MIPAMS v2.1 Architecture

MIPAMS (Multimedia Information Protection and Management System), http://dmag.ac.upc.edu/mipams/, is a service-oriented secure content management and distribution platform developed by the DMAG-UPC (Distributed Multimedia Applications Group, Universitat Politècnica de Catalunya), http://dmag.ac.upc.edu, and currently exploited by the company mediaTG, http://mediatg.com.

The MIPAMS architecture is based on the flexible web services approach, as it consists of several modules and services, which provide a subset of the whole system functionality needed for governing and protecting multimedia content. One of the advantages of having service-oriented functionality relies on the possibility of decoupling it into different subsystems depending on the needs of the application that is going to be implemented, while being able to share the same common services between different applications with different requirements, thus reducing costs. MIPAMS encompasses an important part of the content value chain, from content creation and distribution to its consumption by final users.

Figure 1 depicts the MIPAMS architecture. A general overview of its components and the different services being offered (in alphabetical order) is provided next.

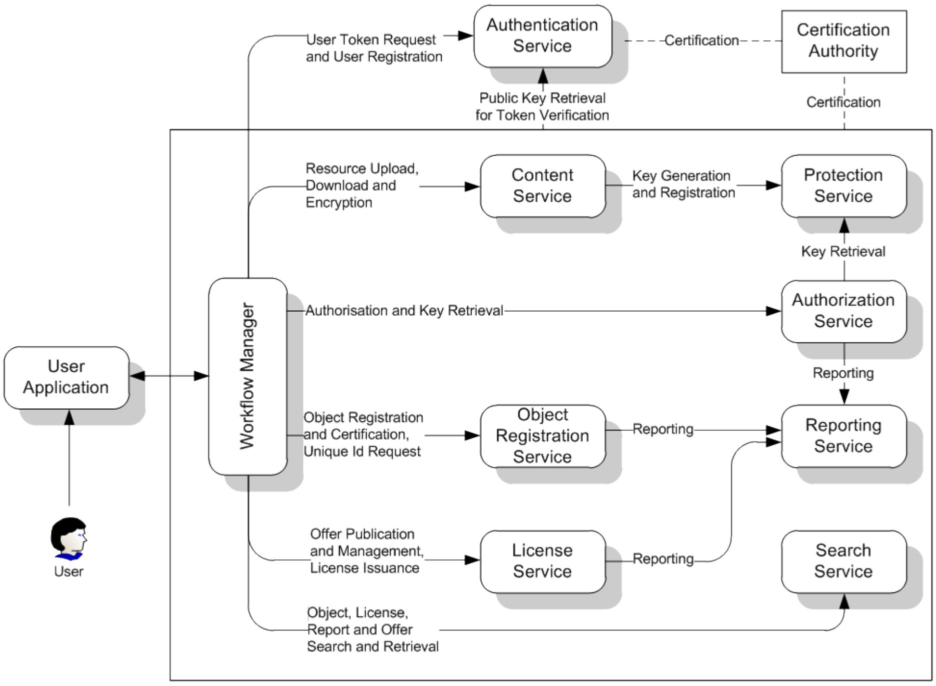
User Token Request and User Registration
Authentication Service
Certification
Certification Authority
Public Key Retrieval for Token Verification
Certification
Resource Upload, Download and Encryption
Content Service
Key Generation and Registration
Protection Service
Key Retrieval
Authorisation and Key Retrieval
Authorization Service
Reporting
User Application
Workflow Manager
Object Registration and Certification, Unique Id Request
Object Registration Service
Reporting
Reporting Service
User
Offer Publication and Management, License Issuance
License Service
Reporting
Search Service
Object, License, Report and Offer Search and Retrieval

Figure 1. MIPAMS v2.1 architecture

MIPAMS Services

The Authentication Service (ATS) is needed to authenticate the identity of users. It generates SAML (Security Assertion Markup Language) based tokens that identify MIPAMS users. Any service in the MIPAMS architecture will require a token argument to be provided in order to authenticate users. Tokens are digitally signed, so that they can be checked for authenticity and integrity by the receiving service. Moreover, the ATS deals with user registration and management (i.e. personal data modification, user account deactivation, etc.).

The Authorization Service (AS) checks whether a user owns any appropriate license that grants him the right to perform a requested action (e.g., play) over a digital object. The authorization is based on the mechanism defined in MPEG-21 REL (Rights Expression Language) standard. The AS shares the access to the license repository with the License Service. If the user is able to perform the action and the requested content is encrypted, the AS will retrieve the encryption keys from the Protection Service and return them to the requesting application. This is the only means for accessing encryption keys, which is performed as an atomic operation.

The Certification Authority (CA), which issues credentials for the different Components and Actors in the system, as X.509 certificates and private keys for the different architectural components.

The Content Service (CS) enables applications to upload and download digital resources such as audio or video files, text documents, etc. Those resources can be optionally encrypted under request, according to the available encryption mechanisms it provides. If encryption is selected, the protection keys will be first requested to the Protection Service and then registered through the same service, once encryption is performed. Content upload requires content to be uniquely identified. Since MIPAMS deals with single resource objects, the identifier being associated to content will be the same one used for the object that contains it, and must be passed as input argument. This identifier can be requested to the Object Registration Service prior to the content upload, or obtained from an external application using MIPAMS (it depends on the scenario).

The License Service (LS) deals with rights offers and the issuance of licenses. Rights offers are set up by content creators or rights holders after registering content. They include the rights being offered for acquisition by other users and the conditions being applicable to those rights. License issuance refers to the process by which a license is generated as the result of a rights purchase, acquisition or because a rights holder directly grants some user a set of rights. Licenses are expressed using MPEG-21 REL.

The Object Registration Service (ORS) enables applications to request a digital representation of content and metadata (i.e. digital objects) to be generated and registered in the system. Content and metadata are packaged together following the MPEG-21 Digital Item approach. Once registered, objects are digitally signed by the ORS so that they can be checked for authenticity and integrity. The ORS also provides unique identifiers for those applications that need to upload content to the CS, as already explained.

The Protection Service (PS), as introduced before, generates encryption keys upon request, registers encryption keys associated to uniquely identified content and provides the encryption keys for protected content to the AS. When using MPEG-21 Intellectual Property Management and Protection scheme and descriptors, the PS also offers the possibility to download the protection tools being used by those applications that might be out-of-date.

The Reporting Service (RS) collects usage reports regarding the registration of objects, the issuance of licenses and the authorizations being performed. It is also capable of building standards-based representations of those reports, such as MPEG-21 Event Reports. Those reports may be used for computing statistics as well as for billing or tracking purposes.

The Search Service (SS) enables applications to perform accurate searches amongst metadata in the MIPAMS system. That is, it is the front-end for requesting any information present in MIPAMS services databases. Thus, it can be used for searching content, licenses, offers or reports or a combination of them.

The User Application (UA) is the player, edition tool, browser or any other means that is managed by the user to deal with the DRM functionality, such as registering and accessing protected contents. The UA may have an internal trusted module or intermediary to enforce DRM, which could consist of a secure local repository for licenses, protection information, offline operation reports and other critical data. In those cases, it may be responsible for estimating tool fingerprints, require offline authorizations, unprotect content, track offline operations and manage content protection information.

The Workflow Manager (WM) may be an integral part of the UA or otherwise be located in the server part (e.g. web portal, brokerage service) to reduce the UA complexity. It can be seen as a broker to whom the UA requests different operations to be performed, as object registration, content upload, rights offer management, license acquisition, authorization, etc.

mediaTG media platform (mtPlatform)

mediaTG media Platform (mtPlatform) is a service-oriented secure content management and distribution platform developed by the company mediaTG (http://www.mediatg.com) in collaboration with the DMAG-UPC. This middleware software platform is implemented using REST-based web services (REST, Representational State Transfer). In this way, it has the advantages of service-oriented functionality, like the possibility of decoupling into different subsystems, whilst providing a more efficient and light-weight web service infrastructure. mtPlatform encompasses an important part of the content value chain, from content creation and distribution to its consumption by final users.

mtPlatform is an evolution of MIPAMS v 2.1 and includes the following new features and improvements:

- REST-based web services design
- Complete refactoring of the web services provided by the system, according to REST design principles, that is, based on resources and not in remote operations. This refactoring has increased service productivity
- Implementation of new services to provide billing, advanced contract management, transcoding and refactoring of object registration, license, protection and upload / download services.

Figure 2 depicts the mediaTG media Platform. A general overview of its components and the different services being offered (in alphabetical order) is provided next.

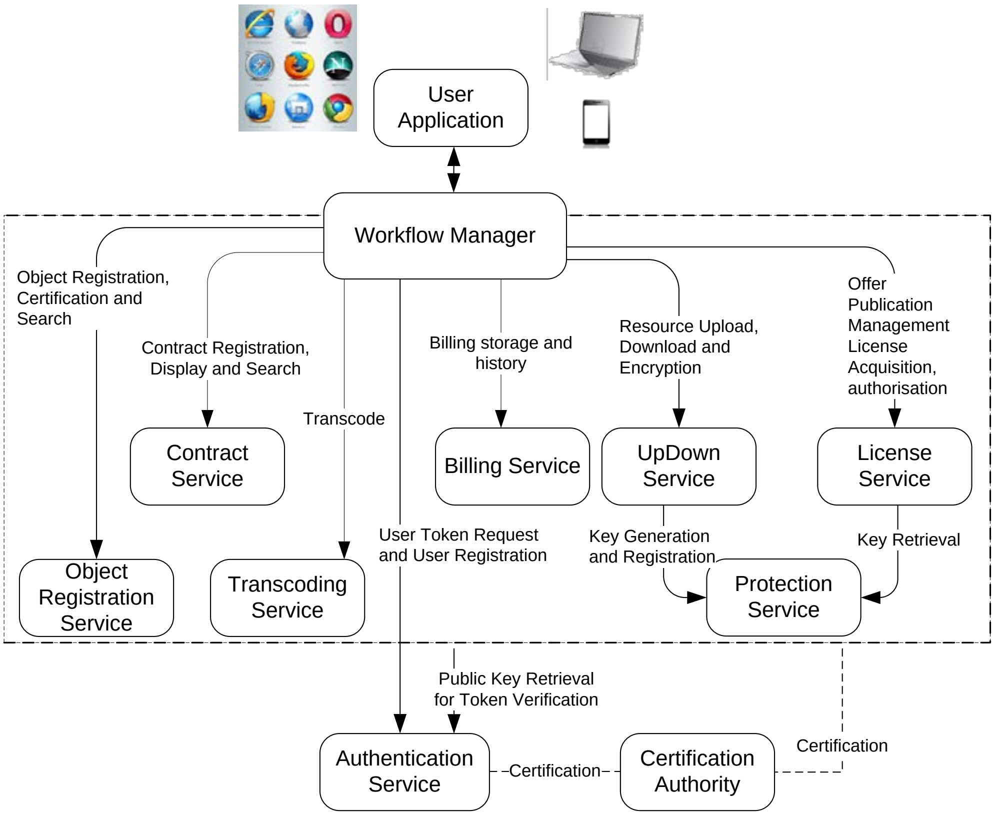
User Application
Workflow Manager
Object Registration, Certification and Search
Contract Registration, Display and Search
Transcode
Billing storage and history
Resource Upload, Download and Encryption
Offer Publication Management License Acquisition, authorisation
Contract Service
Billing Service
UpDown Service
License Service
User Token Request and User Registration
Key Generation and Registration
Key Retrieval
Object Registration Service
Transcoding Service
Protection Service
Public Key Retrieval for Token Verification
Certification
Authentication Service
Certification
Certification Authority

Figure 2. mediaTG media Platform (mtPlatform)

mtPlatform Services

The Authentication Service (ATS) is needed to authenticate the identity of users. It generates SAML (Security Assertion Markup Language) based tokens. Any service in the MIPAMS architecture will require a token argument to be provided in order to authenticate users. Tokens are digitally signed, so that they can be checked for authenticity and integrity by the receiving service. Moreover, the ATS deals with user registration and management (i.e. personal data modification, user account deactivation, etc.).

The Billing Service (BS) deals with the payment operations originated by the users' operations performed in the platform. There are different kinds of payment information supported by this module, as it is differentiated the payment for the service from the payment for the purchases done (licenses purchased). To do so, different billing conditions are taken into account. The service provides operations for service providers (sellers) like retrieval and generation of bills according to different criteria, like date range or specific user. It also offers operations to service clients (purchasers or buyers) for checking consumption done according to different parameters, like number of purchases or date range.

The Certification Authority (CA), which issues credentials for the different Components and Actors in the system, as X.509 certificates and private keys for the different architectural components.

The Contract Service (CS) manages contracts expressed in MPEG-21 CEL (Contracts Expression Language). It provides different search operations over the contracts stored in the system. The main objective of these operations is to help contracts' parties in the management of their contracts, providing contract search by country, rights issued, exclusivity, etc.

The License Service (LS) deals with all operations related to licenses and offers: creation, search, authorization of user operations based on licenses she owns and reporting of authorizations performed. Licenses are expressed using MPEG-21 REL, guaranteeing standards support. The reason for this has been a redesign of the service in order to treat licenses and offers as a unique resource, providing into only one service the operations related to them.

The Object Registration Service (ORS) enables applications to request a digital representation of content and metadata (i.e. digital objects) to be generated and registered in the system. It also provides operations for searching objects and reporting of new object creation. Content and metadata are packaged together following the MPEG-21 Digital Item approach.

The Protection Service (PS), as introduced before, generates encryption keys upon request, registers encryption keys associated to uniquely identified content and provides the encryption keys for protected content to the AS.

The Transcoding Service (TS) is used to convert video files into different formats for its visualization into different platforms and devices. It is connected with the UpDown service, as the current transcoding occurs when the file is uploaded into the system. In this precise moment, several versions of the uploaded video files are created, including one version for video streaming in a lower quality than the original file.

The UpDown Service (CS) enables applications to upload and download digital resources such as audio or video files, text documents, etc. There are several versions of this service, depending on the needs of the application using it and the storage options required. Currently, this service has an FTP version, an HTTP version and an Amazon Web Services version. Resources can be optionally encrypted under request, asking for protection keys to the Protection Services (PS), described next.

MIPAMS vs. mtPlatform

This section provides an exhaustive comparison between MIPAMS and mtPlatform, highlighting the differences between them. There are several aspects in which mtPlatform has improved compared with MIPAMS v2.1. This section is intended to explain the differences regarding the design of both versions, focusing on the advantages mtPlatform offers.

Module comparison

In order to compare the two platforms, we provide a module by module comparison between the two platforms. Even when they provide similar functionality, MIPAMS services work with SOAP protocol and mtPlatform services are based on REST.

MIPAMS module	**mtPlatform module**	**Comparison**
Authentication Service	Authentication Service	Same functionality
Authorization Service	Not present	Module not needed in mtPlatform
Not Present	Billing Service	New service only available in mtPlatform
Certification Authority	Certification Authority	Same functionality
Content Service (CS)	UpDown Service (US)	CS was a socket based application, whilst US has different versions depending on the requirements of the final application
Not Present	Contract Service	New service only available in mtPlatform
License Service	License Service	mtPlatform version of this module provides a complete set of operations over licenses and offers (creation, search, authorization and reporting)
Object Registration Service	Object Registration Service	mtPlatform version of this module provides a complete set of operations over digital objects (creation, search and reporting)
Protection Service	Protection Service	Same functionality
Reporting Service	Not Present	Module not needed in mtPlatform
Search Service	Not Present	Module not needed in mtPlatform
Not Present	Transcoding Service	New service only available in mtPlatform
User Application	User Application	Same functionality
Workflow Manager	Workflow Manager	Same functionality

As a result of this comparison, we can say that mtPlatform provides more functionality with a smaller number of modules, as the new ones, provide completely new features to the platform.

Design Principles

Development environment: Moving from C to Java

The development environments used for each platform are very different, as MIPAMS v2.1 is implemented using C++ and mtPlatform is implemented with Java. The rest of the section describes some development details.

On the one hand, MIPAMS v2.1 implements SOAP-based web services (SOAP, Simple Object Access Protocol), using the gSOAP library (http://www.cs.fsu.edu/~engelen/soap.html), a C++ library which provides an implementation of the SOAP protocol including creation and parsing of XML-based (XML, eXtensible Markup Language) SOAP requests and responses. They are currently implemented using Microsoft Visual Studio IDE and run over Windows Operating Systems (OS). It implies the executable files generated can only be executed on Windows servers, introducing an important restriction. Moreover, changing or adding new functionalities has a direct impact on the development time, increasing it due to the difficulty of maintenance of the current code. The current version of these services is running over Windows 7.0 OS.

On the other hand, mtPlatform is developed in Java with the Eclipse IDE, which solves some of the implementation issues described for MIPAMS v2.1. mtPlatform implements REST-based web services using the Jersey (version 14) Java library (http://jersey.java.net/), which implements REST services. The use of Java as development language provides several advantages in front of C or C++. Java is a portable language and services can be executed both in Windows and Linux operating systems, lowering hardware costs as the platform can be hosted in a Linux server which is cheaper than a Windows one. The new design improves the code development due to its high changeability and addition of modules.

SOAP vs. Rest

In order to compare the two platforms, we provide a summary of the differences between SOAP and REST.

REST web services directly map their operations to HTTP (HyperText Transfer Protocol) methods (GET, POST, PUT, DELETE), using HTTP request – response mechanism and are based on resources. The meaning for each operation is the one defined in the HTTP RFC (http://www.w3.org/Protocols/rfc2616/rfc2616.html). GET is used for retrieving a resource, POST can be used to perform different kinds of operations over a resource (create, modify, even delete), PUT is used for storing a resource and DELETE is used for removing a resource.

Using such REST provides several advantages in front of SOAP-based web services, which can be summarized as follows:

- It is not necessary to use SOAP request and response XML-based messages, which reduces the information sent during web service calls. In contrast, different kinds of information can be used: text, XML or JSON (Javascript Object Notation).

- It is HTTP based, so it is a protocol widely supported and can go through firewalls.
- It is based in resources, not in remote operations. In REST, a URL is used for accessing the resource. In SOAP, a XML message (derived from XML-RPC, Remote Procedure Call) is always required.

MPEG-M

mtPlatform makes use of the new MPEG-M standard in order to ensure interoperability with other platforms implementing this standard. Part 4 of the standard is focused on defining service interfaces – called Elementary Services – to be used independently of the specific system architecture of the platform. These Elementary Services are combined into Aggregated Services, as defined in Part 5 of the standard, to create complex service chains with the objective of implementing new use cases depending on the platform needs.

When a MPEG-M request is sent to mtPlatform, a translation mechanism is enabled in order to generate the proper mtPlatform request. The query, after being processed, is translated again into MPEG-M format and sent back to the requester. By extending this MPEG-M layer located between the user application and the mtPlatform services is possible to increase the compatibility with many MPEG-M Elementary Services on demand, ensuring interoperability with other MPEG-M platforms.