Deliverable 19.0.1

External and Internal Models and Protocols for the PrestoSpace Factory

DOCUMENT IDENTIFIER	PS_WP19_RAI_D19.0.1_Ext_Int_Models_PF_v1.1
DATE	2008-02-25
ABSTRACT	The document describes and analyses the models representing how the PrestoSpace Factory may provide its services to the Archives and the models and protocols defined for interoperability among the PrestoSpace subsystems. Within a high level scenario description, the basic archive business process model and the overview of the PrestoSpace Factory architecture are given. The role of the PrestoSpace Orchestrator component is specifically addressed. The models of transactions between the Archives and the Factory are analysed. The offered services, the exchange mechanisms, and the PrestoSpace units and interfaces are described.
AUTHOR, COMPANY	Laurent Boch (Rai – Radiotelevisione Italiana)
NATURE	Report
DISSEMINATION	Public
KEYWORDS	archive management, archive inventory, archive exploitation, editorial objects, audio visual material sources, essence and metadata storage, transaction, orchestration, preservation, restoration, documentation, services, orders, batches, delivery options, working options, essence and metadata formats, system architecture, models and protocols, system interactions.
RELATED ITEMS	storage architecture, system integration, preservation, restoration, documentation MAD area Business Process model.

DOCUMENT HISTORY

Release	Date	Reason of change	Status	Distribution
0.0	2006-01-05	Draft	Living	Author
0.5	2006-12-22	Draft	Living	http://prestowspace.ina.fr/groups/MAD/maddeliverables/
0.6	2007-01-08	Draft	Living	http://prestowspace.ina.fr/groups/MAD/maddeliverables/
0.8	2007-01-12	Advanced draft	Living	http://prestowspace.ina.fr/groups/MAD/maddeliverables/
0.9	2007-01-16	Advanced draft	Living	http://prestowspace.ina.fr/groups/MAD/maddeliverables/
0.10	2007-01-26	Advanced draft	Living	http://prestowspace.ina.fr/groups/MAD/maddeliverables/
0.11	2007-05-05	Approved within 3rd Review Report	Approved	Confidential
1.0	2007-10-23	Finalised	Final	Confidential
1.1	2008-02-25	Amended references and annexes, made public	Final	Public

Table of contents

1 Scope

The scope of the present PrestoSpace deliverable is the scenario of transactions which have to take place between the PrestoSpace Factory and its institutional customers, namely the audio-visual Archives, and the models and the protocols defined or adopted for interoperability among the PrestoSpace Subsystems.

The PrestoSpace Factory is defined to be a complex set of facilities which are able to provide massive analog-to-digital migration of audio-visual contents, digital restoration, and the documentation and metadata enhancement, obtaining also by means of automatic features extraction, which are going to permit the best exploitation of the archive contents.

The assets of the Archive are therefore the object of the PrestoSpace Factory processes and the accomplishment of the work is achieved with the final delivery to the Archives of new digitised Master Quality Materials – and/or any other requested quality level such as broadcast or browsing - together with the set of identification, description, and indexing information which make up the structure for information and material retrieval within the process of publication of archive contents.

This document is the result of merging the two deliverables D3.1 and D16.4 into a unified reference for all the exchanges dealing with a central orchestration system, within the PrestoSpace Factory, responsible for interfacing towards the archive and for providing the overall work-flow management.

The architecture of the PrestoSpace Factory, with the identification of the main component sub-systems and of the interfaces through which the main interactions occur, is the background of the whole discussion.

2 Executive summary

The PrestoSpace Factory is a set of facilities providing various important services to audiovisual Archives and therefore the models that conduct the relationships of that Factory with its customers, the Archives, and govern the interactions among its components are based on the same scenario on which the Archive activity is based.

The archives operate into a scenario where their assets can be of interest for Content Creators, Content Distributors as well as for the general audience, and where the role of all those actors, including the Archives, is evolving according to the innovations in the technical domain and in their business models.

Specifically the Archive business process model is founded around the Archive Exploitation, which is made of use and fruition of both audiovisual items and/or the related information. Moreover effectual exploitation often represents the way of funding the Archive activities (or claim funding to the government for them). In order to actually achieve that goal several other tasks must be accomplished, the first of which is to have the complete list of assets (inventory) from both the editorial and material points of view, together with storing securely the audiovisual material sources at the highest required quality (preservation / digitisation / restoration) and collecting and enriching the related information (documentation) in order to permit exploitation through search and retrieval activities.

The Architecture of the PrestoSpace Factory is conceived to serve the Archive process model in order to fill (totally or partially) the gap between the current archive situation and that where a good exploitation is possible.

A specific component, the PrestoSpace Orchestrator (PSO), is the pivot of the whole factory, providing the interfaces for both the Archives and the other factory subsystems, acting as the overall work flow manager and controlling the common Essence and Metadata Storage (EMS).

The other components, Units are defined to respond to more specific tasks, the main of which are Preservation, Restoration and Documentation.

The classification of the possible customers according to dimensions, status of assets, organisation, and future planning, provides a variety of cases that can be faced only with negotiation and flexibility.

This fact impacts on the models of transactions between Archive and PrestoSpace Factory, the analysis on which is presented according to the exchanged entity.

The transfer of original media is particularly critical, especially if the Digitisation Unit is located off site, and implies the issue of tracking the item position and that of item identification. The transaction models for essence (digital material sources) are mainly related to the issue of infrastructures for storage and transfer of large amount digital data. The adopted approach for metadata is oriented to keep separate transactions for essence and metadata, relying on the capability of keeping updated the inventory information with all the links between editorial and material items, in order to obtain a greater flexibility for delivery. As the information is generally related to the editorial concepts, the Editorial Object was identified as the entity being the main working unit within the factory and the Editorial Object Document (EOD) is the handle for accessing any related information.

The relationship between the Archive and the PrestoSpace Factory is based on the concept that large amounts of items will be associated to the same defined process (Order) and that the whole activity of submitting items, processing within the Factory, and delivering to the Archive, will be organised into Batches of working units, grouped according to various criteria such as physical transfers, delivery dates, editorial priorities, or logistical questions.

The defined process is basically made of a selection of services, which can be customised by setting the relevant offered options, completed with the selection of other communication details, such as request of events notification, and the supply of estimate figures allowing the factory to best allocate resources along time.

The PrestoSpace Orchestrator will have the responsibility of rendering the requested process into a sequence of tasks that will be assigned to the various units, including the solution of possible dependencies.

The interaction between systems are supported by the use of web-services, most of them provided by PSO, the exchange of defined XML documents, and the use of standard protocols for essence exchange and access.

The management of Editorial Object documents and the exchange and delivery mechanisms for working instructions, materials sources, notifications, and knowledge base repositories are defined and there references to XML Schemas and WSDL specifications are provided.

3 References

[BPM] MAD Area business process model - MAD Area Analysis of Business Process Model MAD Area Analysis of Business Process Model v0.4 – *L. Boch, A. Messina - 2005-07-04.* This a non public PrestoSpace working document, the most relevant outcomes of which are collected into D15.2.

[PMETA] Tech 3295 - P_META Metadata Exchange Scheme - EBU Tech 3295 (v1.2) (notes) versions 1.x – *EBU - 2002-2005.* This has been currently superseded by EBU-T3295v2

[EBU-T3295v2] Metadata Library P_META 2.0 - ebu-tec-doc-t3295v2 - *EBU P/MAG. July 2007*

[PSWP] PrestoSpace project web home page - http://www.prestospace.org/ - *2006-05-31*

[WKPD] wikipedia - http://www.wikipedia.org - *2006-08-23*

[D12.4] A Strategy and Architecture for Storage - D12.4 Storage Tech Architecture– *Richard Wright (BBC) – 2006-03-02*

[D15.1] Analysis of current audiovisual documentation models - D15.1 MDS1 – version 1.6 – *Christoph Bauer, Florian Rosensprung, Stefan Lajtos (ORF), Laurent Boch (RAI), Philippe Poncin (INA), Conny Herben-Leffring (B&G) – 2005-03-16*

[D16.4] Delivery models – D16.4 Delivery Models - version 1.0.1 – *Laurent Boch (RAI) – 2006-03-15*

[D3.1] Models and Protocols for PrestoSpace Factory Process - D3.1 SAS1 - version 2.9.13 – *Jean-Hughes Chenot (INA), Andrea Bernicchia (Reply), Peter Schallauer (JRS), Laurent Boch (RAI), Walter Allasia (Eurix) – 2006-03-02*

[D15.2] PrestoSpace Documentation Platform - D15.2 - version 1.0 – *Giorgio Dimino (RAI), Laurent Boch (RAI), Alberto Messina (RAI), Werner Bailer (JRS), Christoph Bauer (ORF), Valentin Tablan (Un. Sheffield)*

[D19.1] Results of Integration - D19.1 - *W. Allasia, A. Damiani, M. Porzio, S. Ridolfi, F. Toscano, M. Vigilante – 2008-01-08*

[XMAD] MAD XML Schema – Annex of D15.2

[WPSOA] PSO-Archive WebServices WSDL – Annex of D19.1

[WPSOU] PSO-Units WebServices WSDL – Annex of D19.1

[WPRE] Preservation WebServices WSDL – Annex of D19.1

[WDOC] Documentation WebServices WSDL – Annex of D19.1

4 High level scenario description

4.1 Criteria of work

A basic reference of this deliverable is given the objectives defined for whole PrestoSpace Project, [PSWP]:

> "*The project's objective is to provide technical solutions and integrated systems for digital preservation of all types of audiovisual collections.*
>
> *The project intends to provide tangible results in the domain of preservation, restoration, storage and archive management, content description, delivery and access. Economic factors supporting preservation services will be addressed.*
>
> *The principal aim is to prepare the way for preservation factories providing affordable services to all kinds of collection custodians in order to manage and to allow access to their assets."*

This analysis about the high level transaction models of the PrestoSpace Factory has been carried out so as to identify the distinguishing elements of the varied possible operating scenarios and to understand the implications towards the possible delivery models.

It has been considered particularly important the reference to the basic business process model of an audiovisual Archive, because the role of the Factory is, beyond the technical solutions for digital preservation, to enhance the effectiveness of the Archive mission itself.

Also it has been given importance to the identification of the business actors, that are those entities playing a specific role within a relationships.

To this purpose, it is useful to observe that some services are going to be used in order to fill the gap which prevents the Archives to make the most of their collection, while other services aim to assist the Archive exploitation in a continuous way. In the former case it is easy to identify the Factory and the Archive as the two relevant parts in the transaction relationships, while for the latter the offered services work on behalf of the Archive towards the Archive Users.

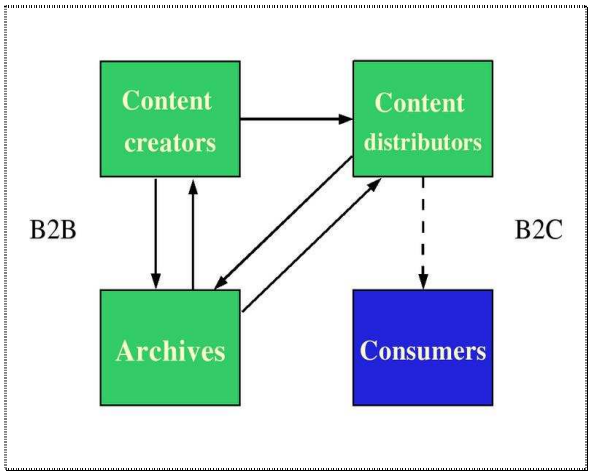
A preliminary classification of the Archives, and of their situations, is the pivotal activity for the analysis of the transaction models and several dimensions are going to be interested:

- Archive dimension – Roughly the amount of items, physical media or hours of content. Although the evaluation of this component can vary considerably according to the expectations of future exploitation.
- Status of assets, physical – This gives the weight of the preservation & restoration components of the Factory required services.
- Status of assets, information – Generally the quality of the legacy information.
- Archive organisation – Is the archive a standalone organisation or rather a department providing archive services to a wider organisation? Is the current structure planned to be changed? Which are the skills of the work force?
- Infrastructure & logistics – Which are the current and possible resources in terms of locations and equipments?
- Archive Users – Who are the current users and who are going to be in the future?
- Budget & Founds – Which kind of investments are contemplated? In which timescale?

It is easy to grasp that all the mentioned components are inter-related, as it hasn't been identified a unique solution for digital audiovisual archives.

4.2 Basic archive business process model

The scenario where the audiovisual archives operate includes basically three other kinds of actors, as found in [PMETA] and reported in Figure 1.



Producers (Content creators) - These are involved in the Production of programmes and other content and make new material available for publication.

Archives - These are involved in the preservation of existing material, that can be re-used as originally produced, or to provide input material for new programmes.

Distributors - These are involved in the publication and delivery of material to the end users. They are the ones who have interchanges in the business-to-consumer scenario. They are also known as Broadcasters or Content Aggregators.

Consumers - These are the beneficiaries at the end of the media supply chain.

Figure 1 - P_META Process Model

The P_META model was depicted in order to define the operating context of business-to-business information exchange about audiovisual assets. In that model the ***Archive Users*** are implicitly defined to be either Content Creators or Content Distributors, hence excluding the end users. However the model identifies roles rather than organisations and actually most main broadcaster companies cover, totally or partially, all of the three business domains.

Nowadays the Archives organisations are considering qualifying more precisely their users, in order to find the resources necessary to feed the processing loop consisting of preservation and exploitation.

4.2.1 The archive management processes

Within the work area MAD (Metadata Access and Delivery) of the PrestoSpace Project an analysis of the business process model was carried out for that domain, by producing and commenting a set of use cases diagrams, and is available in [BPM].

According to the PrestoSpace objectives the audiovisual archive management is described from the point of view of the digital archive and therefore it includes processes, such as Digitisation and Restoration, which are going to be provided by the PS Factory services. The relevant diagram is given in Figure 2

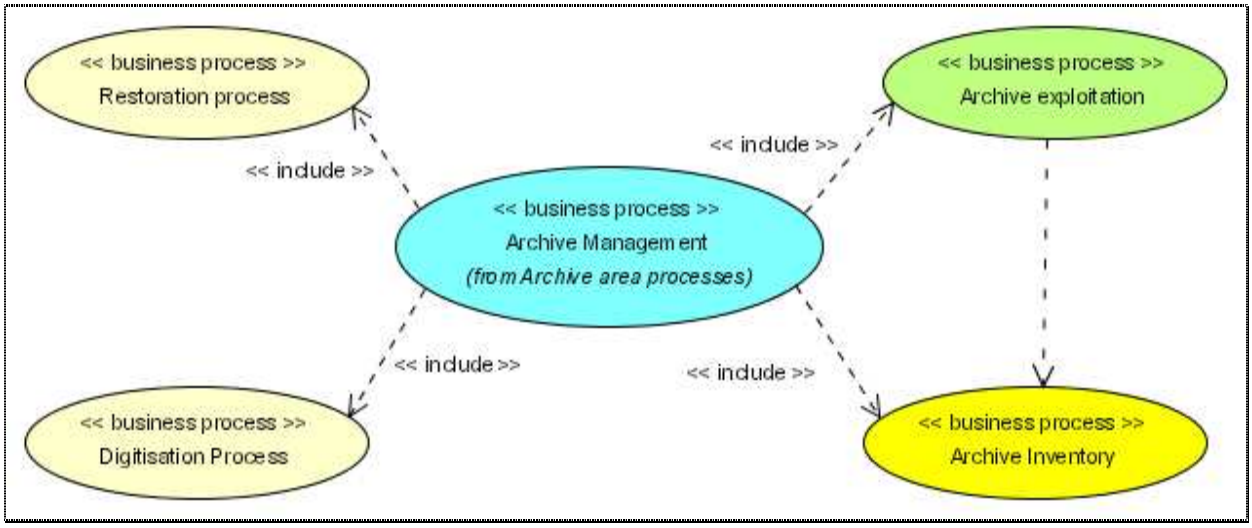


Figure 2 - Archive management processes

The Archive Exploitation process is the objective to which the other sub-processes are finalised. However, in order to actually use the archive content, an Archive Inventory process is necessary, as pointed out by the explicit dependency relation in the diagram. The inventory process is here intended as the ***way to obtain and maintain the information about which are the owned editorial assets and how they are physically realised by the audiovisual items stored in the physical archive***.

Archive inventory process

The inventory process, showed in Figure 3, includes as sub-cases the inventory of Editorial Objects, Editorial Collections, and Material Sources, and it is accomplished through the association between Editorial Objects and their realising materials.

As pointed out in the referenced document, that association requires mutual identification and timelines mapping, a crucial task, the difficulty of which may vary from quite simple to quite complex, and that in practice has been accomplished for the legacy archives with various degrees of accuracy.

An audiovisual archive system (or another system acting on behalf of an archive) can be considered fully inventoried if:

- There is knowledge about the whole set of relevant editorial assets (owned or at least for which some rights are owned)
- There is knowledge about the whole set of physical items (either files or tapes) of which the persistency and safeguard has to be provided
- There is knowledge about how to produce the realisations of the editorial assets from the items physically stored. This includes the knowledge about the editorial object – media timeline mappings.
- For each physical item there is knowledge about which are the related editorial assets that each item contributes to realise.

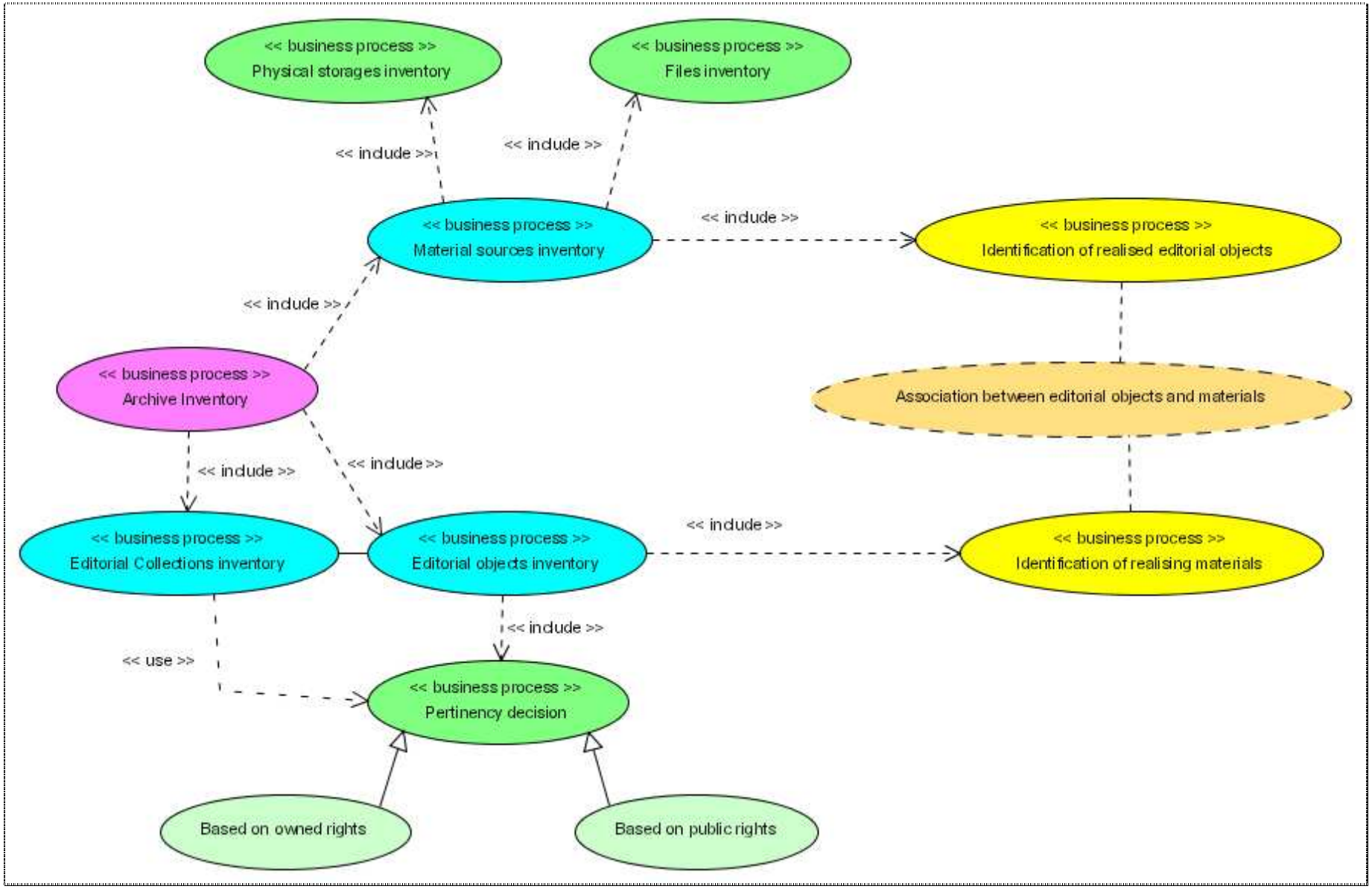


Figure 3 - Archive inventory process

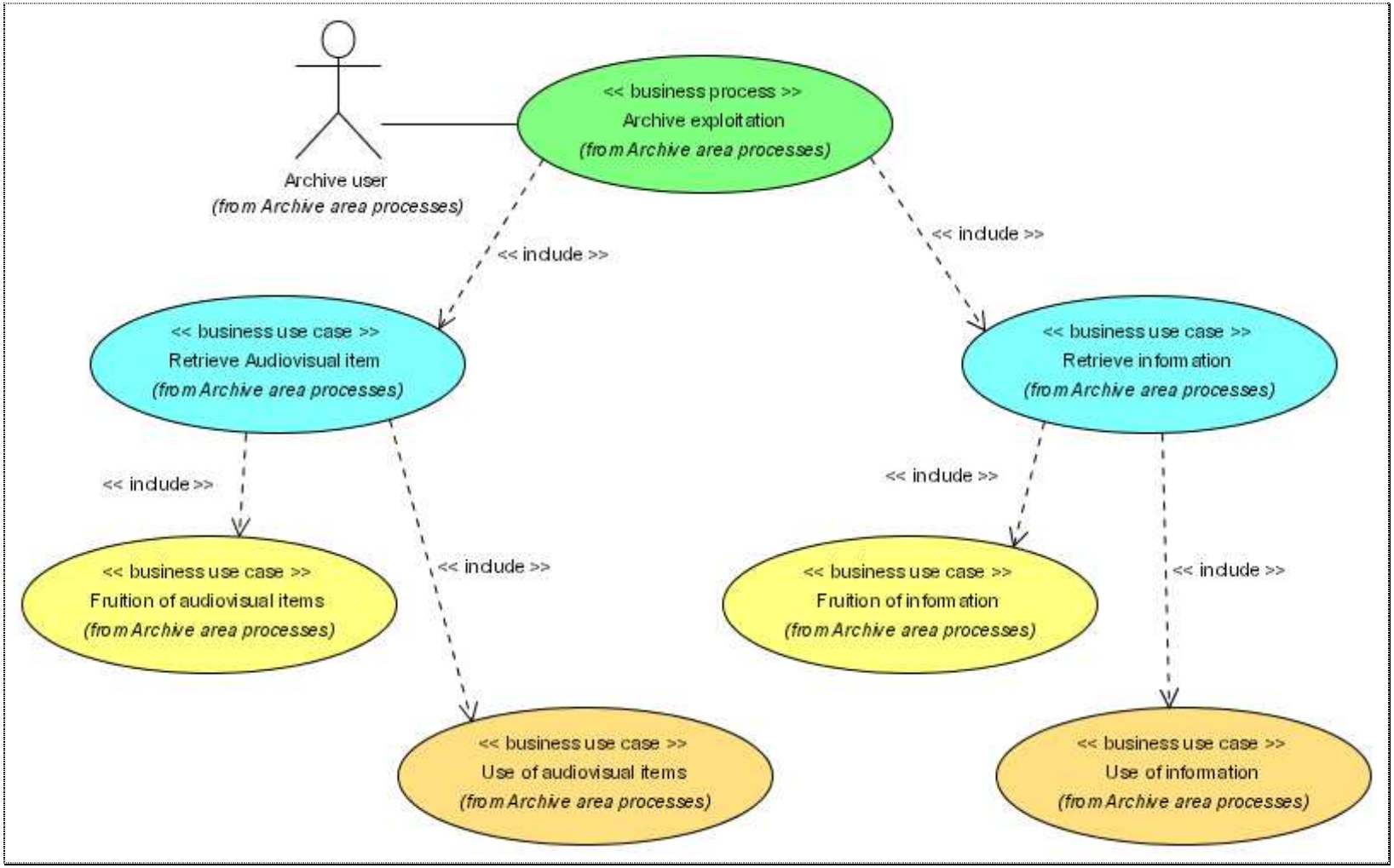


Figure 4 - Archive exploitation process

Archive exploitation

A further detailed diagram for archive exploitation is given in Figure 4. The main objective of the archive user is retrieving either the audiovisual items stored in the archive or the information thereby located in the same place.

The latter gives account of the cases in which archive users are interested in obtaining pieces of knowledge without being directly interested in the audiovisual material somehow connected with it. For example knowing whether certain events happened or not, knowing about the content of a public speech of a famous person, and more in general making reports of certain aspects of social, arts, costume. From this point of view, the information that is living in the archive becomes itself an asset for the organisation owning the archive.

Both the retrieval functions can be distinguished in use and fruition. The use consists in the actual employment and entails exportation from the archive, while the fruition is limited to accessing information and material from the user interface by previewing, browsing, and reading.

4.2.2 Support to archive inventory

The entity which is the fulcrum of interest is the ***Editorial Object***, basically *what comprises the expressive, artistic, and communicable aspects of an audiovisual work* (or an editorial constituent part of it), which is *realised* by at least one instance of audiovisual ***Material***, which can be generated by ***Sources*** (or *Material Sources*), consisting of suitable data recorded on files or media[1].

The data model defined within the MAD work area supports the archive inventory.

Figure 5 provides an example showing the relationships among those entities.

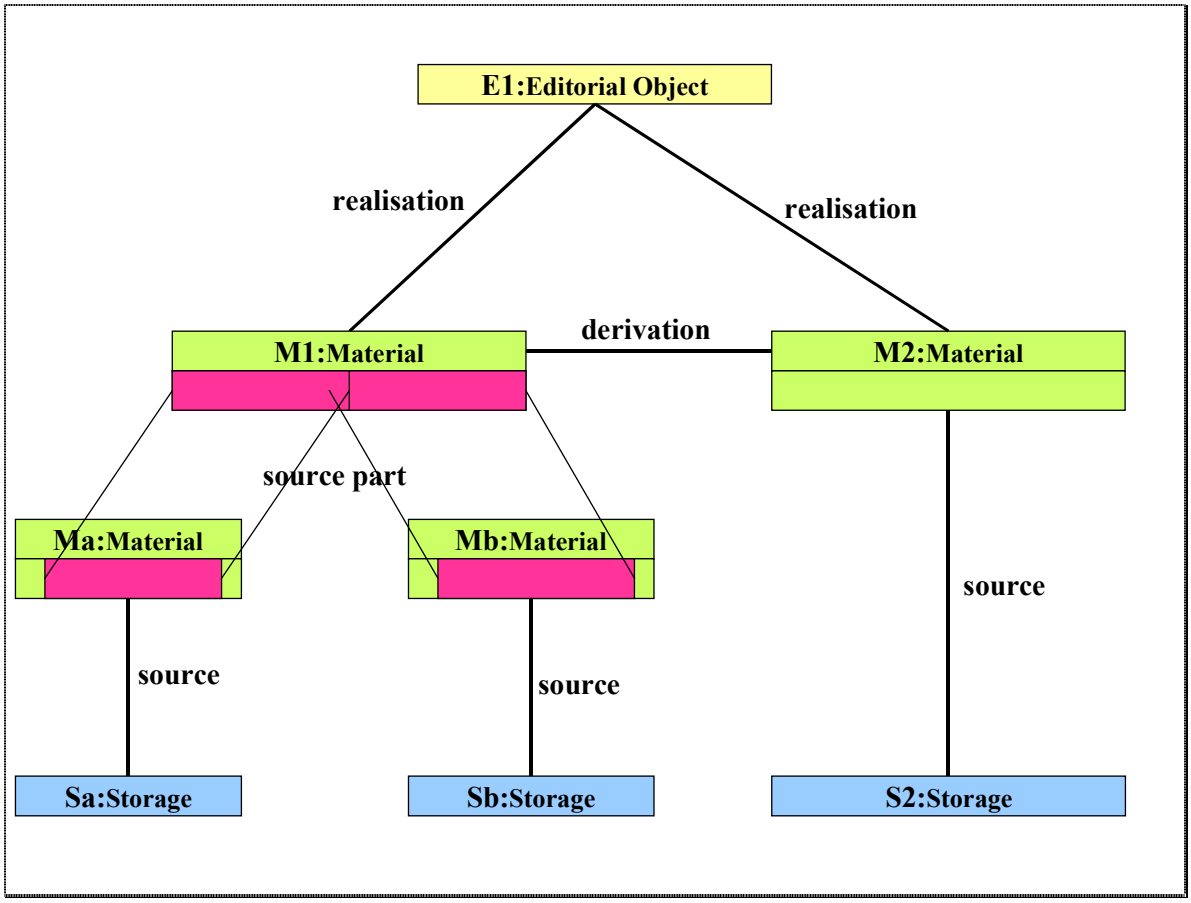


Figure 5 - Objects diagram with Editorial Object, Materials and Storages

[1] At a higher level the model doesn't make much difference between *Files* and *Tapes*, as both are containers for the Material Sources. At a lower level the issue of providing actual *Accesses* to the Sources need clearly to have distinct solutions

It must be also remarked that the complete scenario of mapping between Editorial Objects and Materials include several particular cases, the most typical of which are described in Table 1.

- *Common case* - When a single Tape (or File[2]) gives the Editorial Object realisation. The mapping is completed by giving the start point of the Editorial Object within the Tape Material (a 30 minutes programme can be recorded on a 60 minutes tape).
- *Multi-tape case*. It is not unusual, however, that for recording a single Editorial Object two (or more) Tapes (or Files) have been used. In this case for each contribution to the Editorial Object realisation a complex mapping record (duration, tape, start on tape, start on editorial object, tracks) is required. Moreover gaps and/or overlaps (more often) are possible[3].
- *Multi-programme case*. Sometimes two (or more) Editorial Objects are recorded on the same Tape (or File). Also in this case the mapping records are required.

Table 1 – Cases of mapping Editorial Objects vs. Material

When the inventory process within the Archive is poor or weak, it should be completed as described by Figure 6 or the exploitation of some archive assets might be endangered.

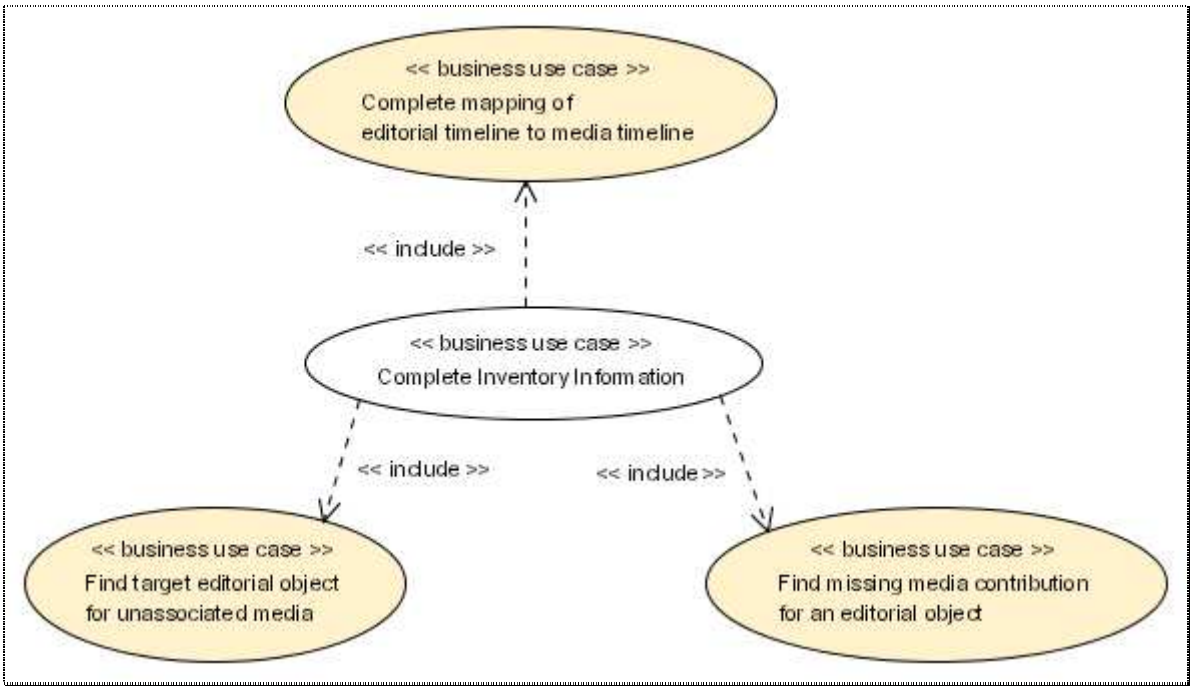


Figure 6 – Complete inventory information process

Finding missing media contributions and finding Editorial Object for un-associated media can be difficult tasks, as the material could be completely unknown or simply physically lost and the puzzle might eventually result incomplete.

When the problem is limited to completing the mapping of editorial timeline to media timelines, an accurate solution is logically achievable. However it is reasonable that it would be more concretely accomplished once that the digitisation process is completed.

[2] The same description apply to File material sources. Often an audio/video file is the product of the digitisation of a whole tape.

[3] This case is generally more complex if we consider multiple copies for the same Editorial Object exist.

4.3 The PrestoSpace Factory

The PrestoSpace Project relies on an assumption that, to allow the AudioVisual Archives to progress towards preservation and making accessible their collections, they have to use an industrial approach to massive analogue-to-digital migration plans. The User Requirements phase has demonstrated that this message was well received in most Archives. Even in the specific case of Film Archives, who reserve the 'Preservation' term to long-term storage of physical artefacts, they admit they have an increasing requirement for making their contents accessible, and are interested in an industrial approach for this.

PrestoSpace technical and non-technical developments tend towards an industrial approach of migration, and the Technical Deliverables are expected to facilitate this industrial approach.

The scale of PrestoSpace Factories, is not fully defined yet. It is extremely likely that factories of different scales will coexist, depending on the size of the collections, type and condition of the media, on the results of the technical developments, and on factors such as the difference in labour costs between the different countries. Shipment costs should also be considered. The scales considered up to now span from movable 2-rack units for on-site migration of cassettes, to very large-scale plants, passing by mobile truck or medium-scale facilities.

The actions to be performed in the PrestoSpace Factories have been broadly split into three main categories :

- Preservation (migration)
- Restoration
- Documentation and Access

Of course, there are overlaps between these actions. The user requirements phase have demonstrated an extreme variation between the different requirements, scales, and urgencies for the three functions. The Preservation effort is clearly driven in most cases by urgency considerations, as the media are often deteriorating at a fast pace, playback machines and expertise harder and harder to maintain. Requirements for urgent Restoration and Documentation are variable from one archive to another.

In this document, the following assumption will be made :

A PrestoSpace Factory is a facility where one or several of the functions above, (Preservation, Restoration, Documentation), are performed. These functions are performed in different Units, which are :

- The Preservation Unit
- The Restoration Unit
- The Documentation Unit

Although in some cases all the Units may not be present within the PrestoSpace Factory, it will be assumed here that all three coexist. Given the very different workflows between Units, the interfaces with the outside world and between the Units have to be defined precisely, as it will limit the consequences of changes within each Unit. The results of this definition activity is given in §6.3 and §6.4.

4.3.1 System architecture overview

The main entities of the system architecture are depicted in Figure 7.

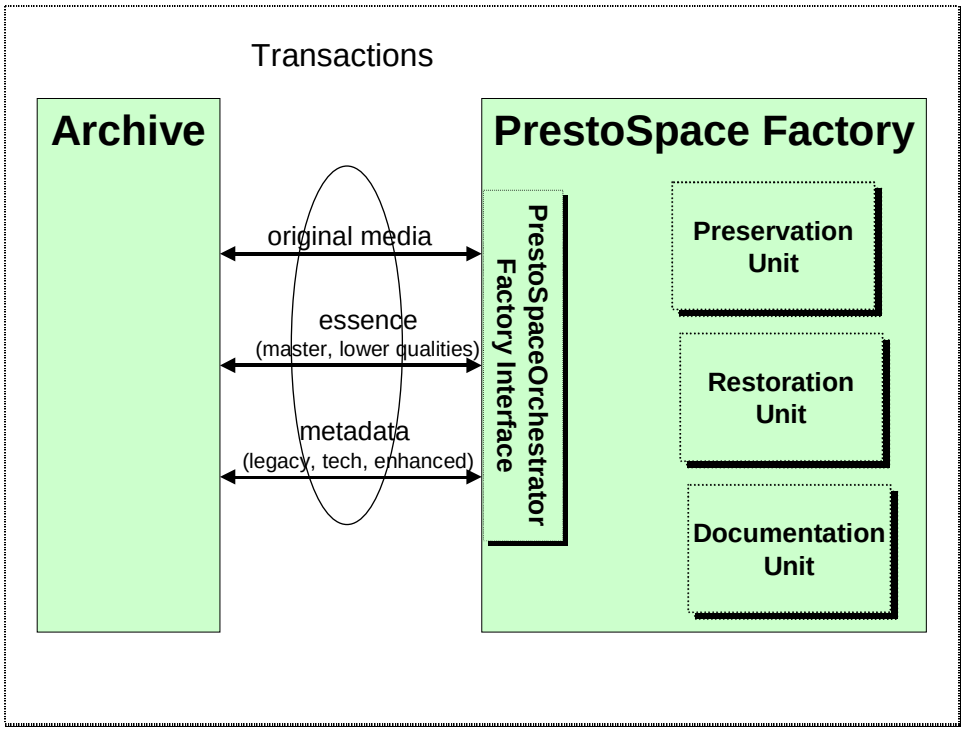


Figure 7 - System architecture overview

PrestoSpace Orchestrator

The PrestoSpace Factory has got a specific component, named *PrestoSpace Orchestrator (PSO)*, for interfacing with its customers, the Archives. All the exchanges between the Archives and the PS Factory will be managed through the PSO, that will offer interfaces (e.g. web interface) for input/output, for the verification of the progresses of the batches, and for implementing the notification mechanism for the required cases (such as completion of batch, major problem or other defined events). These interfaces will be accessible by a human or a computerised system.

Although the setting of the requested services from PrestoSpace Factory will be possible through the PSO interface, it must be reminded that in many cases it will be more effective to define first a rough proposal of deployment of services with the help of a "*Negotiator*", able to inquire, together with the Archive, about both the current and target Archive situation, also in order to discuss and agree on any other specific detail, such as the way in which the original carriers will be delivered.

Furthermore the PSO will also play the role of the workflow manager within the PrestoSpace Factory and each Unit will have its own communication interface with the PSO for implementing a defined work progress protocol.

A common Essence and Metadata Storage system (EMS), under the control of PSO, will be the preferred mean for exchanging contents and metadata files between the Units. A versioning subsystem will be used for metadata (not for essence).

When applicable, the PSO is also responsible for delivering the contents to a Publication Platform, which may be either attached to the PrestoSpace Factory or incorporated to the Archive. However this is not the scenario described in this document.

Unit roles

The Preservation Unit does the digitisation, and generates (automatically and/or manually) the “Migration Metadata” consisting of operators inputs, system diagnostics, and quality reports.

The Restoration Unit's role is to enhance digitally the quality of the contents (sound and image tracks). It uses the digitised essence (and any metadata relevant to the digitisation process) as input. Its outputs are the restored essence (i.e. derived essence) and/or metadata describing the impairments of the input material.

The Documentation Unit's role is to complement and further enhance the documentation associated with the essence. It uses the Essence and the available metadata as inputs.

Exchanged entities

Original media – The physical removable storages where audio and video were originally recorded. Generally they are in an analogue format, although most considerations may apply also to those digital formats that are not “*File Based*”, e.g. Digital Betacam.

Essence – The digital audio and video material sources. Without mentioning coding (compression) and wrapping formats, we can say that essence is stored and exchanged as binary data. Typically a large amount of data, please make reference to [D12.4] for further details.

Metadata – The information about the audiovisual material and its editorial content. The scope of the information is quite wide, including entity identifications, descriptions, and technical details. They can be stored and exchanged as binary data; the data formats adopted require text-based representation in XML, which can be lossless, compressed for exchange/storage efficiency. The size of data is quite lower than for essence.

4.3.1.1 General work flow

The Archive often runs (implicitly or explicitly) a pre-selection process that will decide on the contents of each migration batch. The analogue media are delivered to the Factory for digitisation and then given back to the Archive.

Any further work within the Factory is performed on the digital media, which is delivered to the Archive together with all the required enhanced information that permit the release for exploitation, as "Packaged Media"[4].

This possible general workflow for the PrestoSpace Factory is given in Figure 8:

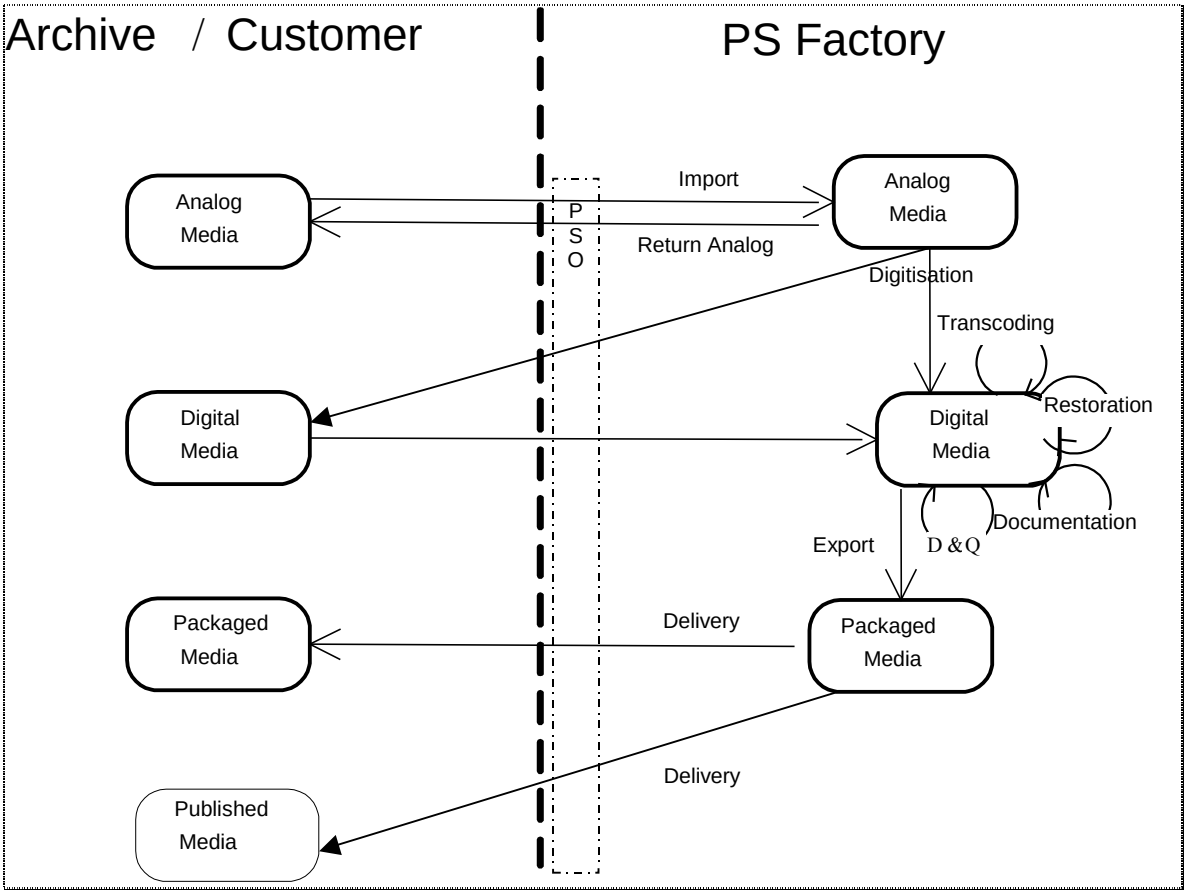


Figure 8 - General workflow for the PrestoSpace Factory

[4] It is not meant here a physical wrapper, but rather the fact the metadata, providing description and access to the digital media, furnish a logical packaging.

4.3.1.2 Overall activity

A possible generalised overall activity diagram is given in Figure 9, which depicts a scenario permitting to enlighten a few key aspects:

- The default activities are *Digitisation/Preservation*, *Update Inventory* and *Delivery*. The other activities provide an enhancement of the output.
- *Update inventory activities*. During the whole process, archive inventory activities can be required whenever new material instances are produced or previously undiscovered editorial objects are identified during documentation (e.g. programmes contained in other programmes), or simply corrections and completion of Editorial Object vs. Material mapping are performed.
- *Roles of restoration processes*. The restoration of material can take place previously to the documentation as well as after it. The first one is indicated as unconditioned restoration, while the last one is indicated as documentation-driven restoration. Particularly in the second case restoration works can be performed on the basis of information deriving from the documentation process (e.g. items requiring particular restoration can be pointed out by documentalists).

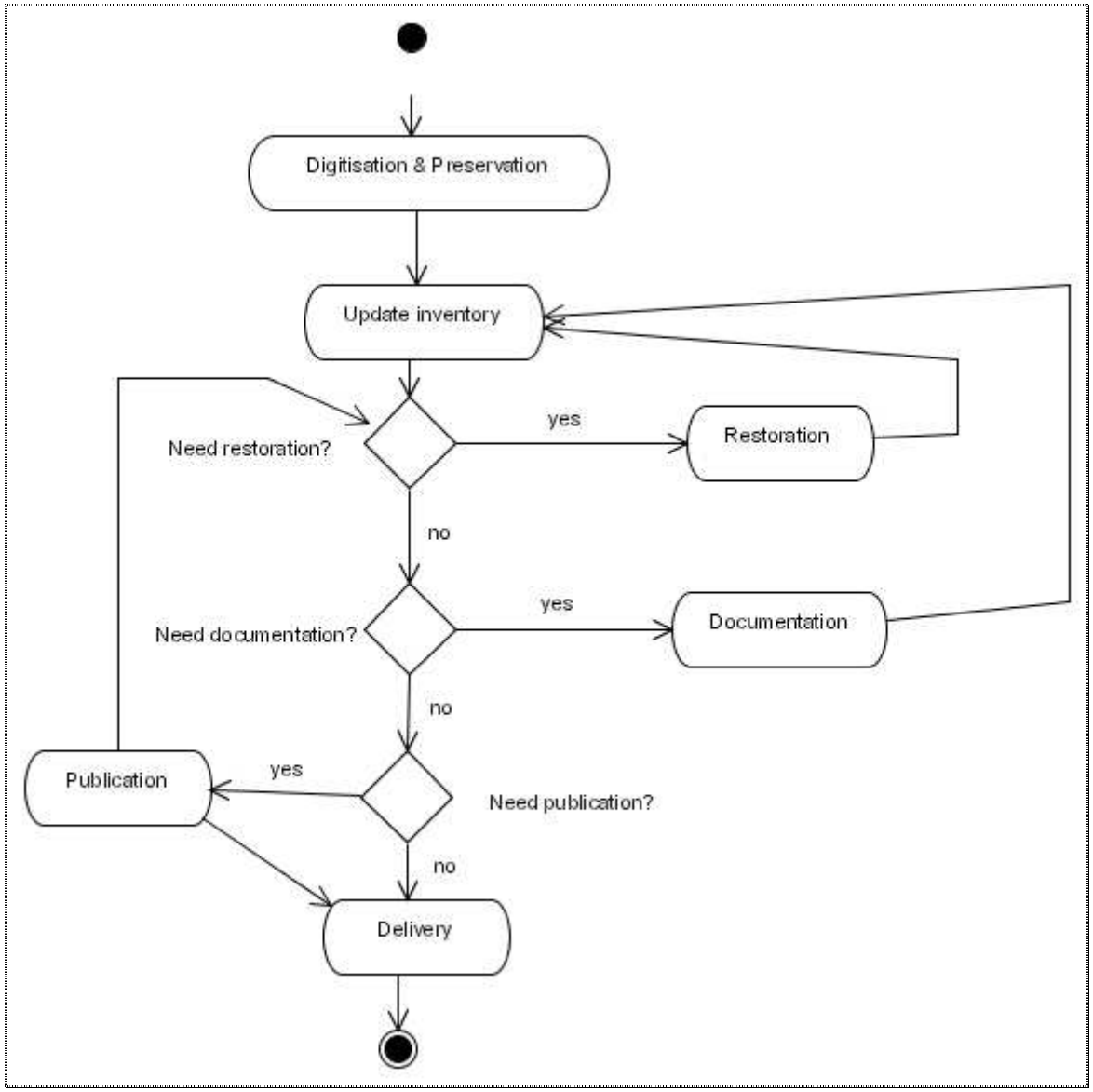


Figure 9 - Generalised overall activity

4.4 Factory customers profiles

As outlined in the “Criteria of work”, there are several components that generate the assortment of the Archives domain from which the factory customers come. A PrestoSpace Negotiator will be certainly interested in Archive dimension, budgets and timescales, however the final question is “how far the current Archive status is far form the possible target?”

4.4.1 Worst case

Archive is more a media repository than a veritable Archive. This occurs when the Archive is not able to run a sufficient management process (Figure 2), in particular concerning the inventory (0). Likely the status of assets is either bad or unknown and the legacy information is very hardly usable. Current archive exploitation is near to nil.

If the size of the “archive” is large and the available budget too, that could be quite a best case, from the factory service provider perspective.

The problem is that, without a starting point good enough, it is difficult to guess which could be the future exploitation possibilities.

A suggestion is to try to define a strategy in order to fall into a better profile, attempting to understand whether it is better to work at preservation without having a minimal inventory or to ask to the archive to do some housekeeping before.

4.4.2 Modernisation case

The profile described here is about Archives aiming to undertake a significant modernisation.

These Archives are currently working, i.e. they have some Inventory and Exploitation processes, but they rely on legacy tools, especially concerning the management of information, which prevent significant enhancement. Often their organisations have already progressed in the information technology, but the Archive itself hasn’t be touched too much because of the complexity in dealing with the legacy systems.

Usually what triggers the modernisation initiative is the need to digitise the analogue media, however there is also the will to obtain substantial improvements with new features, such as multimedia cataloguing, and to fix malfunctions due to old inconsistencies and defective patches.

Of course a wide range of different sub cases fall under this profile. Anyhow the available information permits to estimate the benefits of the increase of exploitation, constituting thus a source of funding for the process.

Being the legacy system old enough, these Archives are prepared to change them almost completely.

4.4.3 Enlargement case

Eventually we have the case of those Archives that achieved a good level of modernisation within the last ten years. Their Archive Management Processes might not rely on state of the art tools, however their systems permit very good performances and their exploitation figures give the evidence of the achieved improvements.

These Archives are in a good situation, but they understand they can get even better. The Archive users, after having taken advantage from the previous modernisation, are full of new expectations. New kinds of users are envisaged.

Likely, the digitisation is not complete, or even late, what an opportunity to also consider new process models and profit of the features provided by the PS Factory services.

These Archives are prepared to change something in their organisation and systems, where they identified bottlenecks, but they particularly require a greater level of integration with their existing resources.

5 Archive vs. PrestoSpace Factory transaction models

5.1 Considered dimensions

The architecture overview given in Figure 7 tells which entities are involved in transactions between the Archives and the PS Factory. Those transactions need to be supported by services which will have to follow defined protocols, use defined formats, and require a wide set of accompanying information which will specify all the required details and working directives (options).

The relevant main transaction models have been analysed according to following dimensions:

- *Space* – Where the PrestoSpace Factory is located regard to the Archive location. The answer impacts especially on the transfer of physical storage media, but also affects the availability and the bandwidth of any required data connection.
- *Time Cycle* – How transactions and workflow are organised along time. This is important when the amount of archive items requires having many working batches. At a given time one item may be completed, within the Factory, scheduled by the archive, or discarded from the process. Moreover if the relationship between Factory and Archive goes on for a while, it is likely that some modifications of the relationship occur.
- *Termination* – Whether the relationship between Factory and Archive has a natural conclusion or may continue indefinitely. The former answer typically applies to the digitisation of an archive, where analogue tapes or film go through preservation and restoration processes in order to obtain new masters. In this case once all legacy formats are transformed to digital the Factory has completed its task, because all newly created archive items should be originated in digital form. However the Archive may intend to apply for the Documentation services even later, without mentioning services for the archive exploitation (publication).
- *Exchanged entity* - (§ 5.2)

5.2 Transaction models according to exchanged entity

5.2.1 Transaction models for original media

Transactions involving the original media are particularly critical when the analogue master storages are moved to the Factory. Indeed inconveniences and failure at this stage may cause the irrecoverable loss of audiovisual material and subsequent lacks for the realised Editorial Objects.

Once a new digital master is available in a file format from the Preservation Unit, the old analogue media become redundant and to give them back to the owner archive is a due action and a further caution.

Factory **on site**

If the Factory, or at least the Preservation Unit, works **on site**, at the Archive locations, the transfer of original media is quite simpler. A hand trolley is sufficient to displace a group of storages from their shelves to the preservation equipments. Usually there is no need to wrap them into a closed box and the transfer time is not that significant.

Factory **off site**

When the two locations are different, the Factory is **off site** and the transfer has a greater level of complexity. A shipment has to be set up. There are a *Sender*, a *Receiver*, and at least a *Carrier*, actors with their defined responsibility. Storages are grouped in batches for the displacing, wrapped in containers (boxes) and the transfer time may not be negligible.

Common aspects

Storage instances can only be in one place at a time, while in case of computer files it is possible to have many perfect copies (bit-to-bit) stored on as many devices. The crucial point is to always know ***What is Where***, and that task applies to both the Archive and the Factory (and to the Carriers too).

Each storage must have its unique identifier, the *TapeNumber*, which should be verifiable by whoever may have to deal with it. A permanent label with the TapeNumber, preferably with a Barcode, has to be stuck on the storage.

Needless to say that the TapeNumber is an important piece of information within the Archive Inventory Process, as being the identifier for the physical storage inventory of Figure 3, and thus of the whole Archive Management.

What is required to all the actors of the transaction - Archive, Factory, Carriers – is to always know the status of each registered storage instance:

- *Present in Archive*
- *Present in Factory*
- *Transferring to Factory*
- *Transferring to Archive*

At the interface points between Archive and Carrier, and between Factory and Carrier (and carrier and sub-carrier) the typical details of a mail posting will be exchanged (sender, receiver, destination address, etc) and don't need to be re-invented here.

However it is recommended to record an identifier for the shipment instance (*ShipmentID*), which should be associated to the storage instance for the time that the transfer is in progress.

Although the concept of *Batch* is presumably derived from displacing issues, it should be kept separate from the Shipment because it encompasses the whole process for which is a management entity.

Moreover it must be remarked that ***the Batch is related to Editorial Objects, while the Shipment is related to physical storages***.

Issues on identification and labelling of removable media

Clearly the removable media must be recognised without inspection of their content, in order to permit the management (displacement) during a whatever work flow.

When the archive sets up the delivery of original media to the Preservation unit of the PS Factory the status of the labelling of each tape can be one of the following:

- *No label at all* - there is here a serious lack of inventory, that is the Archive is not able to manage its assets and, as far as known, the tapes could even be blank. In this case a new labelling and a contextual physical storage inventory can improve the situation, with the advantage of selecting an identification and labelling mechanism suitable for the following processes. The archive cannot be considered inventoried, because of the lacking of any relationship with Editorial entities. See §4.4.1
- *A textual label without barcode* - Also in this case a new labelling and a contextual physical storage inventory improve the situation, with the same advantage mentioned above, but with the additional difficulty that it might be necessary to remove the legacy label in order to correctly stick the new barcode. However it must be remarked that the Archive may have a sort of inventory of editorial objects related to the physical storages by the textual label information (a weak link, but a link). Those relationships are kept only if the storage inventory format includes a field for the label information.
- *A barcode not suitable to process* - This case occurs when the Archive has the inventory of its physical storages, labelled with barcodes, but there are serious technical difficulties for the Preservation Unit to work with those labels (e.g. because of troubles with barcode reading with available devices). A minor issue is that the Preservation Unit may not trust the Archive about storage identifier uniqueness. Apparently the solution is as simple as for the previous cases, a new labelling and a contextual physical storage inventory. However to keep only a reference to the former identifier may cause troubles to the Archive for managing the original media and thus the process should be explicitly assented.
- *A barcode suitable to process* – according to criteria to be defined by the PS Factory.

Eventually, when a new labelling is required, it must go along with the storage inventory, taking into account also the Archive requirements on the former labels or identifiers.

This need is supported by [XMAD] within the definition of *StorageType*, which includes:

- name – that is the current storage identifier (required)
- nameIssuer – at identifier of the entity (person, organisation, process) that issued the current name
- A list of FormerNames including for each of them the issuer name and the withdrawal date.

5.2.2 Transaction models for essence

The discussion of transaction models for essence is strictly related to the issues described D12.4].

As remarked above, essence data can be duplicated as many times as needed, but they are bulky (to store them is expensive). Transaction for essence means also to transfer the responsibility about preserving those data. In most cases the process can be summarised by the following steps:

- A copy of the essence data occurs from the Sender to the Receiver
- The success of the copy is verified at the Receiver side
- The Sender is authorised to remove the source copy
- Update of the Accesses information

Factory on site with common storage service

In this case not only the Factory works at the Archive location, but also both share some kind of advanced storage service. Essence data transfers may apparently happen such as described above, but what actually takes place in the back end depends entirely on the implementation of the common storage architecture and it is mainly opaque.

Basically for this model the Essence transfer is rather the modification of the status and of the permissions.

Higher level software components must be set up in order to take the greatest advantages from the storage services, avoiding operations, which may carry to waste of capacity or other inefficiencies.

Factory on site with LAN

To have some high bandwidth local network is most easily affordable when the Factory works on site. Nowadays Ethernet 100Mbit/s infrastructures are the entry level and GigaEthernet is almost commonplace.

Most of the security issues can be solved at the boundaries and therefore Factory and Archive systems can access a variety of exchange protocols (ftp, smb, http, file).

The basic assumption is that the two actors have a separate jurisdiction on their respective systems.

Although the default scenario implies data duplication for the Essence transfer, it is possible to avoid this step if both the Factory and the Archive systems can share a File server. In this case the transfer of data ownership can be achieved also by moving file links and changing the permissions.

Factory off site with data network connection (WAN)

When the Factory works off site, it should be possible anyway to set up a suitable data connection.

The differences with respect to the previous scenario are:

- There is the need of a provider for the connection; there are greater security issues and limitations of the available protocols.
- The bandwidth will be typically lower with greater transfer times, possible connection interruptions, and more attention required to data integrity (verification of checksums).

Factory off site without data network connection

In this case we have the Factory off site and absent or poor data connection (at least for the Essence data).

This model can be similar to that already described for the original media, because the exchange will be based on the transfer of some kind of removable media (data tapes, DVD, removable discs, etc). The main difference is that Essence data can be recorded on those media as files and thus the Receiver can extract them at any time.

Common aspects

The essence data are actually binary material sources and are contained in files. To say it with the terminology of the MAD Area data model, Files are a type of Storage for Material Sources (see Figure 5), and so they are also involved in the Archive Inventory Process, as shown in Figure 3.

By default the identifier for a given file is its name (*FileName*), but that is clearly a weak identifier because of the easiness for changing it and the difficulties to set up and run a suitable file name policy.

The model developed within the PrestoSpace MAD Area specifies that a defined Material Source be also identified by the Material instance that it is able to generate ("canonical play"). And the Material instance is identified by the UMID[5]. It is useful to remind that the UMID has to be generated and provided by the entity emitting the Material instance.

When a Material Source is required to be transferred (*getMaterial(UMID), insertMaterial(UMID)*) the owner of the Source has to provide the information about the set of available *Accesses*.

The verification of the File identity is permitted through the control of the checksum (md5), and the file size information allows the tracking of the transfer progress.

The exchange mechanism implemented by PSO is defined in §6.2.5

[5] The UMID is the Material Identifier as defined by SMPTE330M. In textual documents, as XML, it must be recorded as a string starting by "0x" and giving then the UMID value in hexadecimal (each byte is coded with a couple of characters in the range [0-9,A-F]). The UMID has 32 bytes with the a possible extension to total 64 bytes.

5.2.3 Transaction models for metadata

A couple of remarks are a premise for discussing the transactions of metadata:

- We don't mean here metadata as process management information, but on the contrary we care about the information about the audiovisual works (Editorial Objects) and materials. Likely that information is going to be limited when going from Archive to the Factory and enriched on its return path.
- The size of metadata is generally considered negligible compared to that of essence, and that's is almost true. However some attention should be paid because it is quite easy to produce large xml files, when using extensive generic schema (where some "verbosity" is required to avoid ambiguities) together with a great level of information detail. Data lossless compression is the possible solution.

Anyway it comes out that the issues about Factory location and network bandwidth are less important in the case of metadata, provided that some kind of connection is available. One could even argue that in the worst case metadata transfer could be arranged through the exchange of cheap removable data storages, such as CDs and memory cards.

So the transaction models on metadata have historically been designed around the relationship between metadata and essence.

Metadata embedded with Essence

This model is supported by several interesting arguments:

- Size of metadata will not impact on size of Essence+Metadata.
- Embedding constitutes a strong link between Essence and Metadata. You cannot lose your metadata (unless you lose the Essence too).
- You don't need to think about metadata transfer, because that's already solved together with the Essence problem.

Such option is also viable at the present time because of the availability of formats, such as MXF, supporting that feature. Moreover some formats, including MXF, also support the ascription of pieces of information to Material timeline intervals or single points.

Disadvantages on this model do exist and will be discussed below. However it is fair to remind that we are describing only the exchange activity, and that this choice should not constitute a constraint for the persistency models at both source and destination points.

Metadata separate from Essence

The advantages of this model partially oppose the drawbacks of the one described above:

- You don't need to process the Essence wrapper in order to access metadata (writing/reading).
- You don't need to move Essence and metadata at the same time; otherwise you need to wait for the last one to be ready.
- Metadata formats are independent from Essence formats (and thus from the tools specific for the latter)
- Metadata are not linked to any specific copy of Essence (master, rather than lower quality copies).

This option is affordable if metadata provide reliable links to the related Essence.

One adverse argument sometimes complains about the fact that handling the Essence you might not have links to the metadata. That's somehow misleading as an inventoried system must always be able to manage the association between Editorial Objects and Materials (see 0).

Adopted approach

The approach implemented by PSO is that originally put in place by the MAD work Area and it is ***oriented to keep separate transactions of Essence and Metadata***, although both are clearly related.

Taking as a reference the diagram provided by Figure 5, the whole metadata set is contained within *the Editorial Object Document* (*EOD*) either directly or by reference to Material instances. In particular the EOD must contain the information about the Materials that realise the Editorial Object, primarily the UMID, which is going to be the key to be used in order to get the information for accessing the Essence data (material sources).

The transaction of an Editorial Object Document is going to be a validation point, which will determine the status of the Editorial Object respect to its Materials:

- *Complete* – All Materials are known and available.
- *Waiting* – All Materials are known but not all of them are available. An Essence transaction may be in progress.
- *Lacking* – At least a Material instance is not known.
- *Error* – In case of failure of actions directed to change the status form Lacking/Waiting to Complete.

Information stored within knowledge base repositories

The Documentation Units working within the PrestoSpace Factories will include tools for semantic analysis that make use of knowledge and information management services. Those services are originally made up of an “ontology” [WKPD search for ontology (computer science)] and an initial knowledge base (KB) [WKPD search for knowledge base], which is going to increase as a side effect of the semantic analysis process. The expanding may reach then saturation in the number of entities.

The information piled up within those knowledge base repositories is going to be very useful for future information retrieval in the Archive exploitation process. However it must be remarked that it is not possible to specifically assign any information elements of the knowledge base to Editorial Object instances.

The Archive exploitation can be considerably improved by the use of information retrieval tools relying on knowledge base repository. Therefore there is an issue about how to transfer the information between a documentation KB instance and a publication/exploitation KB instance.

The currently identified possible scenarios are subject to change in the future because of the evolution of this technology. The PrestoSpace MAD Area has examined the following cases:

- *Services integration* – the KBs of Documentation and Publication are strongly integrated and a continuous update mechanism is configured between the two KBs. This option depends on many constraints and would produce a communication independent from the PSO interface, which is not generally recommended.
- *Traditional archive exploitation services* – In this case there is no KB at the Archive publication end. Therefore the Archive publication has to rely uniquely on the information contained within the EOD.
- *KB export as RDF* – The documentation KB is exported, whenever requested, as RDF XML file [WKPD search for Resource Description Framework]. This option may rise to constraints in the update process of the Publication KB, however it is recommended because it carries to a greater independence between the PrestoSpace Factory and the Archive exploitation.

5.3 Model of relationship Archive-PSO

5.3.1 Main entities

The entities concerned with the work of the PSO, defined in §0 are described underneath, while Figure 10 provides a diagram showing the relationships among them.

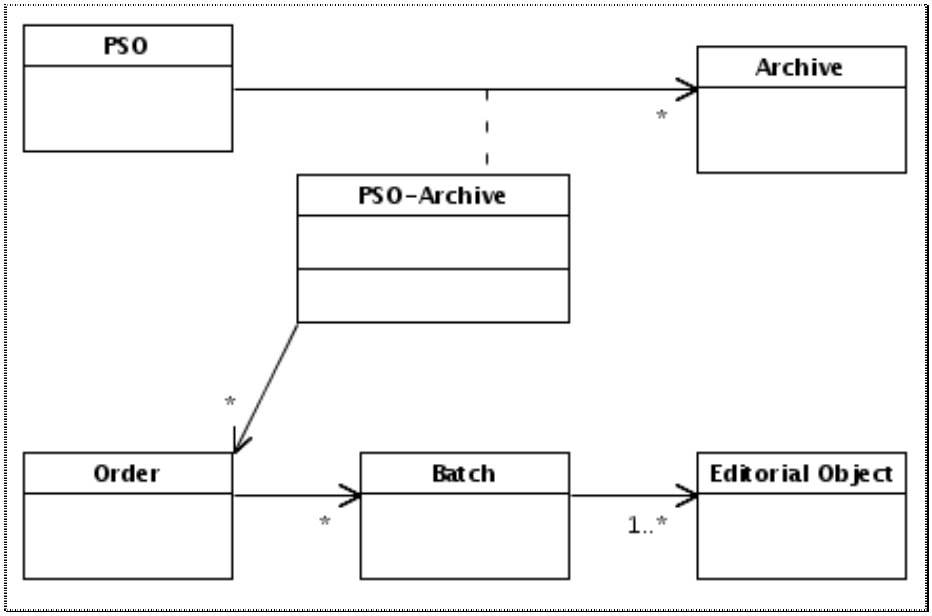


Figure 10 - Diagram of PSO-Archive model

Archives

The Archive is the partner of the PSO for the main identified transactions (see §4.3).

We are not interested directly in the archive entity, but in its relationship with the PS Factory. We assume that a single Factory may have to serve several Archives.

The PSO needs to manage the information about this relationship and to set up all that is required to work regularly.

Orders

An Order is a defined work process that is established and agreed between an Archive and the PS-Factory. For each Archive there may be several Orders. It includes all the relevant details of the requested work.

All the *Working Units* submitted to the PS Factory have to make reference to a defined Order, that has to be considered active all along its lifetime, from creation to expiration or cancellation.

Working Units

By *Working Unit* we mean the elementary entity object of the PS Factory process, and for which the PSO accounts the transaction from/to the Archive and reports the status about the progress of work.

The logical Working Unit of the PS Factory is the Editorial Object, as defined in §4.2.2.

Of course the Editorial Object instances must always carry the reference to the Material Sources that realise them. In fact several processes internal to the PS Factory, especially within the Preservation and Restoration Units, deal basically on the Material Sources and may not interested in the concept of Editorial Object. The PSO, as being responsible of keeping the inventory within the PS Factory, must supervise about the consistency of all the entities related to each Working Unit.

Batches

Several or many Working Units can be grouped by the Archive, according to various criteria such as physical transfers, delivery dates, editorial priorities, and logistical questions, in order to set up transactions with the PSO and simplify the work flow monitoring. These groups of Working Units are named *Batches.* All the Working Units belonging to a Batch must have some common properties, such as the Order or the time of submission to the PS Factory.

5.4 Sequence of processes

5.4.1 Setup & initialisation

In order to allow any subsequent process, it is required that the relationship between Archive and PS-Factory is formally and correctly set up.

In practice this goal may be achieved either in a customised way, in which the *PS-Factory Negotiator* will take into account the various aspects of the Archive status as those discussed in §4.1 and §4.4, or in a more general manner by the simple use of a **RegisterArchive** method provided by the PS-Factory interface (the PSO component).

5.4.2 Submit work

The Archive process of submitting work to the PS-Factory can be divided into the following processes

Register Order

The process is made of the definition of an Order and its registration at the PS-Factory, with validation.

Similarly to the Archive Registration, this task may be accomplished either informally and in a customised way or by the use of a specific service offered by the PS-Factory.

Insert Materials

This process implies the transfer of Material Sources, typically original analogue media, to the PS-Factory, typically the Preservation Platform.

The PSO will receive from the Archive the information about transferred materials, including material identifiers and shipment details (shipment id, shipment date, shipment place) and will wait for a confirmation of the reception from the actual recipient component (e.g. the Preservation Platform).

In the case when Material Sources are media files (digitisation already done), the PSO must be able to access them and insert them into its EMS component.

Register Editorial Object

This process permits the assignment of an Editorial Object to the PS-Factory. Information about basic identification, any useful legacy information, and material realisations have to be communicated at this stage, together with the fact of being member of one (or more) Editorial Collection(s), which has(have) also to be registered.

Register Batch

This process creates a Batch of Editorial Objects for the given Archive and Order. The result of the process is that the PS-Factory can start the work defined within the Order on the given Editorial Objects, provided that all dependencies, such as availability of Materials and registration of Editorial Objects are solved.

Figure 11 shows a use case diagram for the whole **submit-work** process.

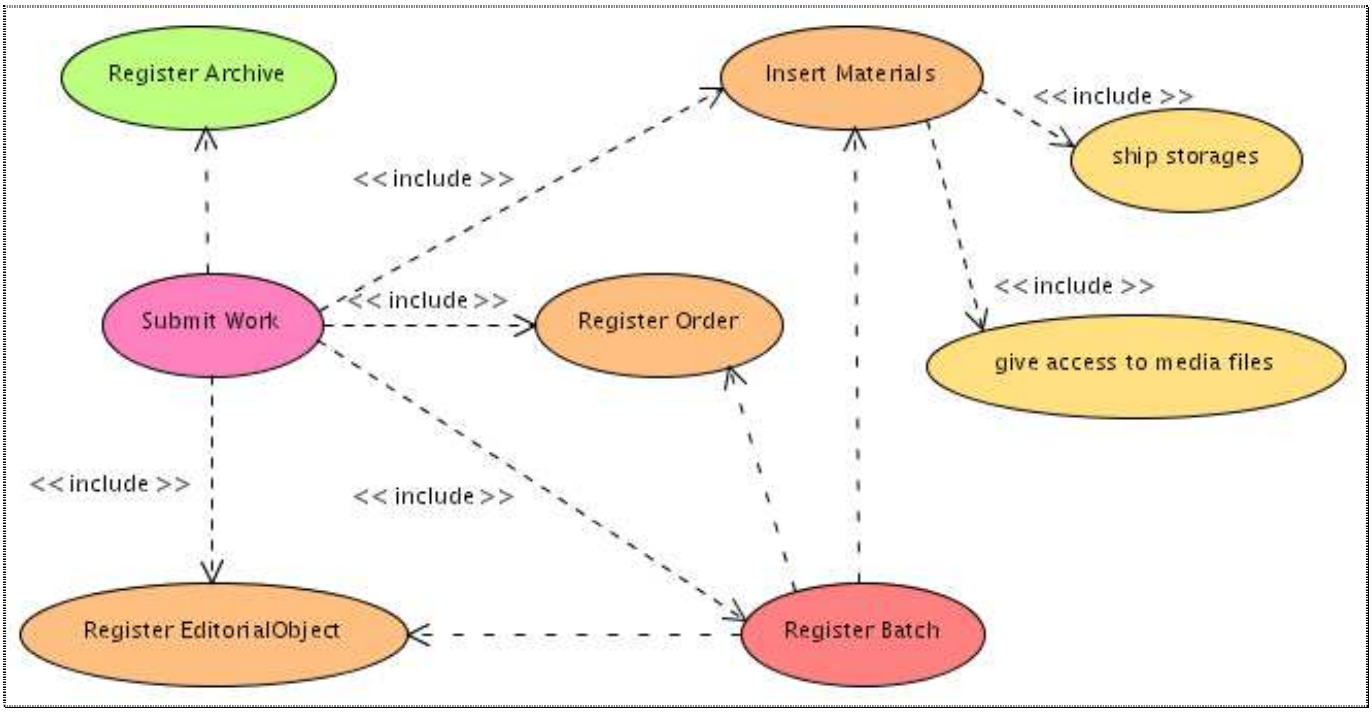


Figure 11 - submit work use case diagram

5.4.3 Get back results

Once the Archive is registered at the PS-Factory, it should be possible for it to access the information about the status of any registered entity and thus to obtain a monitor of the work in progress.

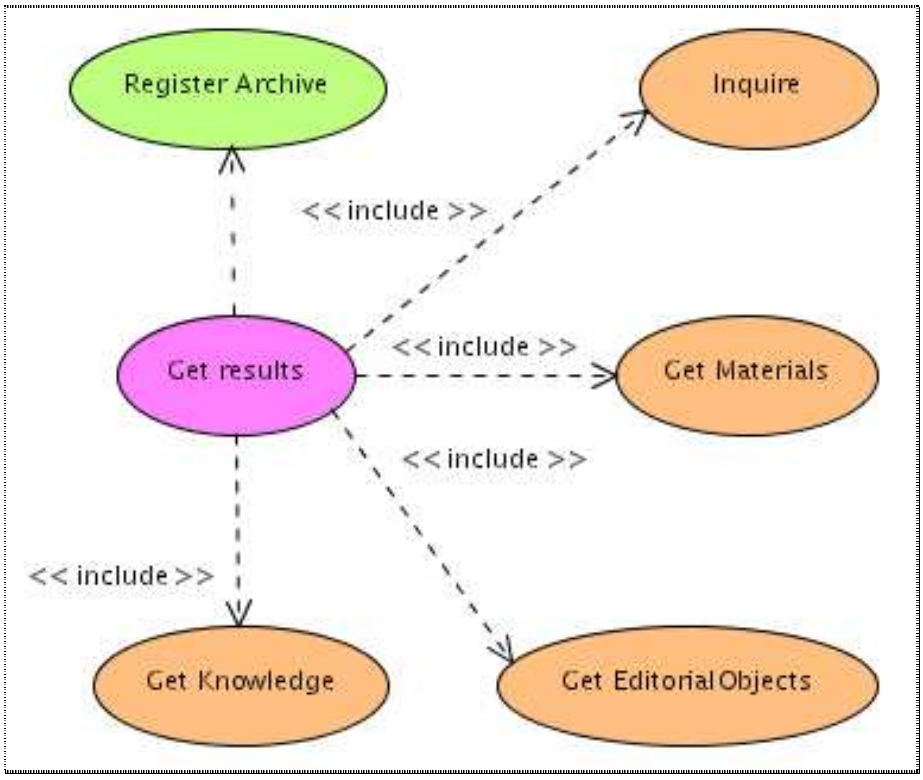


Figure 12 - get results use case diagram

In addition to the **Inquire** process, the diagram shown in Figure 12, there are three actual delivery processes:

- **Get Materials** – That can be seen as both the delivery of any newly created material, e.g. from digitisation, and the return of the original media. The actual delivery of Materials.
- **Get Editorial Objects** – That is the delivery of the complete set of enriched metadata. Differently from Materials, the Editorial Object delivery doesn't necessarily depend on completion of work.
- **Get Knowledge** – That is the delivery of the knowledge base gathered for the given Archive. The provided RDF file may optionally contain only the update of the KB after a given date and/or only the information related to the entities identified by a given list of URIs

5.5 Delivery options

5.5.1 Wrapping formats

The wrapping formats define the packaging of exchanged entities, either essence or metadata or both, and are interesting because of the practicability for use of possible available tools, providing a certain degree of independence from the nature of the contained stuff.

Generally speaking the wrapper is a type of file which is able to host several types of related content. This is specifically used for exchanging all together a multiplicity of components, simplifying the transaction, and sometimes also for providing concurrent access to the various related part.

Concerning the Essence, the wrapping formats provide basically the same functionalities of a multiplexed format (a container for both video and audio data, which is used for audiovisual presentation), such as an MPEG Program Stream, with in addition the capability to not require a single specific coding/compression format for the essence.

It should be remarked that wrapping is not generally required. Essence wrappers can be necessary in order to allow synchronous presentation (play) of Material with separate Sources (video from one file and audio from other files, however this is an issue for the player application and a multimedia playback platform not requiring this is conceivable (e.g. Direct Show).

MXF

The reference wrapping format for high quality essence within the PrestoSpace project is certainly the ***MXF***.

That format is based on set of SMPTE standards which define the wrapper itself and the modalities for encapsulating each kind of content according to its specific nature and format.

For example it allows video data to be uncompressed, MPEG coded, DV coded, etc.

The MXF file is also able to contain metadata. However, in addition to the issues already discussed in §5.2.3, it currently puts several constraints on the metadata formats and coding, which will be hopely solved in the future.

It is also helpful to consider the question of MXF *Operational Patterns*, which define a set of suitable limitations to the ways in which the essence data (sources) produce the output audiovisual Material. The mechanism of Operational Patterns permits the definition of several layers of complexity, allowing simpler implementations to be deployed. The drawback is that the selection of an Operational Pattern, or at least to indicate the most complex desired one, is also required.

TAR and similar bundles

A mechanism deserving consideration and often successfully adopted to gather together a set of strictly related files, typically when exchanging metadata but also sets of essence data files - such as key frames -, is the TAR bundle, also named tarball.

Tar (correctly tar, with low case t) is a type of archive, as created by the homonymous command, without compression option applied. A list of files, including directory structures and file system information, are gathered together in one file. Optionally a lossless data compression algorithm is applied thence (e.g. gzip, compress, bzip2).

Further details are available at [WKPD search for tar (file format)] including limitations.

Clearly the *tar* archive doesn't define the nature of its content that needs to be agreed together with the transaction details.

The PrestoSpace MAD approach includes the *tar bundle* as a mean for exchange Materials. This is particularly directed to keyframes images or stripe image, because it permits to avoid multiple transactions for small data entities. However the same approach can be extended to larger media files.

Moreover nothing prevents to use one wrapper within another one, so MXF files may be gathered together with other files within a tarball.

5.5.2 Metadata standards and options

As outlined in §4.3.1, the PS Factory has to exchange metadata with the Archives, and it is a bi-directional deal. On one hand the Factory expects minimum identification information, and it generally welcomes any legacy information. On the other hand the Archives are going to receive both technical information, including reporting from preservation and restoration processes, and the results from the documentation process (enhanced metadata).

PrestoSpace Deliverable D15.1 provides a description of some current audiovisual documentation models and some relevant metadata standards.

Now it is difficult to find audiovisual Archives whose legacy information is currently compliant to a fresh metadata standard and requiring the Factory documentation services. In most cases the organisations have their own documentation model, some aiming to keep it or slightly enhance it, other ones aiming to change it more radically.

Some years ago the lack of standards about metadata within the audiovisual scope raised many independent initiatives and efforts. Today the selection of a single standard among the differentiated offers is controversial.

Nowadays the ROI from the complexity of metadata standard development is decreasing and it is believed that the best benefits come from clear definition of meaning and formats and from clever setting of mapping.

In particular it is deemed that some mapping is required at the Archive side:

- When exporting their legacy information

  This is required at least for a minimum identification set. Unmapped information can be transferred anyway as opaque ancillary material information, but without any warranty of correct decoding/understanding.

- When importing the enhanced metadata from the Factory.

  This is required for those Archives intending to keep most of their pre-existing data structures

The Editorial Object Document (EOD) defined within the PrestoSpace MAD Area is able to cover the needed requirements.

The EOD format partially defines how to code some information elements, specifically concerning the relationships between Editorial Object and Materials, and makes reference to EBU Tech.3295 (P_META) and to MPEG7 for specific elements, as described in Figure 13

It happens that the P_META standard is particularly relevant, although not exclusively, for coding the legacy information, while the MPEG7 standard is used for information more strictly related to temporal decompositions.

Concerning the information coming from the Archives the statements given in Table 2 apply.

The **"Realisations" are mandatory**.

The **"Identification information"** (such as titles, identifiers, publication details, and credits), which are legacy, are **"Recommended"**, that is:

- Every element is optional by itself
- It is recommended to the archive to provide at least some of the possible information
- If no other information is provided, the only identification datum for the Editorial Object will the attribute "id", which is an arbitrary dumb string/number.

The other legacy information is optional and welcome.

Table 2 – Metadata from the Archives

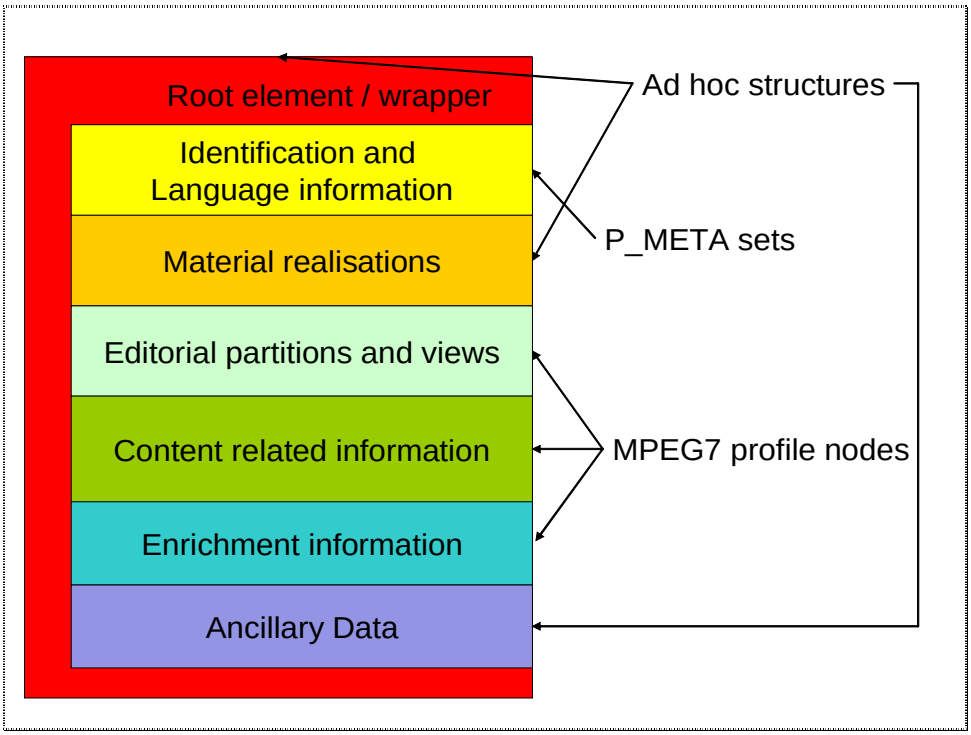


Figure 13 - Structure of Editorial Object Document

Delivery options

The EOD is the default delivery format option. Any other alternative option requires a further mapping effort with concrete possibility of loss of meaning and detail, without providing at the other hand so much advantage.

However the existence of performing tools for XSL transformation allows to conceive the implementation of "*export as*" facilities, as long as we consider XML formats. These options may be suitable to those Archives feeling more comfortable to deal with a particular metadata standard.

- *Dublin Core* – It covers sufficiently well a set of information for identified entities. It suffers for loss of detail and it gets in troubles concerning relationships based on timelines and temporal decomposition.
- *SMPTE/DMS1* – The original descriptive schema conceived for MXF embedding. It may be required in case of completely MXF based transactions, because although any descriptive schema could be used in theory, in practice the standardisation procedures and the issue of KLV coding have discouraged other proposals. There are proposals for embedding any kind of XML coded metadata within MXF, which would be rather opaque in the MXF environment, but this effort is dissuaded by the non-necessity of metadata embedding in most cases. Issue: KLV coding.
- *P_META* – Although a complete mapping to P_META is possible, the exact details of such mapping need to be defined. A smoother result could be obtained by a change request for a specific new set.
- *Mpeg7* - For some information elements there would be some loss of meaning and detail, in a very few cases a request of extension to the standard are needed. However the result would be more than acceptable in most situations.
- *other* – other future options may be added afterwards, if considered interesting

It must be observed that the EOD includes the links to the Material related to the Editorial Objects, and this support can be partially lost when selecting another metadata format. However at the delivery time the service described in §6.2.5 permit to give access to all the materials related to an Editorial Object by simply giving its identifier.

5.5.3 Essence formats

The issue of essence formats deals with the compression, coding, and file formats for material sources.

The most important features of an Essence format are:

- Quality – we won't try to define this here.
- Number of bits – This depends on sampling and on compression/coding. The sampling parameters (possible/suitable) depend on the format of the analogue source (please notice that analogue video is already sampled vertically and temporally). The compression/coding can be generally classified as:
    - Uncompressed – Samples are simply coded without trying further to decrease the number of bits
    - Lossless compressed – Data are compressed in order to reduce the number of bit, taking advantage of *redundancy*, but the process is entirely reversible.
    - Lossy compressed – Data are much more compressed taking into account also *subjective redundancy*, this process is not reversible because some information is lost for ever, and thus it may affect both subjective and objective quality.
- Access modalities – This depends on compression/coding. Complex compression schemes may give further constraints to the timeline access points, impacting on arbitrary access for both seeking and editing.
- Format support – This a format management issue. Is the format open? How versions will be managed in the future? From this it depends the actual possibility to access material in the long term.

Master quality material

By ***master quality*** we mean the material instances with the highest quality, used as reference, and from which any equivalent or lower quality may be derived.

Generally a derivation process doesn't generate a new master, because the quality cannot typically be increased, however a few exceptions exist:

- A lower bit-rate copy obtained by lossless compression can generate another master quality.
- A restoration process, if successful, may generate even a better quality. However one could argue that doing the restoration again, in a different way, could carry to better results. So it is controversial to say that at the output of a restoration process we have a "true master".

The master quality is the post-production application scenario and is the object of Archive long-term storage.

Logically, in the digital file domain, any quality lower that the master could be re-created from the master.

Concerning the Audio, the number of bits generated, once defined the sampling parameters, is so lower than for video Essence that the Audio Master is usually an Uncompressed coding format (PCM).

Conversely, the number of bits for the video is still quite bulky as Uncompressed. Therefore a Video Master is usually either Compressed Lossless or "High-end Compressed", where by "High-end" it is meant a form of lossy compression the parameters of which provide assurance of high quality, on the base of official tests made on critical material.

The identified options for video coding are given in Table 3, where the reported values refer to standard definition even if the given options also support higher (some even any) resolution.

Options supported by PrestoSpace for Master Quality Materials:

- *MPEG-2* - ISO/IEC 13818 – provides high end compression, at 15-50 Mbps for standard definition
- *JPEG2000* - ISO/IEC 15444-1 / ISO 15444-3 – provides both lossless and high end compression, at 80-100 Mbps for standard definition (lossless) [6]

Options not currently supported by PrestoSpace (could be provided, if possible on specific request):

- *DV* - IEC 61834 – provides high end compression Intra coding, at 50 Mbps for standard definition
  - also 25Mbps
- *MPEG-4* - ISO/IEC 14496-10 / H264 – provides high end compression (better than MPEG2 but still less supported) – **Considered interesting for High Definition Materials**

Table 3 - Video essence compression/coding formats for "master quality materials"

Broadcast quality material

We mean by ***broadcast quality*** those material instances the quality of which is appropriate for publication through current audiovisual publication media (terrestrial and satellite television, DVDs). Notice that the same instances might not provide that expected quality in the future, when published though more demanding media, such as over higher definition.

The customer Archive may ask to the PS Factory to deliver, in addition to a master quality material, also a broadcast quality material. Actually although it should always be possible to obtain the latter from the former subsequently, it is acknowledged that a so profitable opportunity to produce such a copy is quite attractive, as later it could be more troublesome.

Clearly the selection of an option here is quite less critical than for the master quality, even though the cost of processing would suggest making the wisest possible choice.

In this case typically also audio is compressed and the options given in Table 4 refer to coding schemes that include both video and audio coding. Combination of schemes belonging to different standard families is discouraged.

Option supported by PrestoSpace for Broadcast Quality Materials:

- *MPEG-2* - ISO/IEC 13818 – provides appropriate quality, at 4-8 Mbps for standard definition
  - *Audio may also be MPEG-1* ISO/IEC 11172

Options not currently supported by PrestoSpace (could be provided, if possible on specific request):

- *MPEG-4* - ISO/IEC 14496-10 / H264 –better than MPEG2 but still less supported
- *Windows Media 9* – Former proprietary Microsoft codec, currently submitted to SMPTE for standardisation
- *Dirac* – Open Source (GPL) wavelet codec developed by BBC/RD, published on SourceForge
- *Theora* – Open Source (GPL) codec developed by Xiph.Org Foundation (part of OGG project)

Table 4 – Audio/video essence compression/coding formats for "broadcast quality materials"

It should be remarked that a broadcast quality material could be produced anyway internally to the PS Factory, as it is appropriate for Content Analysis within the Documentation Unit.

6

ISO/IEC 15444-1:2004 | ITU-T Rec. T.800 defines a set of lossless (bit-preserving) and lossy compression methods for [..] digital still images.

ISO/IEC 15444-2:2004 defines extensions to part 1

ISO/IEC 15444-3:2002 specifies the use of the wavelet-based JPEG2000 codec for the coding and display of timed sequences of images (motion sequences), possibly combined with audio, and composed into an overall presentation. In this specification, a file format is defined, and guidelines for the use of the JPEG2000 codec for motion sequences are supplied.

Browsing quality material

By ***browsing quality*** we mean those material instances the quality of which is sufficient to appreciate and recognise the content, albeit some impairments might be quite perceptible. Those materials are often used as "proxy" for their higher quality corresponding ones when there are lesser resources, in terms of bandwidth and storages, or when there is no intention to make available a better quality copy, as for free sample preview. In the PrestoSpace domain this level is interesting for the Publication Platform

As for the Broadcast Quality, also the Browsing Quality Material could be obtained in any time from higher quality copies, but the Archive might find suitable to ask to the PS Factory the production and delivery of this kind of Material, especially for saving on the cost of processing in another time.

Also in this case the options given in Table 5 refer to coding schemes that include both video and audio coding. There is not a specific support by PrestoSpace, as the format choice doesn't imply particular constraints within the Factory.

Note that typically the listed picture coding schemes allow image downsampling (horizontally, vertically, and even temporally), which is typically required for browsing quality material.

Here the target bit-rate currently falls into the range 0.1-2 Mbps for all schemes, and therefore there is a quite broad range of possible quality results.

- *MPEG-2* - ISO/IEC 13818 – not state of the art scheme for this target application
- *MPEG*-1 - ISO/IEC 11172 -
- *MPEG-4* - ISO/IEC 14496-10 / H264 –
- *Windows Media 9* – Former proprietary Microsoft codec, currently submitted to SMPTE for standardisation
- *Real* - proprietary video codec developed by RealNetworks
- *Dirac* – Open Source (GPL) wavelet codec developed by BBC/RD, published on SourceForge
- *Theora* – Open Source (GPL) codec developed by Xiph.Org Foundation (part of OGG project)

Table 5 - Audio/video essence compression/coding formats for "browsing quality materials"

Again it must be remarked that such browsing quality material might be produced anyway internally to the PS Factory as appropriate for Annotation within the Documentation Unit.

6 PrestoSpace Factory

6.1 The PrestoSpace Services

A PrestoSpace service is modular function within the PS factory which can be utilised (ordered) by an archive, e.g. digitisation, restoration, documentation, transcoding, D&Q analysis,....

The following sheets specify for a certain service the needed input parameters, the needed/produced data input/output qualities and needed/produced metadata input/output information. It not specifies how data, metadata and service parameters are represented (encoded).

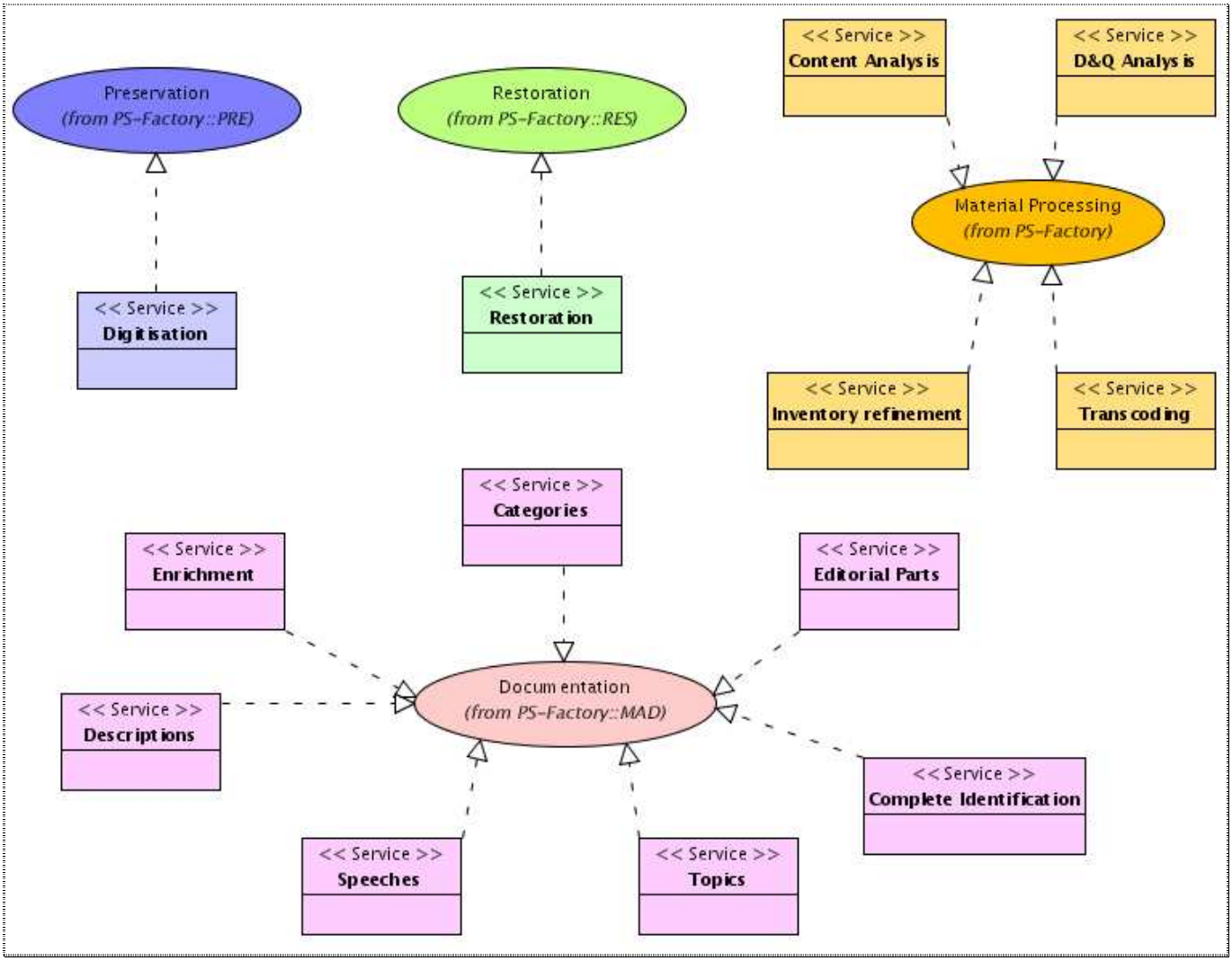


Figure 14 - Use Cases realisation through Services

There is a relationship among Use Cases and Services, it depends on the way a Service on its implementation realizes a Use Case. Services are to be intended as logical interface available at PS factory level in a way that also other Services can interoperate together performing completely or partially the Use Cases objectives. Depending on the granularity of a Service, more than one Services could be necessary to accomplish a Use Case. Figure 14 depicts the described relationship.

6.1.1 Digitisation

The digitisation service enables the audiovisual material transfer into digital format, through the available preservation tools and subsystems. This is the main goal of the preservation process, which includes the physical safeguard and repair of analogue media, possible applying:

- *Inspection* – the objective of which is to determine whether an item is to be processed through a specific preservation chain depending on media status (e.g. normal chain, expert chain or put aside for further examination).
- *Preparation* - to apply the required treatments before media playback, such as cleaning, baking or physical repair.

Table 6 provides some information about the service parameters.

Service Name	**Digitisation**		
Function Overview	Digitisation of A/V analogue material (or non file based digital material)		
Service Input Parameters	• Workflow options o Inspection o Preparation	• Output options o Quality / coding parameters o Format • Original Media re-labelling options	
Essence Input Quality	Original media at master quality	**Essence Output Quality**	Digital Master (& return original media)
Metadata Input	Preservation Batch Document	**Metadata Output**	Digitisation report as Material derivation (EOD Format)

Table 6 - Digitisation service parameters

6.1.2 Restoration

A schema of the service parameter is provided by Table 7, from which it should be remarked that the service can be requested for specified portion of material indicating the track type (audio/video) and the time intervals of interest.

Service Name	**Restoration**		
Function Overview	Film, video and audio restoration of digital essence with respect to specified defects		
Service Input Parameters	Audio Restoration o Yes | No (Boolean) o From, To (MediaTimePoint) o Job description (Text)	Video Restoration o Yes | No (Boolean) o From, To (MediaTimePoint) o Job description (Text)	
Essence Input Quality	Digital Master	**Essence Output Quality**	Digital Master
Metadata Input	EOD (opt.ly including) o Technical metadata o Digitisation reports o Content Analysis data	**Metadata Output**	Restoration Report as Material derivation (EOD Format)

Table 7 - Restoration service parameters

6.1.3 Defects & Quality Analysis

Also this service can be requested for specified portion of material indicating the track type (audio/video) and the time intervals of interest. Details on parameters are given in Table 8.

Service Name	**Defects and Quality Analysis (D&Q)**		
Function Overview	Fully automatic film and video defects and quality analysis of digital essence. Defects and quality measures are described and can be visualised by the D&Q Summary Player		
Service Input Parameters	Audio D&Q Analysis o Yes | No (Boolean) o From, To (MediaTimePoint) o Job description (Text)	Video D&Q Analysis o Yes | No (Boolean) o From, To (MediaTimePoint) o Job description (Text)	
Essence Input Quality	Digital Master	**Essence Output Quality**	None
Metadata Input	EOD (opt.ly including) o Technical metadata o Digitisation reports	**Metadata Output**	D&Q properties, e.g.: o derivation noise/grain level o dust level o big distortions (dropouts and missing frames) o blocking level, sharpness level…. (EOD Format)

Table 8 - D&Q Analyisis service parameters

6.1.4 Content Analysis

Table 9 gives the details about the service parameters. The video analysis in particular can be set up to provide various information.

For a deployment of services point of view, this service can be associated to those provided by the Documentation (or Restoration) Unit.

Service Name	**Content Analysis**		
Function Overview	Provides several information on audiovisual features and content indexing. Requires Digitisation / Transcoding		
Service Input Parameters	Audio Analysis o HigherRecall | Higher Precision	Video Analysis o HigherRecall | Higher Precision o ShotBoundaries o ShotSimilarities (from Shots) o KeyFrames (from Shots) o StripeImages o CameraMotion o MotionActivity o VisualFeatures (from KeyFrames) o SmallerPictures (KeyFrames/StripeImages)	
Essence Input Quality	Broadcast Quality	**Essence Output Quality**	KeyFrames, StripeImages
Metadata Input	EOD	**Metadata Output**	EOD enriched

Table 9 - Content Analysis service parameters

6.1.5 Inventory refinement

This service is intended for increasing the accuracy of timeline information of Materials realising the Editorial Objects, especially concerning the multi-tape and multi-programme cases described in Table 1, partially implementing a complete inventory information sub-case showed in Figure 6. Optionally a new Material instance, made of a single source, might be derived.

Table 10 provides the service parameters.

Service Name	**Inventory refinement**		
Function Overview	Complete mapping of editorial timeline and material timeline. It may derive Material instances made of a single source. Very basic (shot-boundaries+keyframe) Content Analysis can make the process easier		
Service Input Parameters	o selection of Material Realisations by giving either the quality level or the Material identifiers (UMIDs), if known o Derive new Material: Yes | No (Boolean)		
Essence Input Quality	Any Digital	**Essence Output Quality**	Optional, same quality level
Metadata Input	EOD	**Metadata Output**	EOD corrected and opt. enriched

Table 10 – Inventory refinement service parameters

6.1.6 Transcoding

The Transcoding service can either be required by the Archive or needed as a consequence of Service dependencies.

In 5.5.3 three possible material quality levels are defined:

- Master quality material
- Broadcast quality material
- Browsing quality material

For instance the Broadcast quality is the typical input for Content Analysis and Speeches, while the Browsing quality is mostly suitable for other Documentation services. Material created but not requested for final delivery may either be deleted or offered to the Archive as optional additon.

However a specific Transcoding service has to be available within the PS-Factory.

Table 11 provides the suggested options and parameters for the transcoding service.

Service Name	**Transcodig**		
Function Overview	Provides transcoding, and optionally transwrapping, of material		
Service Input Parameters	Video o Yes | No o <Target Quality > o <Target Format>	Audio o Yes | No o <Target Quality> o <Target Format>	Wrapper o Yes | No o <Target Format>
Essence Input Quality	Master | Broadcast Quality	**Essence Output Quality**	Master | Broadcast | Browsing Quality
Metadata Input	EOD	**Metadata Output**	EOD enriched o Material derivation

Table 11 - Transcoding service parameters

6.1.7 Documentation

The documentation area should provide a wide set of services in order to fulfil requirements of Archive documentation models [D15.1] and especially the outcomes of the [BPM].

According to the nature of both the Editorial Object and the service itself, it is possible to understand if the service itself reasonably applies to the main Editorial Object and/or to its Editorial Parts. In some cases, such as Speeches, it is almost irrelevant.

Complete Identification

The service generally applies to the main Editorial Object, however at least *Titles* and *Credits* (seldom the other ones), can be specific of the Editorial Parts. Service sheet is given by Table 12.

Service Name	**Documentation / Complete Identification**		
Function Overview	Add identification information which was missing or lacking from legacy		
Service Input Parameters	On main Editorial Object o Yes | No On Editorial Parts o Yes | No		o Titles o Publications o Credits o Awards o Collections o OriginalClassification
Essence Input Quality	Browsing Quality	**Essence Output Quality**	None
Metadata Input	EOD	**Metadata Output**	EOD enriched

Table 12 – Documentation/Complete identification service parameters

Speeches

This service, details are given in Table 13, will be mainly provided by automatic tools that may require a target language to operate (although some may infer this infermation by themselves) and a higher precision usually requires a human validation and corrections.

Service Name	**Documentation / Speeches**		
Function Overview	Provide text from audio		
Service Input Parameters	o ForceLanguage <languageCode> o HigherRecall | HigherPrecision		
Essence Input Quality	Audio mono PCM 16kHz 16bit WAVE or BWF from BroadCast Quality	**Essence Output Quality**	None
Metadata Input	EOD	**Metadata Output**	EOD enriched with text o Speaker Segmentation o Segment classification (music, noise, speech and silence)

Table 13 – Documentation/Speeches service parameters

Editorial Parts

Segmention into editorial parts is an important documentation service, allowing the other services to provide higher quality results. Service sheet given by Table 14

Service Name	**Documentation / Editorial Parts**		
Function Overview	Find Editorial Parts within Editorial Object		
Service Input Parameters	o HigherRecall | HigherPrecision o NewsItem – ask to find specifically news items as for within Newscast o UseLegacy – ask to try matching with legacy information (if present)		
Essence Input Quality	Broadcast Quality	**Essence Output Quality**	None
Metadata Input	EOD	**Metadata Output**	EOD enriched

Table 14 – Documentation/Editorial Parts service parameters

Categories

The classification of subject, usually performed on Editorial Parts, makes reference to a given set of categories, as shown in Table 15, chosen among those available to the service.

Service Name	**Documentation / Categories**		
Function Overview	Provides subject classification		
Service Input Parameters	o CategorySet <categorySet> - ask to use the given defined category value set o HigherRecall | HigherPrecision		
Essence Input Quality	Optional Browsing Quality	**Essence Output Quality**	None
Metadata Input	EOD including speech and Editorial Parts	**Metadata Output**	EOD enriched

Table 15 – Documentation/Categories service parameters

Topics

This service typically works on Editorial Parts, details given in Table 16.

Service Name	**Documentation / Topic**		
Function Overview	Provides the subject of content		
Service Input Parameters	o HigherRecall | HigherPrecision o Named Entity Recognition (persons, organisations, places, etc.) o Summarization (query-relevant)		
Essence Input Quality	Optional Browsing Quality	**Essence Output Quality**	None
Metadata Input	EOD including speech and Editorial Parts (default)	**Metadata Output**	EOD enriched

Table 16 - Documentation/Topic service parameters

Descriptions

The parameters of the service, given in Table 17, permit various configurations.

Service Name	**Documentation / Descriptions**		
Function Overview	Provides synopsis or itemised description of Video and/or Audio content		
Service Input Parameters	General o Yes | No o Synopsis o Itemised (From, To)	Vision o Yes | No o Synopsis o Itemised (From, To)	Sound o Yes | No o Synopsis o Itemised (From, To)
Essence Input Quality	Browsing | Broadcast Quality	**Essence Output Quality**	None
Metadata Input	EOD including speech and Editorial Parts (default)	**Metadata Output**	EOD enriched

Table 17 - Documentation/Descriptions service parameters

Enrichment

Table 18 describes the features of this service

Service Name	**Documentation / Enrichment**		
Function Overview	Provides references to related published documents		
Service Input Parameters	o HigherRecall | HigherPrecision o Summarization (query-relevant) o External Resource <ResourceURL>– ask to use the given publication source (list)		
Essence Input Quality	Optional Browsing Quality	**Essence Output Quality**	None
Metadata Input	EOD including speech and Editorial Parts (default)	**Metadata Output**	EOD enriched

Table 18 - Documentation/Enrichment service parameters

6.2 Exchange mechanisms

6.2.1 Criteria

The transaction activity needs to be supported by an orchestration between the parties implementing a defined protocol and allowing the management of the relationship by the PSO. The orchestration itself is realised by means of exchanges according to defined formats, while the specific transactions must be supported by the availability of suitable protocols.

The solutions adopted are based on the following criteria:

- Systems interaction supported by ***Web Services*** – [WKPD – search for Web service]
- Structured information exchanged as XML Document according to defined XML Schema
- Standard protocols for essence exchange and access

The overall process generally requires the following sub-cases:

- To exchange (register, update, deliver) Editorial Object Documents (EODs)
- To submit working instructions, including task definition, options, and parameters
- To provide information about work in progress, including notification of task completion or failures.
- To give access to required or created material sources

6.2.2 Management of EODs

The PSO supports the management of Editorial Object Documents by exchange through dedicated web-services in form of XML file.

The provided functionalities are generally the following:

- **Registration** – This is when a new Editorial Object need to be loaded to the PS-Factory from the Archive
- **Get** – When a PS-Factory sub-system, e.g. a Documentation-GAMP, requires a copy in order to perform a process on it (or its related Materials). This implies a lock of the document and the management through CVS mechanism.
- **Put** – When a PS-Factory sub-system asks for updating an Editorial Object, which was previously obtained by "*Get*".
- **View** – Similar to "*Get"*, but without the generating a lock, so it doesn't give the right to subsequent update. This may be used either for monitoring the progress of work or for final delivery
- **Clear** – Ask the PS-Factory to remove records about it (from the Archive).

The PSO offers the same mechanism to both the Archives and the PS-Factory sub-system, while the provided functionalities differ.

6.2.3 Working instructions

The common mechanism is to submit a specific XML file to a Web-service. Alternatively, a specific XML Node can be part of another (wider scope) XML document. Generally the submitter system must indicate:

- The requested services / activities
- The related options & parameters according to the service definition

However the details on format and protocol for submitting instructions are different at the various PSO interfaces

- Archive – within the Order definition, [Orders]. PSO receives instructions.
- Preservation – within PreservationBatchDocument. PSO submits instructions.
- Documentation (&Restoration) – with JobInsertionDocument PSO submits instructions.

6.2.4 Notifications

From PSO to Archive

The communication from the Factory to the Archives about status modifications, process completion or failure, and other events can be achieved in various ways, by agreement with the Archive or subscription of defined notification services.

Document includes a NotificationDetails section where there must be indicated:

- The type of event for which notification is required.
- The Notification Destination and Mechanism, according to an option list, which may include:
  - e-mail – the archive may indicate an e-mail address set up for receiving notifications
  - sms – a mobile telephone number may be given for small message service
  - rss – rich site summary, the archive may access notification messages with a news feed reader
  - web service – the archive may provide the coordinates of a web service set up for receiving notification
  - other rpc mechanism – any other mechanism set up by the archive and based on remote procedure calls
  - any other – anything else suitable, e.g. telephone, fax, standard mail,

From PS-Factory sub-systems to PSO

To be verified with Eurix (Michele)

The general mechanism is defined by a Web Service, offered by the PSO, for receiving notification messages, from sub-systems. Each interface may have defined its own specific events to notify.

The alternative mechanism is when the PSO has to inquire for status of job with the sub-systems.

6.2.5 Material exchanges

The PSO supports and implements the Material exchange mechanisms already defined and in use with the MAD Area activity.

The PSO offers a couple of Web Services for:

- Insert – Materials are provided to the PSO/EMS
- Get – Materials are requested to the PSO/EMS through:
    - list of Material instance identifiers (UMIDs)
    - Editorial Object identifier - for getting access to all the associated materials

Both are based on the exchange of an XML Material Exchange Document. For the *Insert* case it is included in the Web Service call, while for the *Get* case it is given as the result of the call.

The Material Exchange Documents, specified by XML Schema document [XMAD Element MaterialExchange] contains:

- The information about Materials exchanged (& related ones if necessary) – Identifiers (UMID), Material Sources, Derivation information
- The information about how to actually access the Material Sources
    - Direct access through provided services/protocols:
        - file – this is useful only to provide access to resources once they are stored on the same host.
        - smb – Server message block. Protocol for sharing directories, allows both mounting a file system and getting files as for ftp
        - nfs – network file system. Protocol for sharing directories allowing mounting a file system
        - ssh – Secure shell. Protocol for secure remote host access, allows file transfers
        - http (or https)- hypertext transfer protocol (or http secure). Protocol used for web based exchanges, allows file transfer
        - ftp – File transfer protocol
    - bundle – indicates that the requested resource is contained into a bundle (e.g. a tar), the access of which must be granted via another protocol.
    - shipment – used for material sources that are physical storages (tapes or other removable media) that are physically delivered to destination.

The PSO is not meant to send and receive directly the physical storages (such as the original media), however it has to be informed about *Shipments* in order have a minimum track of media transfers. Shipment reception must be notified to the PSO.

For File based Material Sources, the PSO manage the exchanges through the EMS system.

6.2.6 Knowledge base

As discussed above (Information stored within knowledge base repositories), the knowledge base update service might be fully integrated with a communication independent by PSO, but the suggested mechanism contemplates the access to an RDF XML file, as either the result of an Archive request or a regularly made update.

As the size of such RDF file may be not negligible, the access information and sub-sequent exchange are implemented with exactly the same mechanism as for Material Sources (6.2.5).

6.3 Interface PSO-Archive

The PSO supports the sequences of processes defined in §5.4 by offering to the Archives several Web services implementing the mechanisms described in §6.2.

A few XML documents to be exchanged through this interface are defined and the relevant XML Schema are at […]

Concerning Editorial Objects and Materials, the PSO supports the mechanisms defined in §6.2.2 and §6.2.5.

It is useful to remind that transfer of original media occur directly between Archive and Preservation Unit, while the PSO must be kept informed through the Material Exchange Document.

6.3.1 Archive details

The Archive details include information enabling the identification of the Archive for any relevant process into the PS Factory, and permitting to establish any required communication.

The element is specified in XML Schema [XPSOA] and described in Table 19. It is defined to be used in archive registration, archive update – in case it is necessary to modify some details -, and into Archive information, as part of the complete information set including the status.

Element / sub element	Definition	Creation	Update
ArchiveName	It identifies the archive within the PS Factory	Given by the Archive at Registration	No update
Person or Organisation Details	Various optional information on the Archive Organisation including address and contact information.	Given by the Archive at Registration	Possible on Archive request
ValidityDate	Date after which the Archive relationship is scheduled to expire	Optionally given by Archive at Registration. A default value may be given by PSO.	Possible on Archive request

Table 19 - Archive details set

6.3.2 Archive information

The Archive information set defined in Table 20, is specified in XML Schema [XPSOA] as **ArchiveInfoSetType**.

Element / sub element	Definition	Creation	Update
ArchiveDetails	See §6.3.1	Archive Registration	Partially by Archive
OpeningDate	Date of creation of the archive information record.	Generated by PSO at Archive Registration	No update
Status / Date	Current status of the relationship and status modification date	By PSO at Archive Registration as “Open”	By PSO
Orders	Identifiers (OrderNames) of Orders registered by the Archive	By PSO at first Order Registration	By PSO
ClosingInfo / ClosingDate + ClosingReason	Date when relationship is closed. Reason for closing	By PSO at end of activities	No update

Table 20 - Archive information set

6.3.3 Order definition

The Order definition given in Table 21 and specified as XML Schema [XPSOA] must be used for:

- Order registration
- Into the Order information, as part of the complete information set including the status.

Normally the Order definition cannot be updated, because processes already performed cannot be modified. A specific method may be optionally offered by PSO for extending the Validity date.

Element / sub element	Definition	Creation	Update
ArchiveName + OrderName	Identifies the Order within the PS-Factory	OrderName is given by the Archive at Order Registration	No update
ValidityDate	Date after which the Order is scheduled to expire	Optionally given by Archive at Registration. A default value may be given by PSO.	Possible on Archive request
Estimates	Optional Information about the periodicity of Batches, and indication of amount of batches, editorial objects, original media. See	By Archive at Order Registration (Validated by PSO)	No update
Process or ProcessProfile Name	Definition of required processes either making reference to a defined profile or by listing the require services, including service name, options, and parameters See §6.1 and §6.2.3	By Archive at Order Registration (Validated by PSO)	No update
Delivery	Optional requests about delivery See §Delivery options	By Archive at Order Registration (Validated by PSO)	No update
Notification	Optional requests of Notification Services for given Events. See §6.2.4	By Archive at Order Registration (Validated by PSO)	Possible on Archive request

Table 21 - Order definition set

Estimates

The planning of time and resources allocated by the Factory to an Archive Order is based on the information about the approximate quantities provided at the instantiation of the Order itself.

The Order information doesn't specify the precise amount of Batches, Editorial Objects, physical storages etc. because it is deemed that those quantities may change during the life of the Order. So the information defined here should provide quantities in terms of "expected amount" at the time of Order instantiation.

The information may be used in order to perform planning and allocation of resources.

The quantity information is not strictly required, even if useful, almost because it is difficult to guess which information the Archive is easily able to provide depending on its Inventory process.

In the Editorial domain, interesting quantities are the number of items and the total amount of hours[7], which determine the efforts required especially to the Documentation Units.

[7] In [D16.4] there was mention of the typology of Editorial Object. This has been removed as it would clearly impact the Order definition itself.

In the Material domain, the total amount of storages is important, together with the average storage duration, however this information should be associated to the storage type and to an indication about average storage conditions.

Sub elements are:

- Batch amount - approximate amount of Batches which will be related to the Order
- Batch submission cycle - the typical period of Batch submission, as expected at time of Order registration.
  - Daily – about once per day, at least 4 times per week.
  - Weekly – about once per week, at least 3 times per month
  - Monthly – about once per month
  - OnArchiveReady – at any time the Archive is ready, i.e. irregularly driven by Archive
  - OnFactoryIdle – after notification of completion of previous batch
- Editorial Object amount - approximate total amount of Editorial Objects related to the Order
- Editorial Object hours - approximate total duration of Editorial Objects related to the Order
- Storage amount - approximate total amount of Storages, of the given storage type (e.g. the Betacam Metal Small cassettes), related to the Order.
- Storage hours - approximate total duration of recorded Materials, for the given storage type, related to the Order.
- Storage condition - known approximate condition of Storages (e.g. good, bad, very bad), for the given storage type, related to the Order.

6.3.4 Order information

The Order Information Set defined in Table 22, is specified in XML Schema [XPSOA] as **OrderInfoSetType**.

Element / sub element	Definition	Creation	Update
OrderDefinition	See §6.3.3	Order Registration	No update
OpeningDate	Date of creation of the order information instance.	Generated by PSO at Order Registration	No update
Status / Date	Date when the status has been updated	By PSO at Order Registration	By PSO
Status	Current status of the Order	By PSO at Order Registration as “Open”	By PSO
Batches	Identifiers (BatchNames) of Batches registered by the Archive	By PSO at first Batch Registration	By PSO
ClosingInfo / ClosingDate + ClosingReason	Date when Order gets closed. Reason for closing	By PSO at end of activities	No update

Table 22 - Order information set

6.3.5 Batch registration

The Batch registration, specified in XML Schema [XPSOA] and defined in Table 23, simply assigns a list of Editorial Objects, identified by their IDs, to an Order.

Element / sub element	Definition	Creation	Update
ArchiveName + Order Name + BatchName	Identifies the Batch within the PS-Factory	BatchName is given by the Archive at Batch submission	No update
EditorialObjectIds	Identifiers of the Editorial Objects of the Batch	Given by Archive at Batch submission	No update

Table 23 - Batch registration set

6.3.6 Batch information

The Batch Information Set defined in Table 24 is specified in XML Schema [XPSOA] as **BatchInfoSetType**.

Element / sub element	Definition	Creation	Update
ArchiveName + Order Name + BatchName	Identifies the Batch within the PS-Factory	BatchName is given by the Archive at Order Registration	No update
AcceptanceDate	Date of Batch submission	Generated by PSO at Batch Registration	No update
Status / Date	Date when the status of the Batch has been updated	By PSO	By PSO
Status	Current status of Batch	By PSO at Batch submission as “Open”	By PSO
Status / Expected CompletionDate	Date by which the Batch is expected to be completed	Given by PSO after the Batch submission	By PSO
EditorialObjects / EditorialObjectId	Identifiers of the Editorial Objects of the Batch	Given by Archive at Batch submission	No update
EditorialObjects / EditorialObjectId / Status / Date	Date when the status of the Editorial Object has been updated	By PSO	By PSO
EditorialObjects / EditorialObjectId / Status	Current status of Editorial Object	By PSO	By PSO
EditorialObjects / EditorialObjectId / Note	An optional note explaining the reason for the status of Editorial Object	By PSO	By PSO
ClosingInfo / ClosingDate + ClosingReason	Date when Batch gets Closed. Reason for closing	By PSO at end of activities	No update

Table 24 - Batch information set

6.3.7 Web services

The functionalities of the web services offered by the PSO to the Archive are:

- Archive registration
- Update of Archive Details
- Inquire about Archive Information
- Get Knowledge Base on Archive – Returned: access to RDF file containing the whole KB gathered for the given Archive. Optionally the provided RDF may contain only the update of the KB after a given date and/or only the information related to the entities identified by the given list of URIs
- Delete Archive - Asks to the PS Factory to remove all records about Archive, possible only if the status is *Closed*.
- Register Order – Defines a new Order for the given Archive
- Update Order – extension of the *ValidityDate* and or modification of *Notification* details
- Inquire about Order Information
- Close Order - Asks to the PS Factory to STOP all processes concerning the Order
- Submit Batch – Associates a list of Editorial Objects to an Order and asks to start the process on them
- Inquire on Batch – Returned: summarisation of status for the Batch elements
- Get Batch – Asks to the PS Factory to provide the Editorial Object Documents and the Material Access Documents of the whole Batch into the given folder. In practice it's a short cut to making a sequence of calls for all the Editorial Objects
- Close Batch - Asks to the PS Factory to STOP all processes concerning the Batch
- Clear Batch - Enables the PS-Factory to delete all records about the Batch instance, including all Editorial Objects and associated Material Sources. Possible only if the status is *Closed*.
- Insert Materials
- Editorial Object Registration – includes also registration of Editorial Collections
- Get Materials
- Get Editorial Object – (or Editorial Collection)

The complete definition of these web services is provided by the WSDL [WPSOA]

6.4 The PrestoSpace Units and Interfaces

6.4.1 Interface PSO-Preservation

Preservation Batch Submission

The PSO submits works to the Preservation Unit by sending a PreservationBatch document, which is an XML document specified by the XML Schema available at [XPSOU], to a web service provided by the Preservation Unit.

The PreservationBatch document basically contains:

- Service options and parameters – See §6.1.1
- Identification of Materials and Storages (original media) –
- Reference to the Shipments of Storages, occurred from the Archive

Management of newly digitised master material

On success of the digitisation process the Preservation Unit submit the newly created Material to the PSO by sending a Material Exchange Document to a web service provided by the PSO. See §6.2.5

The format of the Material Exchange Document is defined by the XML Schema [XMAD Element MaterialExchange].

The Digitisation reporting information must be included within the Material Exchange Document as Derivation Information.

Notifications

The PSO exposes a web service for receiving notifications from the Preservation Unit about Material instances. See §6.2.4

Notification messages include:

- Shipment Reception – Confirmation of successful reception of a shipment
- Shipment Error - The referenced shipment wasn't received
- Missing Storage – The referenced storage was not found
- Digitisation failure – The process of digitisation failed for given reason
- Progress status (optional) -

Return of original media

The Preservation Unit must operate the return of original media to the Archive according to the related option selected by the Archive.

Return Shipments must be communicated to the PSO through the Material Exchange mechanism.

Web services

The list of web services involved with this interface is:

- Provided by PSO to Preservation – WSDL available at [WPSOU]
  - Insert Materials
  - Notify
- Provided by Preservation to PSO – WSDL available at [WPRE]
  - Submit Preservation Batch

6.4.2 Interface PSO-Restoration

The interface between PSO and Restoration is defined exactly as the same as that of PSO and Documentation (see §6.4.3), as the main difference is that the required service is simpler to be defined and there is no issue about transfer of original media as for Preservation. The Restoration working unit is the Editorial Object.

Besides, the internal architecture of the Restoration Unit is not described here, but it is meant to use and offers the same web-services as for the Documentation Unit.

6.4.3 Interface PSO-Documentation

Job Insertion

The PSO must submit Jobs, in the form of a defined XML Document (XML Schema [XPSOU]) to a Web service offered by the Documentation Unit, specifically the Documentation-CorePlatform-WorkFlow subsystem.

Each Job pertains to a single Editorial Object and indicates the list of activities (processes) that are requested to the Documentation Unit. For each activity, options and parameters can be specified. The activities will be generally performed in the given sequence.

The PSO can be notified on the completion or failure of any activity.

Editorial Object and Material Management

The processors working under the Documentation Unit (GAMP – Generic Activity MAD Processor) need to access the Editorial Object and the Material sources on the EMS with the mechanism defined in §6.2.2 and §6.2.5.

The requests are centrally managed by the Documentation Core Platform that is in charge of the interface with the PSO. It must be remarked that the PSO can in any time have access to the last updated version of each Editorial Object.

The general work-flow for a single processor is:

- Get Job from the Documentation work flow
- Get Editorial Object from EMS
- Identify and get Material relevant with the task (optional) from EMS
- Actual work according to job options
- Insert newly created Material (for instance key frames) into EMS
- Update Editorial Object into EMS
- Notify completion to Documentation work flow.

Web services

The list of web services involved with this interface is:

- Provided by PSO to Documentation – WSDL available at [WPSOU]
  - Get Materials
  - Insert Materials
  - CheckOut Editorial Object – it can be either with or without lock
  - CheckIn Editorial Object
  - Notify
- Provided by Documentation to PSO – WSDL available at [WDOC]
  - Insert Job
  - Inquire Job status

6.5 Interfaces PSO-other services

Some PrestoSpace Services, described in §6.1, and notably Defects & Quality Analysis, Content Analysis, and Transcoding, are not strictly associated in this document to any PrestoSpace Unit (see §4.3.1).

Actually for each of them there are a few options of Architecture implementation within the PrestoSpace Factory.

- Defects & Quality Analysis service could be provided by:
  - the Restoration Unit
  - an independent sub-system, in practice a simplified Unit
- Content Analysis service could be provided by:
  - the Restoration Unit
  - the Documentation Unit, as GAMP
  - an independent sub-system, in practice a simplified Unit
- Transcoding service could be provided by:
  - an independent sub-system, in practice a simplified Unit
  - the Documentation Unit, as GAMP.

In order to simplify the PSO implementation, it is however recommended that any other PrestoSpace Factory sub-system providing Services has a PSO interface defined exactly as the same as that of PSO and Documentation (see §6.4.3).

7 Glossary

Term	Definition
Broadcast Quality	The material instances the quality of which is appropriate for publication through current audiovisual publication media.
Browsing Quality	The material instances the quality of which is sufficient to appreciate and recognise the content, albeit some impairments might be quite perceptible.
Carrier	Organisation or Person responsible for transferring removable storages, especially original media, between two locations.
CVS	Concurrent Version System, a system able to manage different version of the same artefact, keeping track on changes. It provide the methods for checking-out and checking-in documents.
Doc/CA Quality	The material instances, with digital sources, the quality of which is appropriate for documentation and content analysis processes. With a medium encoding quality, it doesn't show visible encoding defects. About the same as Broadcast quality.
Editorial Object (EDOB)	Editorial Object. An audiovisual work, or an editorial constituent part of it, from the perspective of its artistic, communicable, and expressive aspects. It is identified as the PrestoSpace Factory Working Unit.
EMS	Essence and Metadata Storage. The system which is responsible for storing the essence and metadata within the PrestoSpace Factory sharing these resources between the several Units, according a defined interface. The system is under the responsibility of PSO.
EOD	Editorial Object Document. The XML Document defined in [XMAD] including all the information related to it and the needed references to external data such as material sources.
Essence	Basically the same as Material Source. A subtle distinction often indicates as Essence only digital Material Sources.
GAMP	Generic Activity MAD Processor. A sub-system of the Documentation Unit performing a specific task. It acts as a client of the CorePlatform of the Documentation Unit.
KB	Knowledge Base. A special kind of database for knowledge management providing the means for computerised collection, organisation, and retrieval of knowledge, by using an ontology to specify its structure.
Master Quality	The Material instances with the highest quality, used as reference, and from which any equivalent or lower quality may be derived. Digital master quality, highest encoding quality, mathematically or visually lossless over multiple generations.
Material	An audiovisual entity which can be obtained by the playback of some audio/video recording. Material properties are the duration, the type and number of tracks, the video aspect ratio, and other one. Not to be confused with Material Source.
Material Access Document	The XML document providing the available access information enabling an exchange of Material Sources for a list of Material instances. This document is provided either as a reply to a getMaterial request or as the parameter of insertMaterial request. For each Material instance, identified with the UMID, a list of access mechanism options (at least one) is given. The access information shall include protocols, ports, filename, file size, file checksum, and other details. Account information such as user login and passwords are NOT included.
Material Source	The data which are the coded representation of audio-video recording. Material can be obtained by a Material Source by means of a decoding process. Examples of properties of Material Source (esp. digital) are the encoding scheme and the number of bits.

Migration metadata	Metadata that are generated during the digitisation process (Derivation Information). These include reports automatically generated by playback equipment and measurement tools, as well as any other input filled in by operators in the Preservation Unit.
PSO	The component of the PrestoSpace Factory responsible for interfacing towards the Archives, for accounting all information about the processes on working units, and for dispatching activity tasks to the various Units. The PSO provides the interfaces for accessing the Factory, based on web services for machine to machine communication and web applications for the human being users. The PSO has an internal work flow engine in order to process the jobs to be done and for resolving their dependencies.
Realisation	The relationship accounting how an Editorial Object is realised by one or more Material instances.
Shipment	The delivery of a bunch of removable storages, especially original media, either from Archive to the PrestoSpace Factory or vice-versa.
Storage	A container for a Material Source or for other storages (in which case is also called Bundle). Either a computer file or a physical removable device, such as a tape.
Transcoding	Process of deriving a Material instance from another one by decoding the Material Source and producing a new one with a different encoder. The process typically affects the properties of both Source and Material.
Transwrapping	Process aiming to change the file format of a Material Source container, without modifying the Material Source itself. The Essence is the same, while how it is organised within the container may change. Therefore the quality of Material is the same.
UMID	The UMID is the Material Identifier as defined by SMPTE330M. In textual documents, as XML, it must be recorded as a string starting by “0x” and giving then the UMID value in hexadecimal (each byte is coded with a couple of characters in the range [0-9,A-F]). The UMID has 32 bytes with the a possible extension to total 64 bytes. The UMID has to be generated and provided by the entity emitting the Material instance.
Working Unit	The item on which a given process is focused. From the PrestoSpace Factory perspective the Working Unit is the Editorial Object. Other components of the PrestoSpace Factory may have different working units.

8 Annex A: XML Schemas of PSO interfaces

8.1 PSO-Archive interface XML Schema [XPSOA]

Table 25 contains the full text of file **psoarc.xsd**, which is the XML Schema defining the formats of document exchanched on the PSO-Archive interface. It is an extension of MAD and CorePlatformDef Schemas (XMAD)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsd:schema xmlns:pmeta="urn:ebu:schema:pmeta:2007" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.prestospace.org/MAD/PSO/Archives/xml-schema" xmlns:ns1="urn:mpeg:mpeg7:schema:2001"
xmlns:ns2="http://www.prestospace.org/MAD/CorePlatform/xml-schema"
targetNamespace="http://www.prestospace.org/MAD/PSO/Archives/xml-schema" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="2008-01-31">

<xsd:import namespace="urn:ebu:schema:pmeta:2007"
schemaLocation="PMETA2.0/EBU_PMETAv020001p/EBU_PMETAv020001p.xsd"/>
<xsd:import namespace="urn:mpeg:mpeg7:schema:2001" schemaLocation="mpeg7/schema/davp-2005.xsd"/>
<xsd:import namespace=http://www.prestospace.org/MAD/CorePlatform/xml-schema
schemaLocation="CorePlatformDef.xsd"/>

<!-- simple Types -->
<xsd:simpleType name="StatusValueType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="Open"/><xsd:enumeration value="Active"/>
<xsd:enumeration value="Finalising"/><xsd:enumeration value="Closed"/>
</xsd:restriction></xsd:simpleType>

<xsd:simpleType name="ClosingReasonType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="Completed"/><xsd:enumeration value="Expired"/><xsd:enumeration value="Cancelled"/>
</xsd:restriction></xsd:simpleType>

<xsd:simpleType name="EventType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="OrderStatus"/><xsd:enumeration value="BatchStatus"/>
<xsd:enumeration value="BatchCompletion"/><xsd:enumeration value="BatchProblems"/>
<xsd:enumeration value="NewMaterialRealisationAvailable"/><xsd:enumeration value="UnspecifiedProblems"/>
</xsd:restriction></xsd:simpleType>

<xsd:simpleType name="notificationMechanismType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="webservice"/><xsd:enumeration value="email"/><xsd:enumeration value="sms"/>
<xsd:enumeration value="rss"/><xsd:enumeration value="phone"/><xsd:enumeration value="fax"/>
<xsd:enumeration value="mail"/><xsd:enumeration value="rpc"/><xsd:enumeration value="other"/>
</xsd:restriction></xsd:simpleType>

<xsd:simpleType name="metadataFormatType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="PrestoSpace EOD"/><xsd:enumeration value="P_META"/><xsd:enumeration value="Mpeg7"/>
<xsd:enumeration value="Dublin Core"/><xsd:enumeration value="SMPTE DMS1"/><xsd:enumeration value="other"/>
</xsd:restriction></xsd:simpleType>

<xsd:simpleType name="EdobStatusType"><xsd:restriction base="xsd:string">
<xsd:enumeration value="Acceptance pending"/><xsd:enumeration value="Accepted"/>
<xsd:enumeration value="Refused"/><xsd:enumeration value="Working"/><xsd:enumeration value="Stalled"/>
<xsd:enumeration value="Unsolvable failure"/><xsd:enumeration value="Partial failure"/>
<xsd:enumeration value="Completed"/>
</xsd:restriction></xsd:simpleType>
```

```
<!-- complex Types -->
<xsd:complexType name="ArchiveInfoSetType"><xsd:complexContent><xsd:extension
base="ArchiveDetailsType"><xsd:sequence><xsd:element name="OpeningDate" type="xsd:date"/><xsd:element
name="Status" type="StatusType"/><xsd:element name="OrderName" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ClosingInfo" type="ClosingInfoType" minOccurs="0"/>
</xsd:sequence></xsd:extension></xsd:complexContent>  </xsd:complexType>

<xsd:complexType name="ArchiveDetailsType"><xsd:sequence><xsd:element name="ArchiveName" type="xsd:string"/>
<xsd:choice minOccurs="0" maxOccurs="2"><xsd:element ref="pmeta:OrganisationDetails"/><xsd:element
ref="pmeta:PersonDetails"/></xsd:choice><xsd:element name="ValidityDate" type="xsd:date" minOccurs="0"/>
</xsd:sequence></xsd:complexType>

<xsd:complexType name="ClosingInfoType"><xsd:sequence><xsd:element name="ClosingDate"
type="xsd:date"/><xsd:element name="ClosingReason" type="ClosingReasonType"/></xsd:sequence></xsd:complexType>

<xsd:complexType name="StatusType"><xsd:simpleContent><xsd:extension base="StatusValueType">
<xsd:attribute name="Date" type="xsd:date" use="required"/></xsd:extension></xsd:simpleContent></xsd:complexType>

<xsd:complexType name="OrderDefinitionType"><xsd:sequence>
<xsd:element name="ArchiveName" type="xsd:string"/><xsd:element name="OrderName" type="xsd:string"/>
<xsd:element name="ValidityDate" type="xsd:date" minOccurs="0"/><xsd:element name="Estimates" minOccurs="0">
<xsd:complexType><xsd:sequence>
<xsd:element name="BatchAmount" type="xsd:positiveInteger" minOccurs="0"/>
<xsd:element name="BatchSubmissionCycle" minOccurs="0"><xsd:simpleType><xsd:restriction base="xsd:string">
<xsd:enumeration value="Daily"/><xsd:enumeration value="Weekly"/><xsd:enumeration value="Monthly"/>
<xsd:enumeration value="OnFactoryIdle"/><xsd:enumeration value="OnArchiveReady"/>
</xsd:restriction></xsd:simpleType></xsd:element>
<xsd:element name="EditorialObjectAmount" type="xsd:positiveInteger" minOccurs="0"/>
<xsd:element name="EditorialObjectHours" type="xsd:positiveInteger" minOccurs="0"/>
<xsd:element name="Storages" minOccurs="0" maxOccurs="unbounded"><xsd:complexType>
<xsd:sequence><xsd:element name="StorageAmount" type="xsd:positiveInteger" minOccurs="0"/>
<xsd:element name="StorageHours" type="xsd:positiveInteger" minOccurs="0"/>
<xsd:element name="StorageConditions" type="xsd:string" minOccurs="0"/></xsd:sequence>
<xsd:attribute name="storageType" type="xsd:anyURI"/><!--reference:  ebu:StorageMediaTypeCode -->
</xsd:complexType></xsd:element></xsd:sequence></xsd:complexType></xsd:element>
<xsd:choice><xsd:element name="ProcessProfileName" type="xsd:string"/><xsd:element name="Process"
type="ProcessProfileType"/></xsd:choice>
<xsd:element name="Delivery" minOccurs="0"><xsd:complexType><xsd:all>
<xsd:element name="ExportMetadataAs" type="metadataFormatType" minOccurs="0"/>
<xsd:element name="PublicationService" type="ServiceType" minOccurs="0"/>
</xsd:all></xsd:complexType></xsd:element>
<xsd:element name="Notification" minOccurs="0" maxOccurs="unbounded"><xsd:complexType><xsd:sequence>
<xsd:element name="EventType" type="EventType"/><xsd:element name="Destination" type="xsd:string"/>
<xsd:element name="NotificationMechanism" type="notificationMechanismType"/>
</xsd:sequence> </xsd:complexType></xsd:element>
</xsd:sequence></xsd:complexType>

<xsd:complexType name="ProcessProfileType"><xsd:sequence>
<xsd:element name="Service" type="ServiceType" maxOccurs="unbounded"/></xsd:sequence>
<xsd:attribute name="name" type="xsd:string"/> <xsd:attribute name="version" type="xsd:date"/>
</xsd:complexType>

<xsd:complexType name="OrderInfoSetType"><xsd:complexContent><xsd:extension base="OrderDefinitionType">
<xsd:sequence><xsd:element name="OpeningDate" type="xsd:date"/><xsd:element name="Status" type="StatusType"/>
<xsd:element name="BatchName" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ClosingInfo" type="ClosingInfoType" minOccurs="0"/>
</xsd:sequence></xsd:extension></xsd:complexContent></xsd:complexType>
```

```
<xsd:complexType name="ServiceType"><xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="Option"><xsd:complexType><xsd:attribute name="Name" type="xsd:string"
use="required"/></xsd:complexType></xsd:element><xsd:element name="Parameter"><xsd:complexType><xsd:attribute
name="Name" type="xsd:string" use="required"/><xsd:attribute name="Value" type="xsd:string"
use="required"/></xsd:complexType></xsd:element></xsd:choice><xsd:attribute name="ServiceName" type="xsd:string"
use="required"/></xsd:complexType>

<xsd:complexType name="BatchSubmissionType"><xsd:sequence>
<xsd:element name="ArchiveName" type="xsd:string"/><xsd:element name="OrderName" type="xsd:string"/>
<xsd:element name="BatchName" type="xsd:string"/><xsd:element name="EditorialObjectID" type="xsd:string"
maxOccurs="unbounded"/></xsd:sequence></xsd:complexType>

<xsd:complexType name="BatchInfoSetType"><xsd:sequence>
<xsd:element name="ArchiveName" type="xsd:string"/><xsd:element name="OrderName" type="xsd:string"/>
<xsd:element name="BatchName" type="xsd:string"/>
<xsd:element name="Status"><xsd:complexType><xsd:sequence maxOccurs="unbounded"><xsd:element
name="EditorialObjectInfo"><xsd:complexType><xsd:sequence minOccurs="0"><xsd:element name="Note"
type="xsd:string"/></xsd:sequence><xsd:attribute name="EditorialObjectID" type="xsd:string"
use="required"/><xsd:attribute name="StatusDate" type="xsd:date" use="required"/><xsd:attribute name="Status"
type="EdobStatusType" use="required"/> </xsd:complexType></xsd:element></xsd:sequence><xsd:attribute
name="AcceptanceDate" type="xsd:date" use="required"/> <xsd:attribute name="StatusDate" type="xsd:date"
use="required"/> <xsd:attribute name="ExpectedCompletionDate" type="xsd:date"
use="optional"/></xsd:complexType></xsd:element>
<xsd:element name="ClosingInfo" type="ClosingInfoType" minOccurs="0"/>
</xsd:sequence></xsd:complexType>

<!-- global elements -->
<xsd:element name="ArchiveRegistration"><xsd:complexType><xsd:complexContent>
<xsd:extension base="ArchiveDetailsType"><xsd:attribute name="schemaVersion" type="xsd:string"
use="required"/></xsd:extension>
</xsd:complexContent></xsd:complexType></xsd:element>

<xsd:element name="ArchiveInformation"><xsd:complexType><xsd:complexContent>
<xsd:extension base="ArchiveInfoSetType"><xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:extension></xsd:complexContent></xsd:complexType></xsd:element>

<xsd:element name="OrderRegistration"><xsd:complexType><xsd:complexContent>
<xsd:extension base="OrderDefinitionType"><xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:extension></xsd:complexContent></xsd:complexType></xsd:element>

<xsd:element name="OrderInformation"><xsd:complexType><xsd:complexContent>
<xsd:extension base="OrderInfoSetType"><xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:extension></xsd:complexContent></xsd:complexType></xsd:element>


<xsd:element name="BatchSubmission"><xsd:complexType><xsd:complexContent>
<xsd:extension base="BatchSubmissionType"><xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:extension></xsd:complexContent></xsd:complexType></xsd:element>

<xsd:element name="BatchInformation"><xsd:complexType><xsd:complexContent>
<xsd:extension base="BatchInfoSetType"><xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:extension></xsd:complexContent></xsd:complexType></xsd:element>


</xsd:schema>
```

Table 25 – PSO-Archive XML Schema document

8.2 PSO-Units interfaces XML Schema [XPSOU]

Table 26 contains the full text of file **psounits.xsd**, which is the XML Schema defining the formats of document exchanched on the PSO-Units interface. It is an extension of MAD and CorePlatformDef Schemas (XMAD)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsd:schema xmlns="http://www.prestospace.org/MAD/DataModel/xml-schema"
xmlns:wu="http://www.prestospace.org/MAD/CorePlatform/xml-schema"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:pso="http://www.prestospace.org/MAD/PSO/xml-schema"
xmlns:pmeta="urn:ebu:schema:pmeta:2007" xmlns:ns1="urn:mpeg:mpeg7:schema:2001"
xmlns:ns2="http://www.ebu.ch/P_META/ExternalReferenceData"
targetNamespace="http://www.prestospace.org/MAD/PSO/xml-schema" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="2008-01-31">

<xsd:import namespace="http://www.prestospace.org/MAD/DataModel/xml-schema" schemaLocation="MAD.xsd"/>
<xsd:import namespace="urn:ebu:schema:pmeta:2007"
schemaLocation="PMETA2.0/EBU_PMETAv020001p/EBU_PMETAv020001p.xsd"/>
<xsd:import namespace="urn:mpeg:mpeg7:schema:2001" schemaLocation="mpeg7/schema/davp-2005.xsd"/>
<xsd:import namespace=http://www.prestospace.org/MAD/CorePlatform/xml-schema
schemaLocation="CorePlatformDef.xsd"/>

<xsd:element name="preservationBatch"><xsd:complexType>
<xsd:sequence><xsd:element name="serviceInput" minOccurs="0">
<xsd:complexType><xsd:sequence><xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="option"><xsd:complexType><xsd:attribute name="name" type="xsd:string"
use="required"/></xsd:complexType></xsd:element>
<xsd:element name="parameter"><xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="value" type="xsd:anySimpleType" use="required"/>
</xsd:complexType></xsd:element>
</xsd:choice></xsd:sequence></xsd:complexType></xsd:element>
<xsd:element ref="wu:MaterialExchange"/></xsd:sequence>
<xsd:attribute name="id" type="xsd:string" use="required"/>
<xsd:attribute name="schemaVersion" type="xsd:string" use="required"/>
</xsd:complexType></xsd:element>

<xsd:element name="job"><xsd:complexType>
<xsd:sequence><xsd:element name="jobaplist"><xsd:complexType>
<xsd:sequence><xsd:element name="jobap" maxOccurs="unbounded"><xsd:complexType>
<xsd:attribute name="activity" use="required"/><xsd:attribute name="id"/>
<xsd:attribute name="queue" use="required"/><xsd:attribute name="status"/>
</xsd:complexType></xsd:element>
</xsd:sequence></xsd:complexType></xsd:element>
</xsd:sequence><xsd:attribute name="edob" use="required"/><xsd:attribute name="id"/>
</xsd:complexType></xsd:element>

</xsd:schema>
```

Table 26 - PSO-Archive XML Schema document