenario a final user consumes a content governed by a licence he/she owns.

Scenario 2. In the second, a final user wants to consume a content, but he/she needs to purchase a licence first.

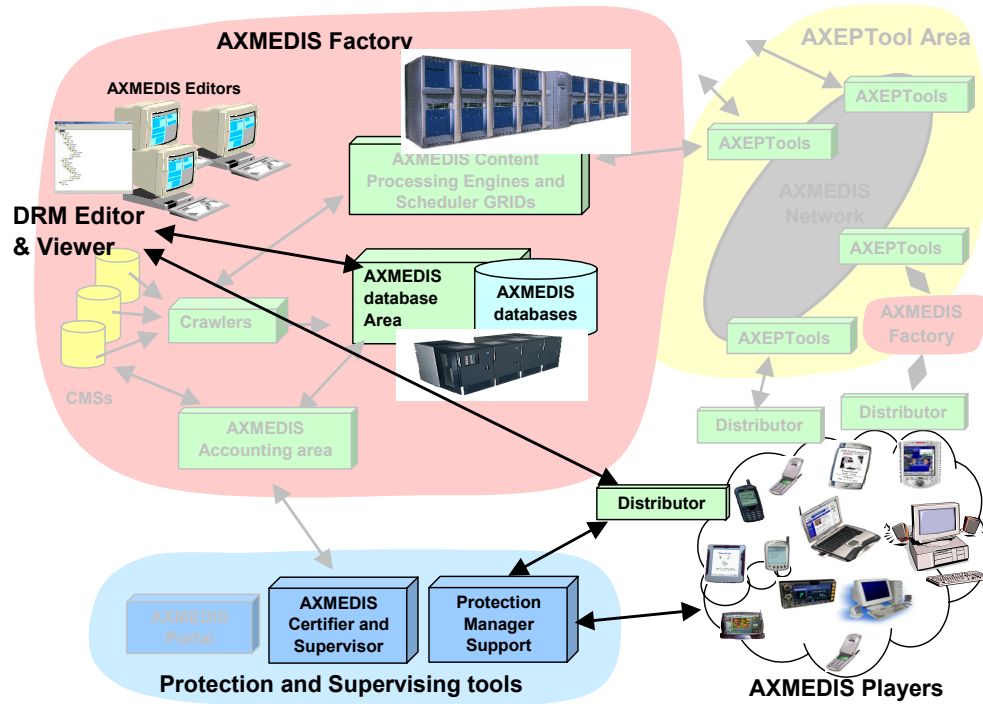


Figure: AXMEDIS Architecture highlighting the DRM Tools area

Scenario 1. Successful content consumption

The following figure shows the scenario on which the final user wants to consume a protected and AXMEDIS governed object.

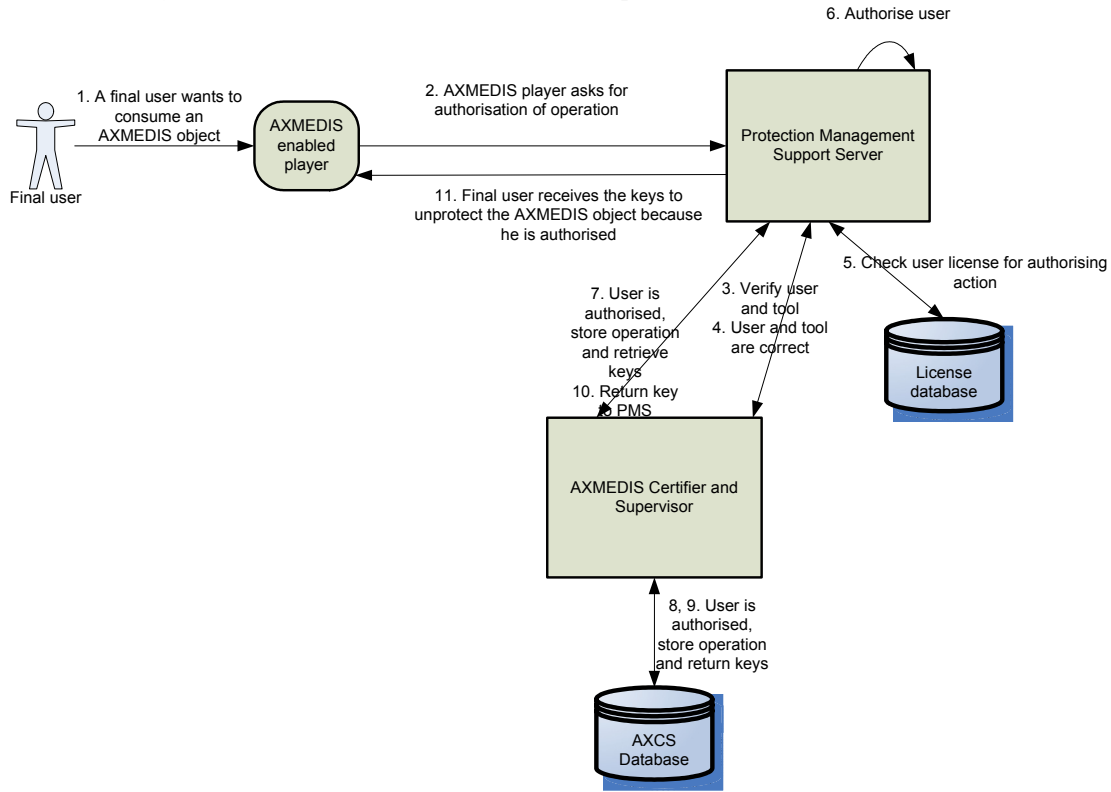


Figure: Final user successful content consumption scenario

1. A final user wants to consume a protected and governed AXMEDIS object. In order to do so, he/she needs to be authorised.
2. An AXMEDIS enabled player asks for authorisation of operation to Protection Management Support (PMS) Server module. This module is in charge of authorising user actions.
3. Before the authorisation can be done, the user and the tool used by the user are verified through the AXMEDIS Certifier and Supervisor (AXCS). If the verification fails, then the user will not be able to consume the object and he is usually blocked.
4. The user and the tool are verified and an OK response is returned to the PMS Server.
5. The PMS Server looks for the corresponding licences in the database in order to authorise user action. To do so, the MPEG-21 REL authorisation algorithm is used.
6. The authorisation algorithm is executed and the user is authorised to consume the content.
7. The AXCS modules registers that the user has been authorised.
8. The user action is stored in the database.
9. The keys for unprotecting the AXMEDIS object are retrieved from the AXCS database.
10. The keys are returned to the PMS Server.
11. Final user action is authorised and the keys for unprotecting the object are sent to the user

Scenario 2. Purchase of licence for content consumption

The following figure shows the scenario, on which the final user wants to consume a protected and AXMEDIS governed object, but he/she has not obtained the licence and he/she has to purchase it.

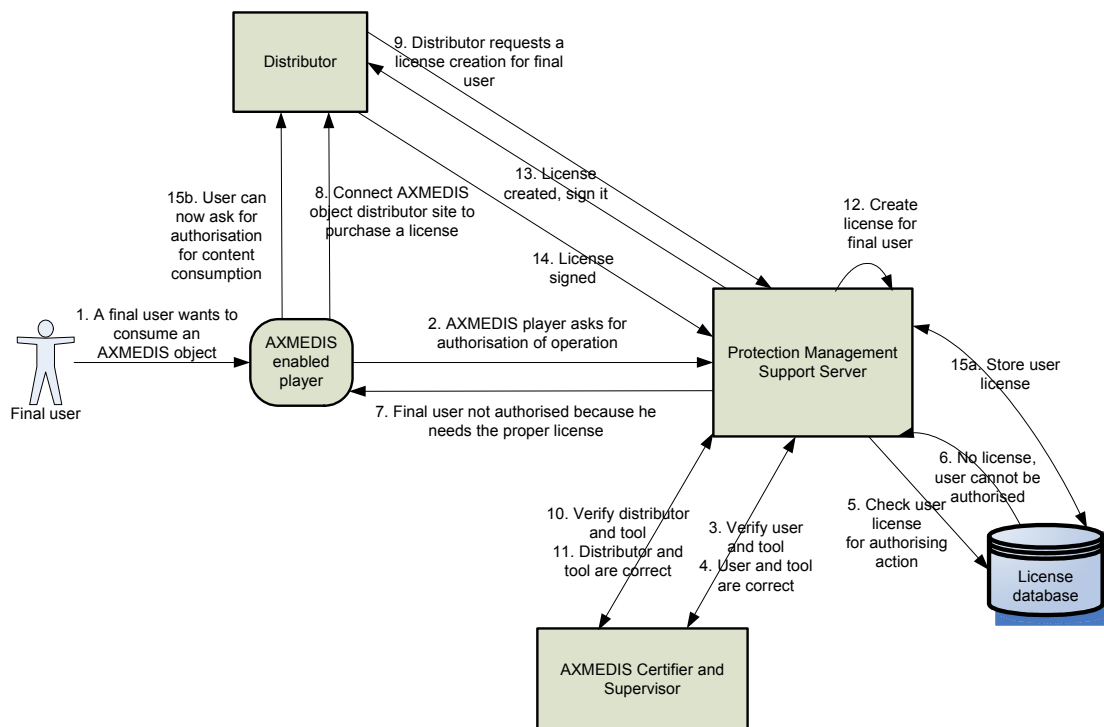


Figure: Final user licence purchase scenario

- A final user wants to consume a protected and governed AXMEDIS object. In order to do so, he needs to be authorised.
- An AXMEDIS enabled player asks for authorisation of operation to Protection Management Support (PMS) Server module. This module is in charge of authorising user actions.
- Before the authorisation can be done, the user and the tool used by the user are verified through the AXMEDIS Certifier and Supervisor (AXCS). If the verification fails, then the user will not be able to consume the object and he/she is usually blocked.
- If the user and the tool are verified and an OK response is returned to the PMS Server.
- The PMS Server looks for the corresponding licences in the database in order to authorise user action. To do so, the MPEG-21 REL authorisation algorithm is used.
- No licence can be found for this user and content, a negative response is returned.
- If a negative response is given to the user, he/she is informed that he/she needs a licence to consume the AXMEDIS object.
- Final user has to connect to the AXMEDIS object distributor's site in order to purchase the appropriate licence to consume the content. The reference to this site could be inside the AXMEDIS object.
- Distributor requests final user licence creation to the PMS Server. It has to provide the appropriate parameters to create the licence.
- Before the licence can be created, the distributor and the tool used by the user are verified through the AXMEDIS Certifier and Supervisor (AXCS). If the verification fails, then the distributor will not be able to request the licence creation and he/she is usually blocked.
- If the distributor and the tool are verified and an OK response is returned to the PMS Server.
- Licence for final user is created on the basis of the parameters received from the distributor.

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- The licence is sent to the distributor in order for him/her to sign it for further authentication of the licence.
- The signed licence is returned in the PMS Server, which stores it in its database.
- a The licence is stored in the database.
- b The final user is informed that he/she can consume now the AXMEDIS object. Then, the Scenario 1, Successful content consumption applies.

There are several variations to these scenarios, considering when the final user purchases the AXMEDIS object and associated licences, but the main modules and steps have been described in these two scenarios.

6.4.3 Legal Aspects (AFI)

The law is relevant to the DRM process at all stages in the life cycle of a piece of digital content. Copyright is the main category of law relevant to DRM systems as copyright arises automatically when original material is created or produced and confers certain rights on the owner of the material. This section is intended to show AXMEDIS impact with respect to IPR and copyright management and to provide an overview of the current legal framework and issues related to DRM. It will aim to clarify the main changes in the rights management framework as well as the benefit of the reporting and monitoring mechanism for the rights collection and sharing. AFI will also contribute to the training activities related to the interaction between contracts practices and DRM.

6.4.3.1 Demonstration of Scenarios

Demonstration will lead to present the following scenario

- Scenario on how AXMEDIS solutions evolve to meet the content industry requirements and to support revenue collection and sharing. This will be illustrated by showing AXMEDIS impact with respect to
 - The Individual management and collective management of the rights
 - The multiplicity of rights holders
 - Different nature of the rights involved in a single content and in a compound content
 - The royalties collection and distribution
 - The interaction with the collecting societies

Other training activities will illustrate the EU legal framework for DRM and in particular the EUCD main aspects and outstanding issues including the implication of the differences between the Member States' laws that govern DRM

- Exceptions
- Private copy
- TPM infringement

In the course of the project, AFI will collaborate with DSI and FUPF in the development of training material aiming at illustrating main aspects, issues and differences between business contracts and DRM licensing models and at explaining how contracts elements will interact with AXMEDIS licensing models.

Conclusion

This module deals with the legal and IPR aspects related to the exploitation of AXMEDIS object: particular attention will be given to the aspect of copyright and related rights management and to the rights usage monitoring process which will enable a more effective and transparent collection and sharing of revenues due to the rights holders. EU legal framework of DRM and current issues will be also illustrated. AFI will also support DSI and FUPF in the development of training material and activities related to the integration between contracts elements and licensing models.

Learning Outcomes

The attendants will

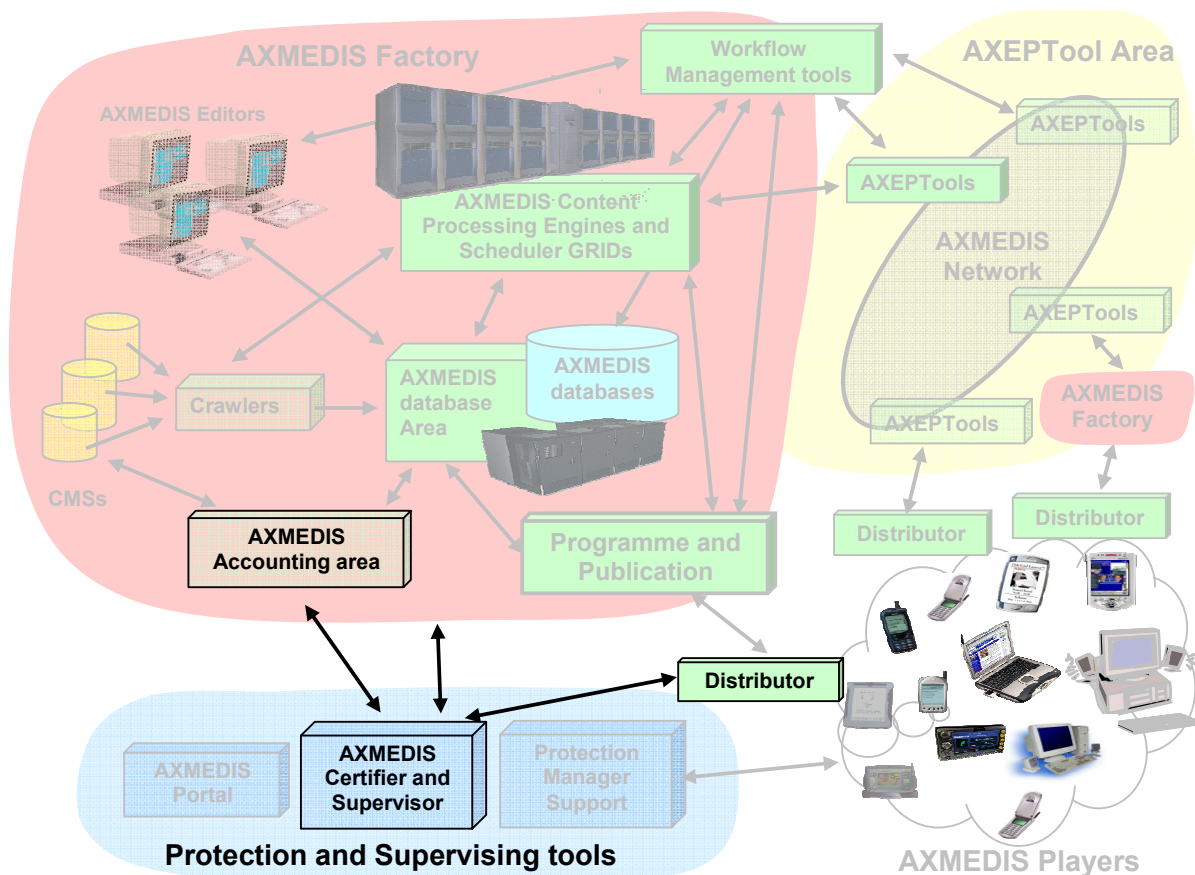
- Understand how will AXMEDIS cope with the management of copyright and related rights aspects for multi channels distribution
- Have an overview of the European legal framework for DRM and of the main issues and aspects of this evolving scenario
- Understand how AXMEDIS licensing models are changing contracts practices and help business contracts parties to identify the new elements and conditions that have to be taken into account in the negotiating process

6.4.4 Certifier and Supervisor (DSI)

Since the AXCS is a server (and not a client or a stand-alone application) it is difficult to provide a “nice” demonstration: there are no GUIs to show and no multimedia files to play. It provides several services (most organized as Web Services) to other AXMEDIS components. The demonstration provided in this seat involves three sections: the first concerns final user registration and in particular “Self registration of a final user”. The second concerns reporting of data about actions performed by users on objects for accounting purposes: it involves the AXCS Reporting web service. The third concerns reporting of data about actions performed by users on objects for statistic analysis purposes: it involves the AXCS Statistics web service.

6.4.4.1 Contextualisation of Certifier and Supervisor

AXCS is a component of the Protection and Supervising tools area, as shown in the following figure.



DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

AXCS is a server that provides services to other AXMEDIS components. It provides services to register new users in AXMEDIS (both final users and business user), to assign IDs to objects and register them, to gather data for reporting and accounting purposes.

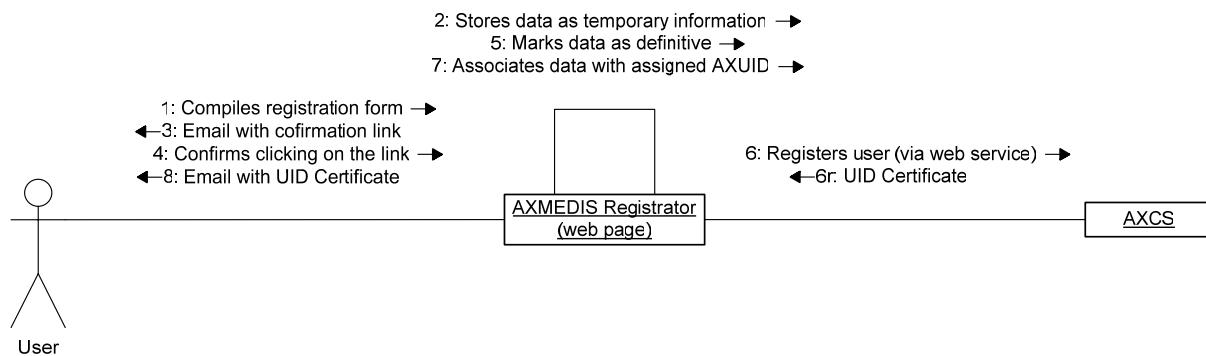
So, AXCS relates to various entity:

- distributors, as they can directly register new final users;
- PMS, as it uses AXCS to:
 - store data about created licences;
 - store data about actions performed by users;
- CAMART (a component of the accounting area) as AXCS provides to it the required data for accounting and statistic analysis purposes.

6.4.4.2 Demonstration of Certifier and Supervisor Scenarios

Final user self registration

In order to demonstrate self user registration of final users the following scenario will be enacted.

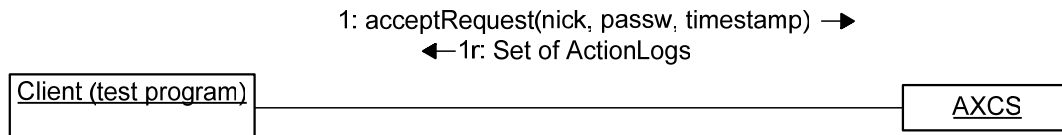


In this scenario a final user registers himself/herself to AXMEDIS through a predefined web page hosted on the AXMEDIS Registrator. Confirmation via email is used both to check the validity of the email communicated and to prevent automatically generated requests; if the confirmation does not take place in a certain eriod temporary data are deleted and the real registration on the AXCS is never performed.

Please note that while this demonstration requires the user to carry out the registration process through a browser, the final version of the system will guide the user through some AXMEDIS compliant tool.

Reporting web service

In order to demonstrate the reporting web service the following scenario will be enacted.

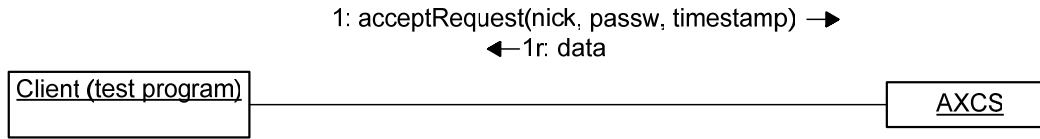


In this scenario a client request a set of ActionLogs specifying a timestamp, which is used to filter all the ActionLogs stored in the AXCS in order to select only those newer of that date.

The data returned are not elaborated in any way because the purpose of the AXCS reporting web service is only to return a selected amount of ActionLogs to be elaborated by the CAMART module.

Statistics web service

In order to demonstrate the reporting web service the following scenario will be enacted.



In this scenario a client request a set of data specifying a timestamp, which is used to filter all the data stored in the AXCS in order to select only those newer of that date. Note that these data represent a subset of those contained in an ActionLog, which cannot be directly returned for privacy issues.

The data returned are not elaborated in any way because the purpose of the AXCS statistics web service is only to return a selected amount of data to be elaborated by the CAMART module.

6.4.5 Accounting and Reporting (EXITECH)

In this section, the Accounting and reporting tools are reported and described. It will be mainly considered CAMART and Administrative Information Intergrator (AII).

6.4.5.1 Contextualisation of CAMART and AII

The Core Accounting Manager and Reporting Tool (CAMART) has two main aspects: the interface toward the system (AXCS and AXDB) and the direct interface toward the user. Both are discussed below. The role of CAMART is strictly bound with database for logs (provided by AXCS) since it has to collect information from AXMEDIS Certifier and Supervisor about Action Log and store them into the AXMEDIS database, homogenizing fragmented information the different tools sent to the AXCS. This is very important as the AXCS or the SuperAXCS will not store data related to each user permanently, but will use policies to remove detailed data, after a certain amount of time. The information is collected according to the schedule and CAMART will act as a client of the AXCS Reporting Web Service.

Administrative and management user uses CAMART user interface to make queries and browse the information related to the usage of the AXMEDIS objects. For example:

For example:

- A Distributor could be interested in seeing the list of Action Logs related to a given second distributor, integrator, etc.
- An Integrator could be interested in viewing the list of Action Logs related to a given AXMEDIS Object, etc.
- A Distributor could be interested in seeing the list of second Distributors that have exploited some specific AXMEDIS object, etc.
- A Distributor could be interested to know how many transactions have been registered on its AXMEDIS objects in the last two months.

The general role of CAMART with respect to the end user that uses it is to provide a web interface for making reporting and statistical queries directly onto the system.

We have to consider in a different way Reporting and statistical queries since:

- Reporting queries are always executed on the AXMEDIS Database Interface for collecting information that are contained in the local database that in turn had been gathered from the AXCS. In this way the actor sees only the data for which is authorized;
- Statistical queries will always be resolved by the SuperAXCS by the means of Statistical Analysis Tool, Web Service. CAMART will act as the client side of the Web Service and will display results to the user in the web interface.

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

Administrative Information integrator is a critical part of the AXMEDIS system since it is the real bridge between the AXMEDIS world and the world of company's CMS and CRM for taking in account administrative and legal aspects (such reclaim for payment not done and so on).

This component has also a double face since it can operate in a dual manner: used for polling information from AXMEDIS system when needed by distributor for example, or user for pushing information in the CMS as soon as they are available for example in the case of collecting societies.

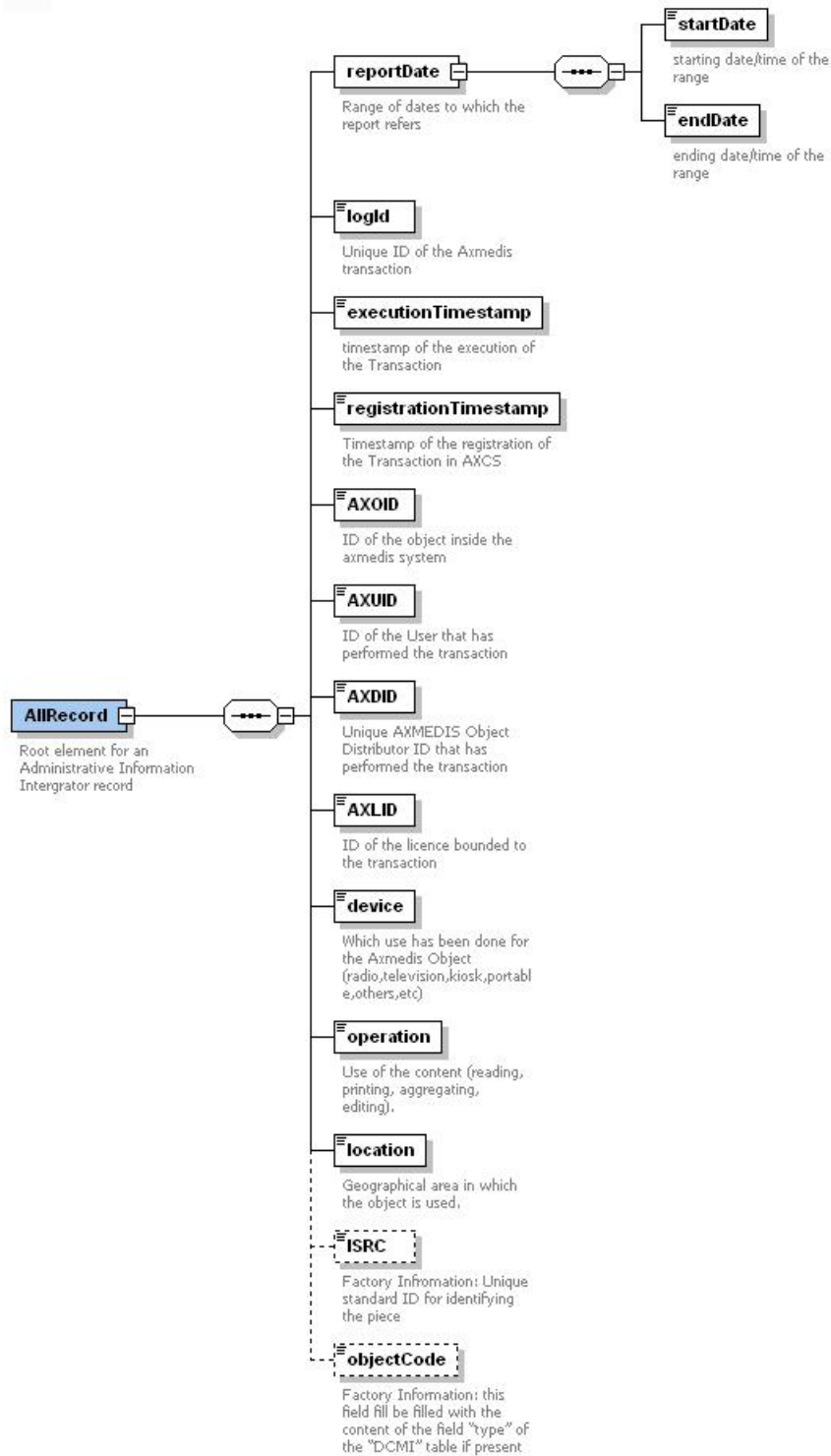
The operating mode is determined by accountant people during the installation/configuration of the system when it will be established whose fields have to be exported from the DB to the CMS and the frequency of exporting. When a frequency is set, the Administrative Information Integrator will work in push mode, pushing information in the CMS import area, otherwise it operates in polling mode by starting the update in the CMS by a link to a web page.

AII is capable of:

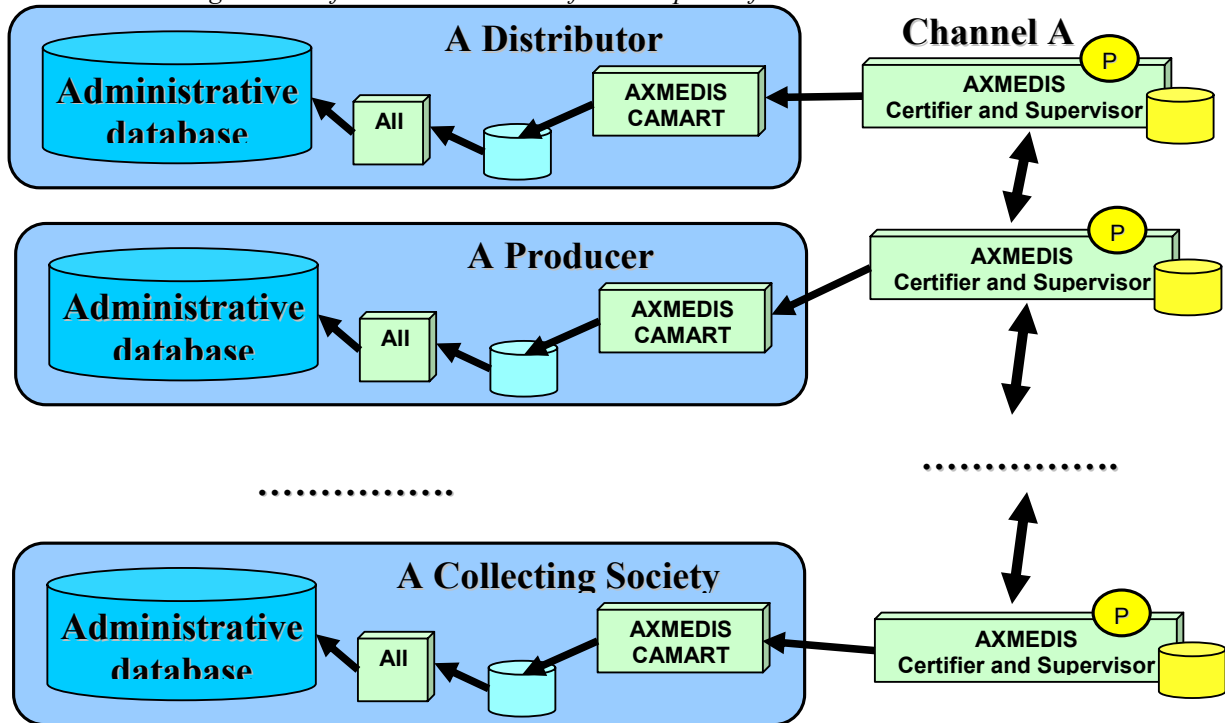
- interfacing with different CMS technologies;;
- store administrative information into the Content Provider database;
- communicate with the AXDB to get administrative information related to a specific Content Provider;
- guarantee privacy of sensitive data via protection mechanism;

AII uses an internal representation of the information that is an XML file that is translated to the different target formats by the means of XSLT files, each one specific for the target CMS.

The common format is reported below:



The relationships among CAMART, AII and the Certifier and supervisors are evidenced in the following picture.



6.4.5.2 Demonstration of All Scenarios and mock-ups

CAMART is a service and not a tool with a user interface, and therefore the user interface is a simple web interface that interacts as a client of the web services. A raw sketch of the user interface is shown below.

The screenshot shows a web browser window titled 'Your Title Here - Mozilla Firefox'. The address bar shows a file path: 'file:///D:/Documents/Docs/Exitech/AXMEDIS/Deliverables/Specification/AXMEDIS-DE3-1-2-AXFW/Current-PART-E/CAMART/alley77564536.html'. The browser's toolbar includes various icons and a search bar. Below the browser window, the 'CAMART Log Extractor' interface is displayed. It features a 'User' input field, a 'Password' input field, a 'Timestamp' input field, and two radio buttons labeled 'Registration' and 'Execution'. Below these are five input fields for 'AXOID', 'AXUID', 'AXCID', 'AXDID', and 'AXWID'. At the bottom of the form are two buttons: 'Extract' and 'Clear'.

This interface allows extracting toward tools that are not the CMS (for which the operation is performed by Administrative Information Integrator) for statistical or reporting purposes. This operation is not bound to the log extraction from AXCS that is performed always in an automatic manner. The data that are extracted here are already present in the AXDB and no communication will be established with AXCS.

In any case, the extracted actions logs are those related to:

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- Distributor, for which AXDID is implicit, optionally asking for logs related to a certain combination of AXCID, AXOID, AXUID, AXWID
- Creator, for which AXCID is implicit, asking for logs related to a certain combination of AXDID, AXOID, AXUID, AXWID
- Collecting societies, for which Nationality is implicit with user name, asking for Logs related to a certain combination of AXWID, AXDID, AXCID, AXOID, AXUID

The user interface for the Administrative Information Integrator is able to configure the tool in the different modes (pushing or polling mode). In the case of pushing mode the frequency at which the push will be made can be set, while in polling mode, the file can be displayed or downloaded. In any case the formatting style can be selected among those present in the database.

The screenshot shows a web browser window titled "Your Title Here - Mozilla Firefox". The address bar displays a file path: `file:///d:/Documents/Docs/Exitech/AXMEDIS/Deliverables/Specification/AXMEDIS-DE3-1-2-AXFW/Current-PART-E/AII`. The browser's toolbar includes various icons for navigation and actions. Below the browser window, the "Administrative Information Integrator Manager" interface is visible. It contains several input fields and buttons:

- User** and **Password** text input fields.
- Exporting Style** dropdown menu, currently set to "CMS1 style".
- Exporting Path for Pushing** text input field.
- Save Configuration** button.
- Pushing Mode:** section with:
 - Enable Pushing Mode** checkbox, currently unchecked.
 - Push Data every** dropdown menu, currently set to "10 minutes".
 - Ok** and **Clear** buttons.
- Polling Mode:** section with:
 - Download** and **View** radio buttons, both currently unselected.
 - Start** and **Clear** buttons.

In the case of pushing mode, a daemon is active, that for each user enabled to get data from Administrative Information Integrator will automatically download in the configured path the fresh logs. System takes care or keeping in the database the timestamp of last download (both for polling and pushing) in order to have for each user a stored profile with the last logs downloaded.

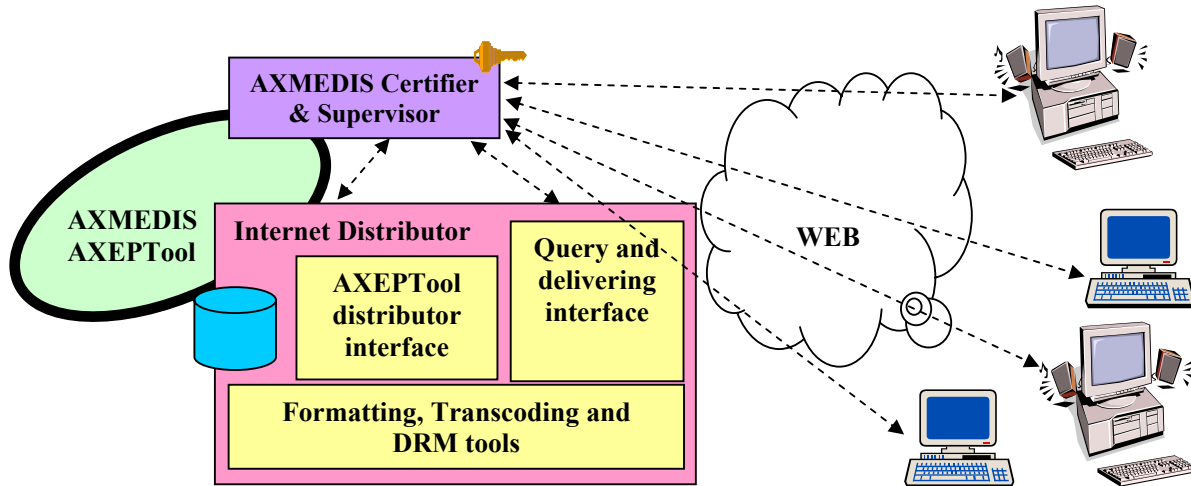
This user interface is for configuring how the AII will interact with CMS in terms of periodic update of the file to be imported by the CMS, the style that have to be applied in order to have multiplatform capability for different CMS. The timestamp of the extraction will be registered automatically by the system and there is no need for the user to take care of that.

6.5 Distribution

6.5.1 Distribution to PC (TISCALI, SEJER)

The main goal is to transform XAURA and the Media Centre, XAURA's main instance devoted to handling multimedia premium content, into stable, documented and fully featured solutions open to integration with all the components of the AXMEDIS project so to enable a complete and easy to use environment for the publication and up-selling of content over the Internet, accessing the content available on the AXMEDIS P2P network AXEPTool.

The basic structure for AXMEDIS content distribution toward Internet is shown below.



AXMEDIS Distribution of PC-TV Content via Internet

The functionalities of the system for content distribution toward Internet could be classified into two groups, the first describes all back office functionalities provided by the system, the second one describes the end user functionalities.

The back office main functionalities are the following:

- Content acquisition within the AXMEDIS P2P network (using the AXEPTool the medioclub manager to search for new content in the network (AXMEDIS Network) acquire a distribution licence for this content (off line step in the first stage) and add the content acquired to the medioclub catalogue
- CRM services: using the (AXMEDIS CAMART) for reporting and customer care purposes

The front-end main functionalities are include those as described in the catalogue browsing and the content/licence acquisition operation (interaction between AMEDIS Player, PMS and Distributor)

6.5.1.1 Demonstration of Internet Distribution via Web Scenarios

The demonstration scenarios focus on the two type of transmissions: the content acquisition from the AXMEDIS B2B network and the AXMEDIS object end user delivering.

Scenario 1: Content acquisition from AXMEDIS Network

1. The Distribution Manager (DM) submit a query search (via AXEPTool) to find contents that should be proposed to the web site users. select content and import data in the distributor CMS.
2. DM analyse the query result and select contents to import in the distribution CMS.

Scenario 2: Publish content via the website

1. The content is integrated with additional information and published via the web site
2. The user, browsing the site, selects, buy and downloads a content available from the list of proposed content.

6.5.1.2 Demonstration of Internet Distribution via Web Tools

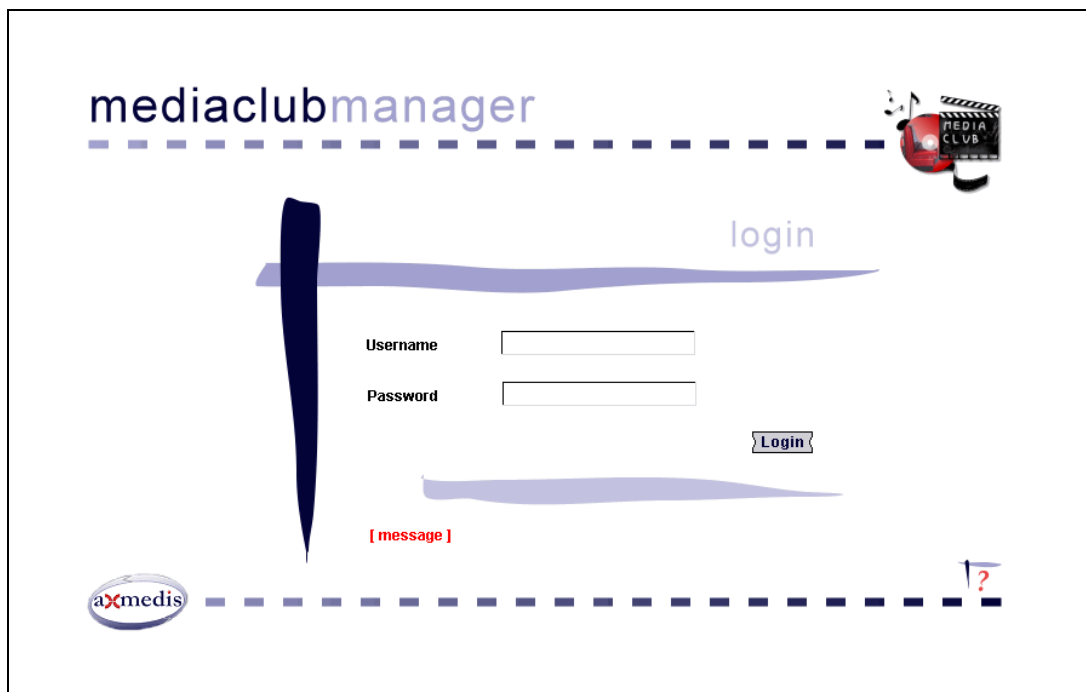
The first prototype of Internet Distribution content system does not fully integrate the AXMEDIS framework, although APIs and interfaces are fully identified and specified in the WP 9.4.1.

The demonstration of the Internet Distribution channel is composed of the following steps:

- the first shows how to search and acquire contents from the AXMEDIS B2B Network;
- second shows content publication in the web site and distribution to the end user.

Scenario 1

The internet content distribution system prototype is designed to be web based, this does not require extra software to be installed in the computer of the user/operator. Distributor manager will login the system using the web-site interface as shown in the figure below.



Screen snapshot of the web interface for login.

Once it is logged in, s/he browses the administration interface using the left column links in the page. For the distributor manager the functionalities allowed at this stage are:

- import media;
- media catalog;
- report & statistics

Import media

This is the interface for querying contents in the AXMEDIS B2B Network. An example Query result is shown in the figure below.

Selecting one of the content listed in the page a content page will be loaded containing all the relevant metadata of the AXMEDIS object selected.

Media Catalogue

From the Content details page the content can be imported in the distribution CMS system (Xaura2). At the end of the import process the media catalogue list page will be loaded as shown in the figure below.

The screenshot shows the 'medioclubmanager' web interface. On the left is a sidebar with three menu items: 'import media', 'media catalog' (selected), and 'reports & statistics'. The main area is titled 'media catalog' and includes an 'Edit catalog' button. Below this is a search bar with a dropdown menu set to 'name', an 'Imported from' date range selector, and a 'to' date field. A status bar indicates '60 Records found', 'Page 1/10', and 'Result for page: 6'. A table displays the following data:

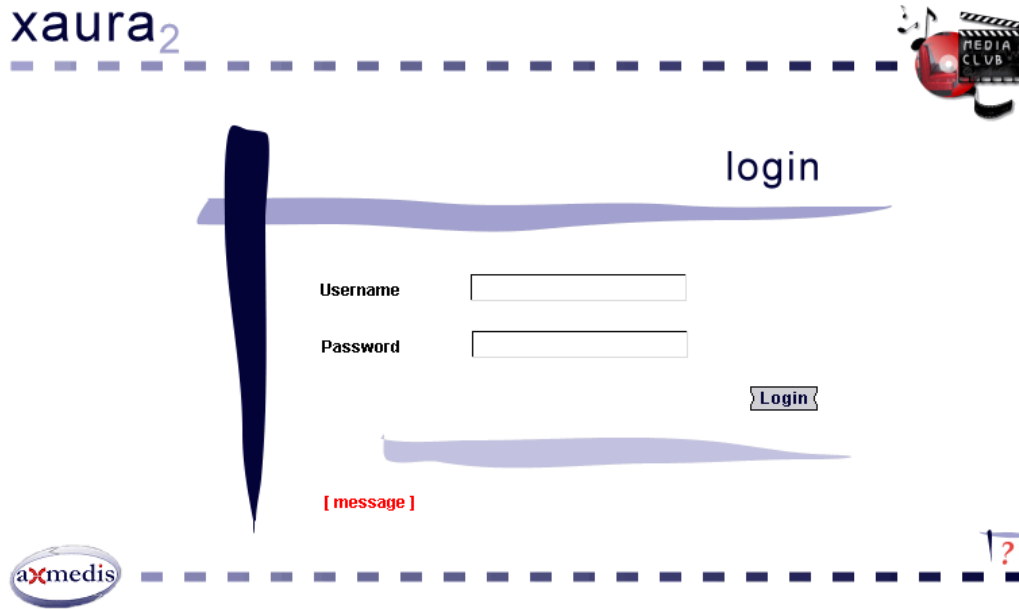
#	id	name	type	author	imported	
1	123456789	this an axmedis media	document	author	2005-08-01 09:00	
2	123456789	this an axmedis media	image	author	2005-08-01 09:00	
3	123456789	this an axmedis media	document	author	2005-08-01 09:00	
4	123456789	this an axmedis media	document	author	2005-08-01 09:00	
5	123456789	this an axmedis media	video	author	2005-08-01 09:00	
6	123456789	this an axmedis media	video	author	2005-08-01 09:00	

At the bottom of the table area is a '[message]' link and pagination controls. The footer features the 'axmedis' logo.

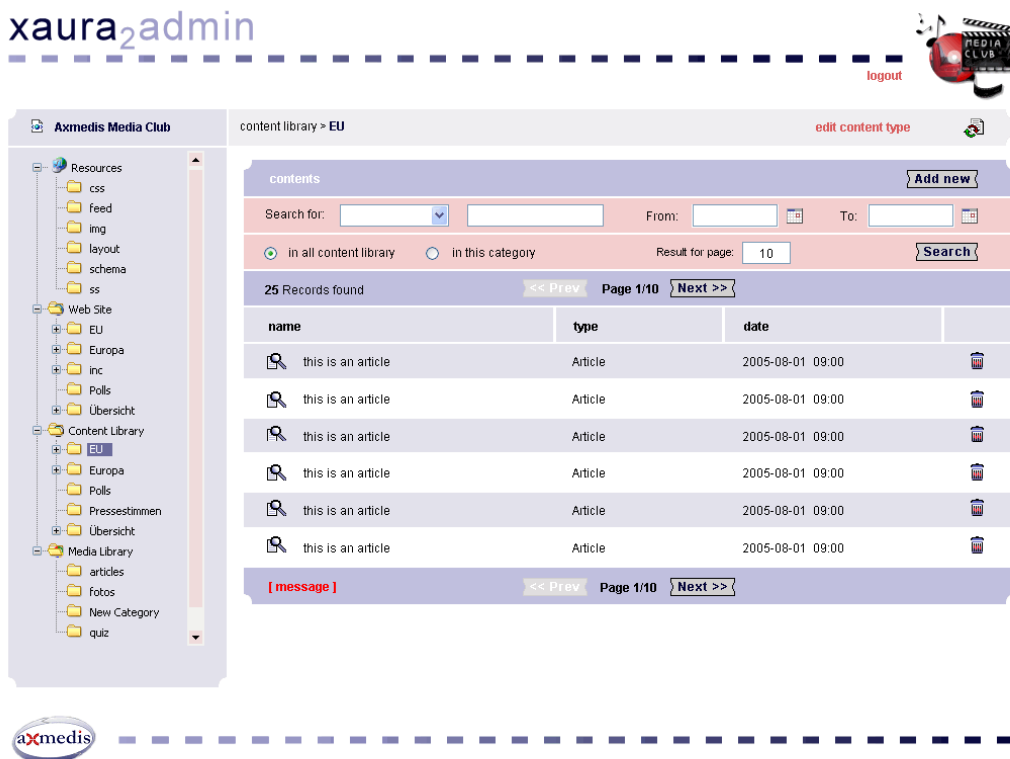
At this stage the content is fully imported in the Distribution CMS and is ready to be managed from the editor to publish the content in the web site.

Scenario 2

The Editor will login in the system using the web site interface as shown in the figure below.





Once it is logged in, he brows the administration interface using the left column links in the page. To access to a content the editor brows the left page three clicking in the Content Library folder.



At this stage the editor can search in the catalogue the content that need to be edited and published using the search box or simply browsing the content listed in the page.

Selecting on content the content page will be loaded providing al information present in the CMS for that content.

xaura₂admin  

Axmedis Media Club

- Resources
 - css
 - Feed
 - img
 - layout
 - schema
 - ss
- Web Site
 - EU
 - Europa
 - inc
 - Polls
 - Übersicht
- Content Library
 - EU
 - Europa
 - Polls
 - Pressestimmen
 - Übersicht
- Media Library
 - articles
 - Fotos
 - New Category
 - quiz

content library > EU > **this is an article**

content details

content name:

type: article

date: 2005-08-01 08:00


categories

- ☐ EU
 - ☐ finanzierungen
 - ☐ glossar
 - ☐ wegweiser
- ☐ Europa

[message]

publish

name	website	section	publish	
EU	web	eu	<input type="checkbox"/>	2004-08-01 09:00
Euro News	web	eu/news	<input type="checkbox"/>	2004-08-01 08:10
Euro Meteo	web	eu/meteo	<input type="checkbox"/>	

[message] 

content editor

[message]

Font... Size...      **B** **I** **U**      

Title > Wer wird gewählt?

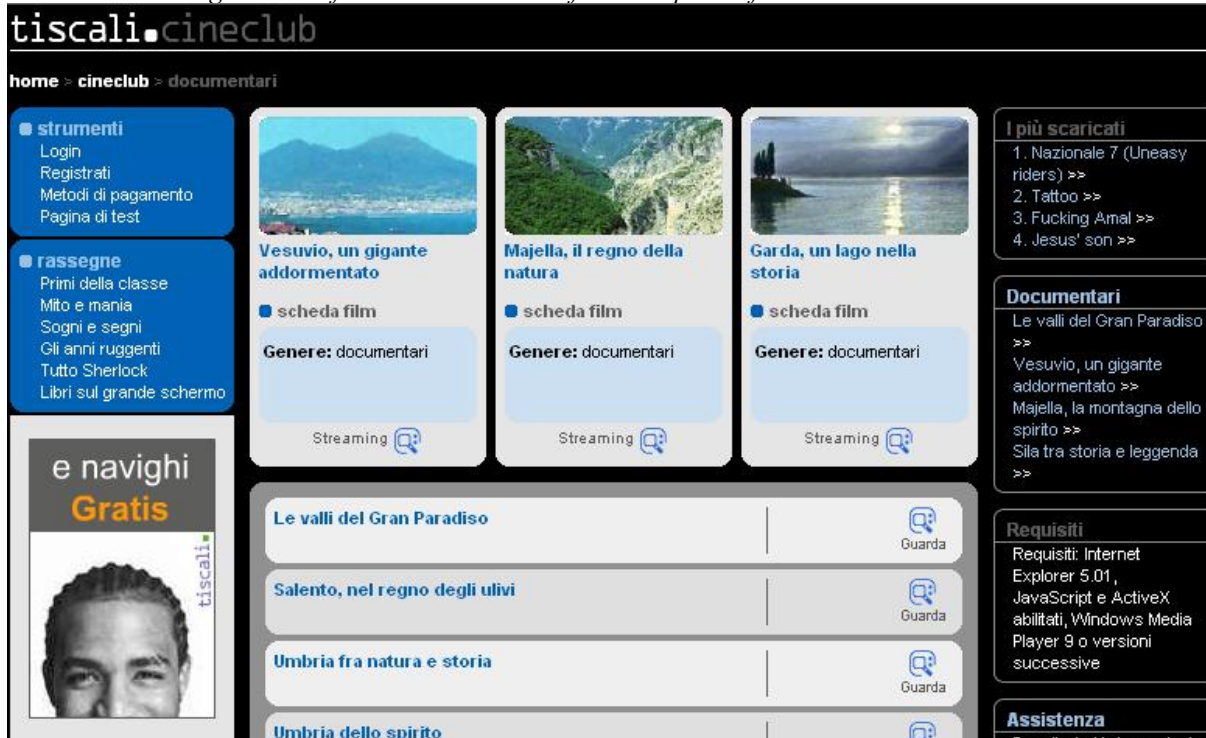
Subtitle > Schluss mit Doppelmandaten

Abstract Text > Grundsätzlich gilt: Wählbar ist, wer am Tag der Wahl die deutsche oder die Staatsangehörigkeit eines anderen Mitgliedlands der EU hat. Auch hier gilt die Voraussetzung des vollendeten achtzehnten Lebensjahrs. Die Kandidaten für die Europawahl werden von Parteien oder sonstigen politischen Vereinigungen aufgestellt und auf einer bundesweiten Liste (Bundesliste) bzw. auf den Listen einzelner Länder (Landeslisten) platziert. Die endgültige Entscheidung darüber, welche Listen und somit welche Kandidaten zur Europawahl zugelassen werden, treffen der Bundeswahlausschuss bzw. die Landwahlausschüsse.

The editor edit the content adding some information like price, assigning the web site categories where the content will be published and, at the end of this process, decide to publish the content in the web page.

Now the content is published and available in the website for the users that can access to it.



6.5.2 Distribution to Mobiles (COMVERSE, DSI, ILAB, IRC)

Demonstration on mobile platform include a player for mobile phone (DSI), and many other related aspects including:

- DRM support from user clients. This task includes the implementation of the common profile between MPEG-21 REL and OMA/ODRL. This work is to be performed within WP4.7 and the results is to be applied to WP9.5
- MPEG-21 DIA expressions for supporting mobile transcoding, includes the description of MPEG-21 DIA expressions for the authorisation of operations in the mobile environment. This work is to be carried out in WP4.3 with WP4.5 and results applied to WP9.5.
- With WP9.5, ILABS is to manage the production of a demonstrator to distribute mobile content for educational purpose (mainly) via GPRS. Creating in their authoring and client applications and AXOM plugin. While the server will be mainly for the front end LexMOBILE and for the adaptation AXCP.
- AXMEDIS Plugin solution for the identified algorithm to process ringtones as described in the DE4.7 is to be developed by IRC
- etc

6.5.2.1 Demonstration of licence translation for the mobile scenario

Regarding distribution for mobile scenario, there is an important issue related to the format of the licences that can be used in them. From the one side, Open Mobile Alliance is describing OMA DRM REL, a rights expression language derived from Open Digital Rights Language (ODRL), adapted for the available operations in the mobile environment. From the other side, MPEG-21 standard is defining its own rights expression language as its fifth part, MPEG-21 REL. In this standard, it is also described an authorisation model that allows the authorisation of actions that can be done by the different actors in the distribution chain (from content creator to final user), on the basis of the licences they own. MPEG-21 REL language is

the one selected by AXMEDIS project to initially implement its DRM tools, as many of the features implemented in this project are based on the different parts of the MPEG-21 REL standard.

In this context, and in order to relate these two worlds, we will demonstrate how licences described in one language can be translated to the other language, if both of them accomplish the same profile, that is, a common subset of both languages that can be translated in both senses.

Translating a licence from language to language provides several advantages. One of them is that a user can purchase a licence for its PC expressed in MPEG-21 REL and use the same content (or a transcoded version) in its mobile, with the corresponding translated licence, understandable by its mobile. Other advantage could be the other way around, a user has purchased a licence to consume a content for its mobile, but he also wants to consume it on its pc or other player, which understands MPEG-21 REL licences. This situation will be more likely to occur when mobiles increase their screen and processing capabilities.

6.5.2.2 Demonstration of licence authorisation for the mobile scenario

Related to the section, we could demonstrate how we can authorise actions for the mobile environment, combining and/or translating licences from different languages in order to have all of them expressed in MPEG-21 REL language. Then, the authorisation algorithm defined for MPEG-21 REL can be applied and the authorisation can be given to the user or not, depending on the licences and the context.

6.5.3 Distribution via Kiosks (ILABS)

The presentation of the Kiosk-based distribution scenarios was derived from the Use Cases in DE2.1.1b Section 17.

Introduction

What is an AXMEDIS kiosk? Well this is a specialised distribution channel operating at B2C level and covering a quite dispersed set of user needs and activities. In more details it represents a point of service inside an infrastructure that may be spread over the territory with multiple differentiated instances all related to each other in terms provided content (museums kiosk will all deliver content related to the museum net while tourism devoted kiosks will provide content related to tourists activities and interests) with a common usage of technology and distribution tools. Lastly, kiosks will be characterised by the availability of a LAN where will be connected point of services (terminals) and wireless PDA.

How does the kiosk area fit in to the rest of AXMEDIS? The kiosk sits at the Distributors level (as shown in the following figure)

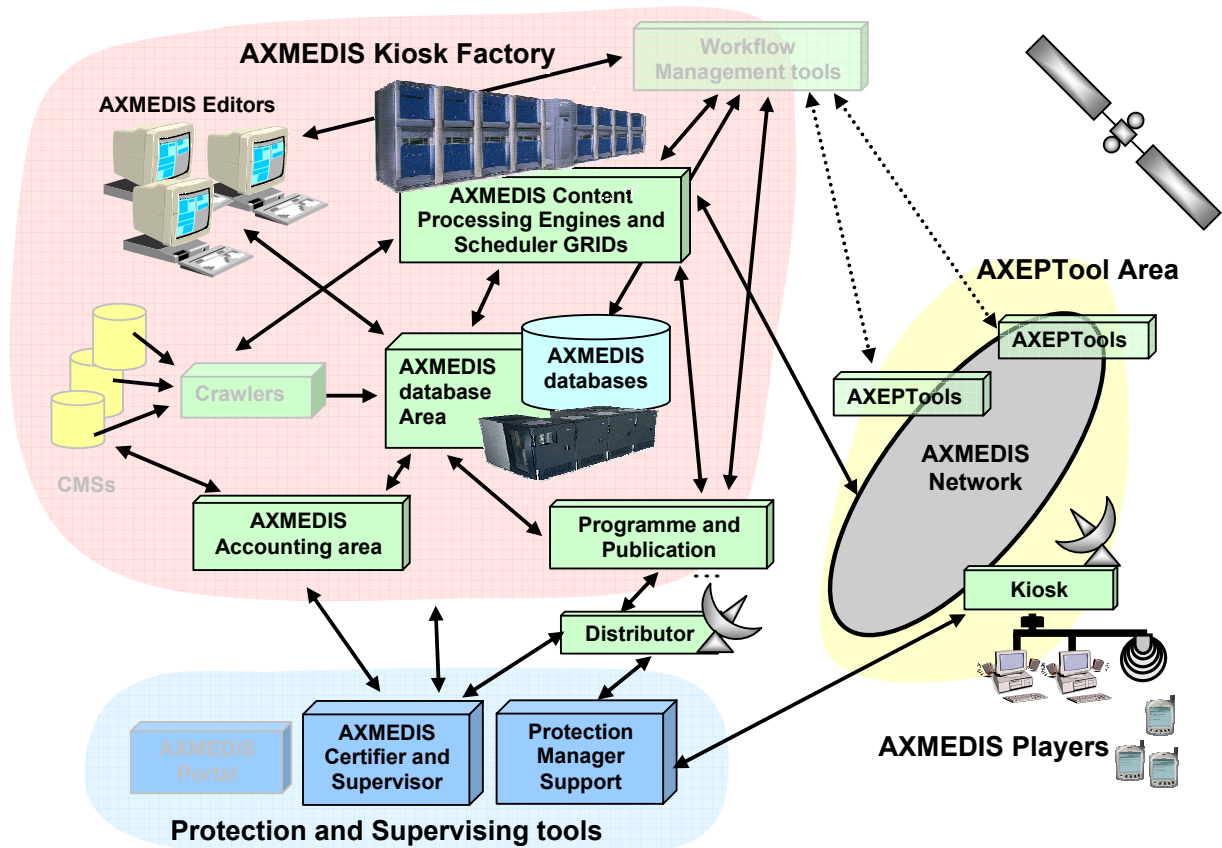


Figure: AXMEDIS Architecture highlighting the Kiosk Area

To deliver AXMEDIS objects using kiosks, there are two scenarios:

Scenario 1. In the first scenario a publisher prepares content that wants to distribute on a particular target kiosk set (e.g. museums) via satellite. The publisher needs to select a list of objects, a

catalogue template, an object template and specify when to deliver each object (including the catalogue) and to whom (i.e. the distribution kiosk).

Scenario 2. In the second (On-Demand) scenario, a user selects a multimedia object in the AXMEDIS Distribution Area. The request from the distribution area starts a series of events behind the scenes where the user sees the object requested.

Scenario 1: Kiosk Content and Catalogue Production & Distribution

A publisher prepares content that wants to distribute on a particular target kiosk set (e.g. museums) via satellite. The publisher needs to select a list of objects, a catalogue template, an object template and specify when to deliver each object (including the catalogue) and to whom (i.e. the distribution kiosk).

By using the AXMEDIS Query support the publisher searches the desired objects and selects the one to be used to build the catalogue.

By using the AXMEDIS Editor and a specific application the publisher produces the catalogue starting from a set of pre-defined templates.

By using the P&P Editor the programme manager can create a programme to specify the time and destination channel of the objects to be distributed. On completion, the publisher activates the programme and during the programme life cycle the objects will be distributed until the programme has completed or the programme is removed from being activated.

Scenario 2: Kiosk Catalogue and Content Distribution & Fruition

So what happens when a kiosk is used? The user shall register or log onto the kiosk infrastructure to be recognised and authorised to access to provided services. The user will be able to browse the catalogue and select content for delivery and fruition. In this latter phase the user will be able to select also the fruition model (acquisition, rental, pay per use...) and experiment what it looks like a DRM empowered system specifically designed to cover the whole value-chain from production to fruition. As a matter of facts the system will show how only allowed operation can be performed while all others are inhibited.

Conclusion

The kiosk environment shall enable to test basically all components of the AXMEDIS framework partially in the content preparation and primary distribution (that will be performed at Kiosk-Factory level) and partially at end user level during fruition at kiosk level. Moreover this demonstration scenario covers the benefit of satellite based targeted broadcast of “content” proving how it is feasible to achieve a high degree of efficiency and flexibility in “specialised” content distribution on a “geographically-dispersed” environment. At user level this demonstrator will trial solution related to combination of business model merging into a single local infrastructure for content search, selection, acquisition and fruition.

Learning Outcomes

At the end of this course/demonstrations, attendees will understand how to:

- search and select content,
- select a distribution channel and distribute content,
- acquire content,
- use acquired content.

6.5.4 Distribution via Satellite Data Broadcast (EUTELSAT)

One of the distribution mechanisms used by AXMEDIS is the Satellite Data Broadcast. After a programme has been created with the P&P Editor, it is scheduled using the P&P Engine, to be sent through the selected distribution channel, the satellite in this case (for P&P see Section: Programme and Publication).

The satellite channel distribution is particularly recommended to deliver content, at the same time, to several clients spread all over the territory at the same time. Intended uses are, for example, the update of

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools distributors' mediatheques, or the kiosks servers in the Kiosk distribution (see Section: Distribution via Kiosks). Satellite broadcast suits both B2B and B2C domain.

The picture below highlights the role of the Satellite Data Broadcast inside the AXMEDIS architecture.

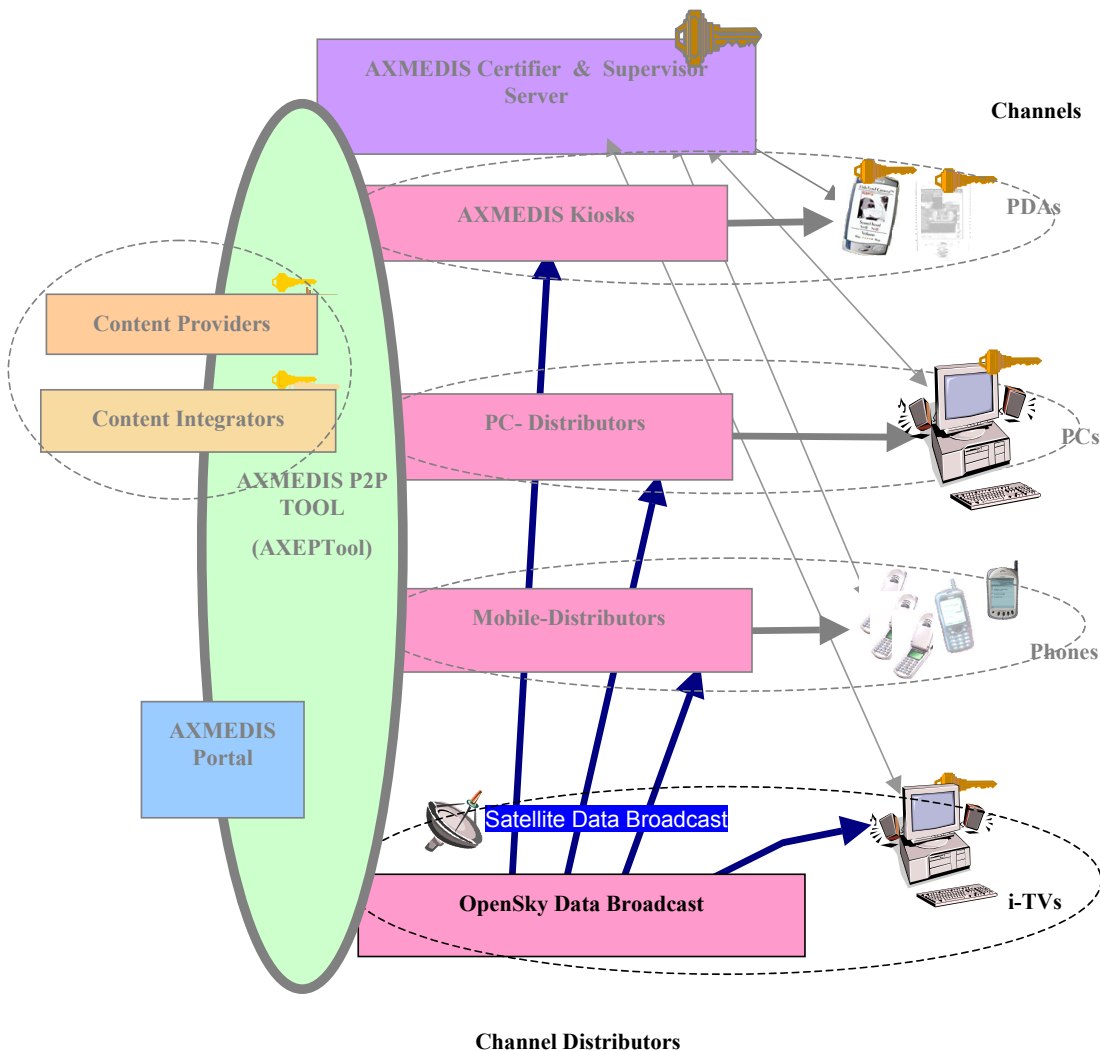


Figure: AXMEDIS Architecture highlighting the Satellite Data Broadcast among the channels distributors

6.5.4.1 Demonstration of Satellite Distribution Scenarios

The demonstration of the Satellite Data Broadcast scenarios has been derived from the Use Cases in DE2.1.1b.

The demonstration scenarios focus on the two sides of the transmission: the sender and the receiver.

Scenario 1: Sending Content through Satellite Data Broadcast

1. The content is upload and packaged according to the transmission channel's rules
2. The content is scheduled as a program, specifying the when and how of the transmission
3. The program is activated / paused / stopped

Scenario 2: Receiving Content via Satellite Data Broadcast

1. The user selects and downloads a content available from the list of proposed content on air

2. The user station automatically receives content that has been addressed to it, without explicitly selecting it

Scenario 3: Filtering Content and Cache Strategies

1. The content carousel is optimised according to content filtering on the client side and cache management
2. Personalised content composition

6.5.4.2 Demonstration of Satellite Distribution Tools

The first prototype of the P&P Engine does not yet integrate a real interaction between the P&P Engine itself and the Satellite Data Broadcast powered by EUTELSAT, although APIs are already available in this order. That's the reason why the demonstration of the Distribution via Satellite is, at this stage, led separately. The demonstration of the Satellite Distribution channel is composed of two parts: the first one shows how to prepare a transmission via Satellite and the second one shows what happens on the receiving side.

Scenario 1

This scenario includes all the steps required to use the satellite channel as delivery channel. The same operations that will be later automatically produced by the P&P Engine are, for the moment, reproduced on the EUTELSAT platform using a Web Interface.

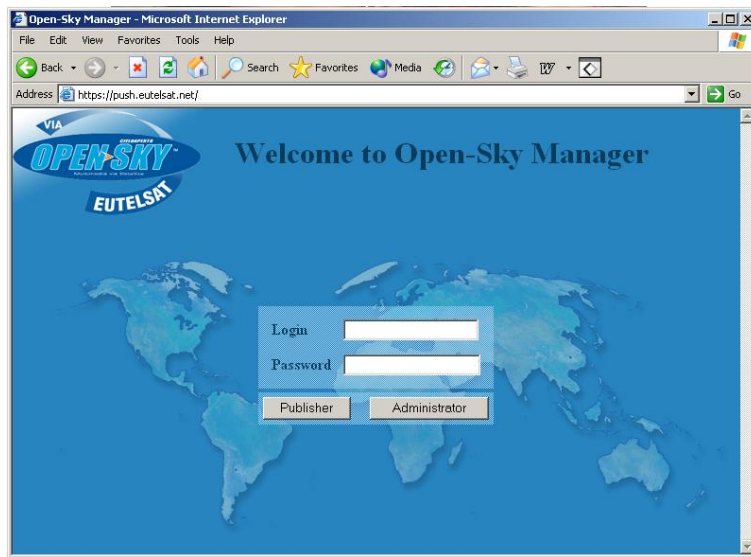
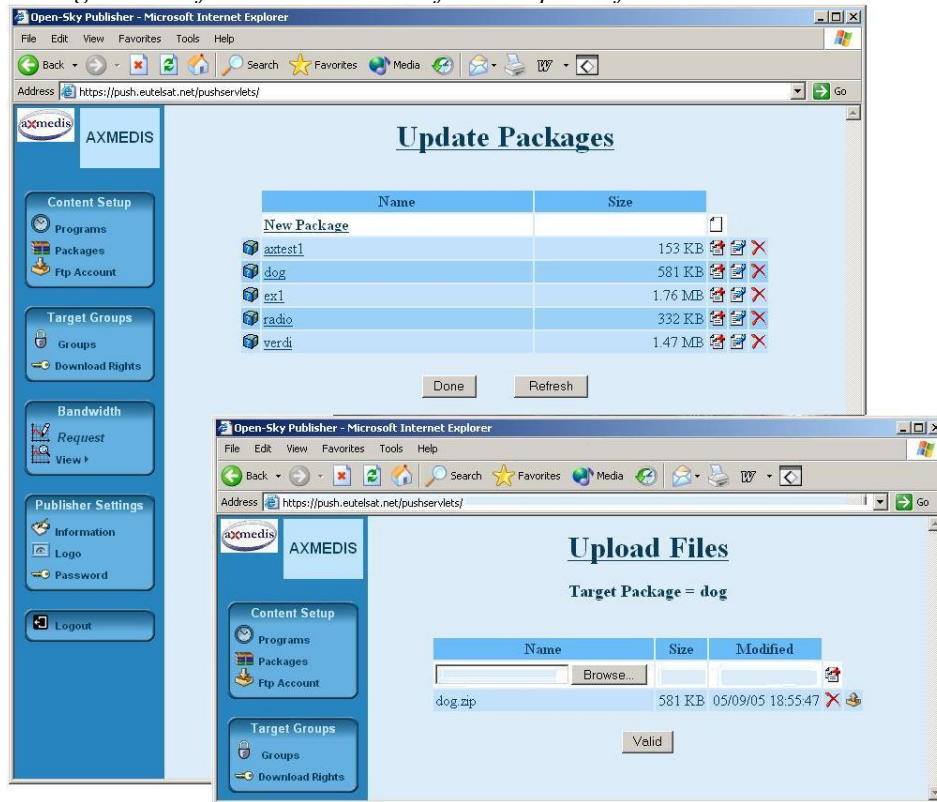


Figure: OPENSKY Web Interface

Through this interface, the programme created by the P&P is transposed into an OPENSKY program, and then scheduled again according to the choices made with the P&P Editor.

Content Packaging

The content has to be transformed in an OPENSKY package before being inserted into a program. The OPENSKY package contains a set of metadata used by the satellite distribution protocol. Once the package created, the content is associated uploading it on the OPENSKY platform.



OPENSKY Content Packaging

Program Scheduling

A program is a collection of content with a transmission schedule. Packages are grouped together inside a program, and then the program is scheduled for transmission. The schedule defines the speed (bandwidth), the start, the duration and possible repetitions of the distribution.

Open-Sky Publisher - Microsoft Internet Explorer

Address: <https://push.eutelsat.net/pushservlets/>

AXMEDIS

Schedule Program

axmedis
4.30 MB at 512 Kbps takes 2 min 36 sec.
1 transmission of 3 days from the 30/11/2005, 16:00 to the 03/12/2005, 16:00.

Program Type: **Carousel** Loop transmission of the same static content. You have to pause the transmission in order to add or modify content.

Bandwidth: 64 512 10240

Program Start / End:
Start Date: 30/11/2005 16:00 ☐ Now
☒ End ☐ Date 03/12/2005 16:00 ☐ Duration 3 days

Slot Length:
☐ Duration 3 days ☒ # full sequence 1652

☐ Periodicity
every 0 min # repetition 1

Preview: 512 kbps
3 days
30/11/2005, 16:00 03/12/2005, 16:00

Figure: OPENSky Program Scheduling

Program activation / stop

When the program has associated a schedule, the program enters an “active” state. When is such a state, the program can receive commands like play / pause / stop.

Open-Sky Publisher - Microsoft Internet Explorer

Address: <https://push.eutelsat.net/pushservlets/>

AXMEDIS

Update Programs

Name	Size	Status	Action
New Program			
axmedis	3.71 MB	init	⏮ ⏪ ⏩ ⏭ 🔄 🗑

Figure: OPENSky Program Updating

Scenario 2

The reception is then validated on a client station previously installed with needed material, i.e. DVB card connected to a dish. The AXObjects sent with the programme are received on the enabled station, and stocked in the AXMEDIS cache area for further filtering. The pictures below show the GUI of the client application from which is possible to know which content is on air, what is currently download, and the content already received.



Figure: Program Guide with on-air content



Figure: List of received content

Note that this graphical application is used to better explain how the reception works. It is a merely visual aid on the demonstration. The client application for the satellite data broadcast proposed by EUTELSAT includes this GUI, but, of course, it is not essential. The core of the client application is composed by other components that run in background and automatically let the receiving station download and store the content, without the necessity of a human interaction (except for the initial configuration phase).

Scenario 3

Once the AXMEDIS content is stored in the cache area, different application models are possible. The first one includes the optimisation of the content carousel, the client-side content filtering and the cache management. This aim shall be addressed by the definition of suitable algorithms (resulting from research activities) able to implement processes to adapt contents fulfilling users preferences.

The second model represents an application enhancement by including personalised content composition feature (customised AXMEDIS channels build up). The mentioned personalisation occurs taking in account appropriate profiles as result of the user choices/preferences. This information shall be stored in files which are compliant to MPEG 7 standard (ref. DE4.8.1) fitting the application requirements and managed by the user through a suitable graphical application

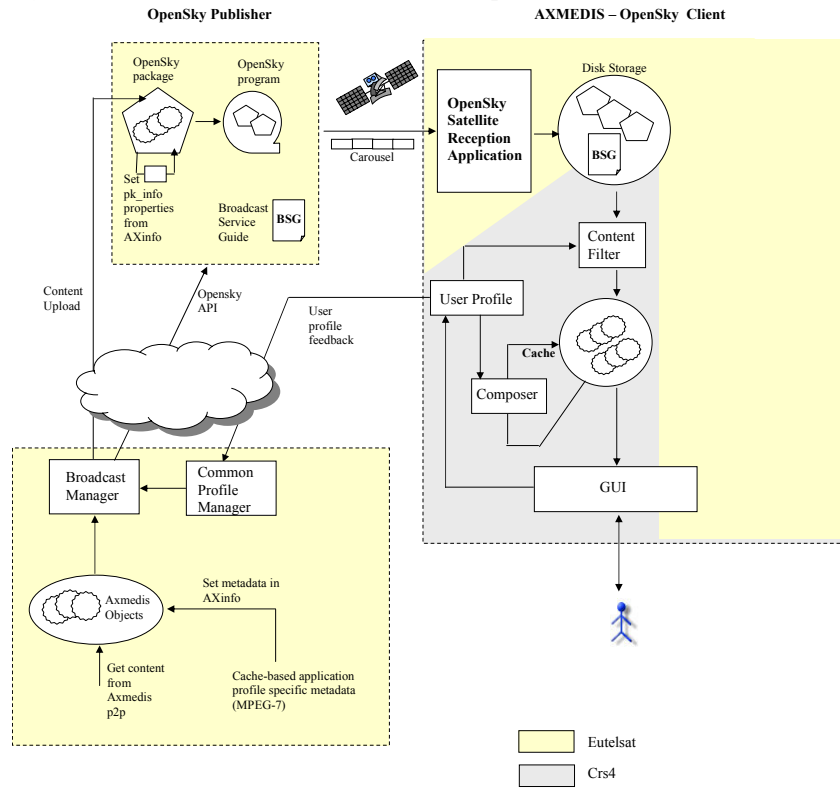


Figure: General Architecture of Scenario 3

Conclusion

This demonstration session can be seen as a completion of the Programme and Publication Area. The tutees will have an idea of what is beside the creation and schedule of a programme, and what is the final result on the destinations.

Learning Outcomes

At the end of this demonstration session, the attendees will understand:

- How the Satellite Data Broadcast fits within the AXMEDIS architecture and with which advantages
- How the content is received on the client stations, with or without human interaction (selection)
- How the content is handled on the client stations, with filtering and personalized composition on cache

7 AXMEDIS Training Material Publication (ILABS, UNIVLEEDS, DSI, ...)

Given the specific project nature and the limitations in cost it is recommended to have only training material produced and delivered in digital format. Documents will be delivered in closed and secured PDF or in PS format.

The final draft of the document should be delivered to the dissemination responsible in a format suitable for finalisation. Once received, the document will be validated (in respect to policies and formats described earlier on in the present document and then endorsed and closed. Once closed, the PDF / PS version will be produced and loaded in the final location of the project website (as described earlier on). Please remember that Acknowledgments are a due part of the publications and that IPR ownership of content has to be cleared as this are going to be public documents.

8 Bibliography

- AXMEDIS DE7-2-1 template resource – Training Manual blueprint, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1183
- AXMEDIS DE7-2-1 template resource – Training Slides, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1184
- AXMEDIS DE7-2-1 template resource – Training Event Questionnaire, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1185
- AXMEDIS DE7-2-1 template resource – Training Attendance sheet, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1186
- AXMEDIS DE1.7.1 Project Presentation, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1440
- AXMEDIS2.1.1 User Requirements, http://www.axmedis.org/documenti/view_documenti.php?doc_id=570
- AXMEDIS DE2.3.2 User Group Maintenance, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1358
- AXMEDIS DE4.7.1 Distribution to Mobile, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1453
- AXMEDIS DE4.4.1 Content Sharing and Production on P2P, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1451
- AXMEDIS DE4.5.1 Content Protection and Supervision, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1430
- AXMEDIS DE5.1.1 AXMEDIS Framework Infrastructure, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1201
- AXMEDIS DE4.1.1 Content Modelling and Management, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1426
- AXMEDIS 4.3.1 Content Composition and Formatting, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1359
- AXMEDIS DE4.6.1 Content Distribution via Internet, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1335
- AXMEDIS DE4.9.1 Usability Issues for the AXMEDIS Production Tools, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1280
- AXMEDIS DE5.2.1 AXMEDIS Framework Validation, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1267
- AXMEDIS DE9.1.1 Specification of CMS Integration and Feedback, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1269
- AXMEDIS DE9.2.1 Specification of Automating Content Production and Formatting into CMSs of Integrators, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1418

DE7.2.1 – Training Material for the Introduction of User Aspects of AXMEDIS Production Tools

- AXMEDIS DE9.4.1 Specification of content production and distribution on-demand for PC, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1365
- AXMEDIS DE9.6.1 Specification of Content Production and Distribution to Kiosks and local PDAs, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1262
- AXMEDIS DE9.3.1 Specification of Content Production and Distribution in Push and for i-TV, http://www.axmedis.org/documenti/view_documenti.php?doc_id=1292