www.forgetit-project.eu

ForgetIT

Concise Preservation by Combining Managed Forgetting and Contextualized Remembering

Grant Agreement No. 600826

Deliverable D8.5

Work-package	WP8: The Preserve-or-Forget Reference Model and Framework
Deliverable	D8.5: The Preserve-or-Forget Reference Model - Final Model
Deliverable Leader	Francesco Gallo (EURIX)
Quality Assessor	Gurkan Gur (TT)
Dissemination level	PU
Delivery date in Annex I	M36 (January 2016)
Actual delivery date	31 March 2016
Revisions	13
Status	Final Version
Keywords	Preserve-or-Forget Reference Model, Digital Preservation, Standards

Revision History

Version	Major changes	Authors
0.01	Document skeleton, ToC created	EURIX
0.02	Added introduction, preliminary information model entities	LUH, EURIX, DFKI, LTU
0.03	Added model foundations	LUH, DFKI, LTU
0.04	Updated information model, preservation standards	LUH, EURIX
0.05	Added functional model	LUH, EURIX, DFKI, LTU
0.06	Added mapping to PoF architecture	EURIX
0.07	Added active system extension	DFKI, LTU
0.08	Reviewed functional and information model	LUH
0.09	Reviewed references and glossary, created executive summary	EURIX
0.10	First complete version for project wide review	EURIX
0.11	Version for internal QA	EURIX
0.12	Updated version after internal QA	EURIX
1.00	Submitted version	EURIX

List of Authors

Partner Acronym	Authors
LUH	Claudia Niederee
LTU	Ingemar Andersson, Jörgen Nilsson
DFKI	Heiko Maus
EURIX	Walter Allasia, Francesco Gallo

Table of Contents

Executive Summary 7

1 Introduction 8

1.1 Purpose of the PoF Reference Model 8

1.2 Target Audience 10

1.3 Structure of the Deliverable 10

2 Foundations of the PoF Reference Model 11

2.1 Integrative: Bridging the Gap between Information and Preservation Management 12

2.2 Value-driven: Acting upon Short-term and Long-term Information Value . . 13

2.3 Brain-inspired: What can we Learn from Human Forgetting and Remembering? 14

2.4 Forgetful: Focus on the Important Things 16

2.5 Evolution-aware: Embracing the Long-term Perspective 17

3 Functional Model 19

3.1 Layers, Workflows and Functional Entities 19

3.2 Core Layer 21

3.2.1 Basic Preservation Preparation 22

3.2.2 Basic Re-activation 22

3.3 Remember & Forget Layer 23

3.3.1 Preservation Preparation 23

3.3.2 Re-activation 26

3.3.3 Metadata and Preservation Contract 27

3.4 Evolution Layer 27

3.4.1 Situation Change 28

3.4.2 Setting Change 30

3.4.3 System Change 32

4 Information Model **35**
4.1 Situations as Units of Preservation 35
4.2 Core Information Model: User Perspective 37
4.3 Extended Information Model: Middleware Perspective 40

5 Mapping to ForgetIT Architecture **49**
5.1 PoF Framework Architecture 49
5.2 Relationship with PoF Architecture 49
5.3 Model Implementation in PoF Architecture 51

6 Mapping to Digital Preservation Standards **57**

7 Information Management Systems Extensions **61**
7.1 Extensions of the Active System 61
7.1.1 Supporting Information Exchanges 61
7.1.2 Information Value Evidences 62
7.1.3 Supporting Contextualization 62
7.2 Creating Benefit in the Active System 63
7.2.1 Forgetful Information Presentation 63
7.2.2 Forgetful and Archive-aware Search 64
7.2.3 Creating Awareness for Content Value 64
7.2.4 Content Value Assessment in the Active System 64
7.3 Preservation Strategies 65

8 Conclusion **67**
8.1 Assessment of Performance Indicators 68
8.2 Lessons Learned 69
8.3 Vision for the Future 70

9 References **71**

Glossary **76**

Executive summary

The Reference Model aims to encapsulate the core ideas of the ForgetIT approach into a re-usable model. Therefore, it is inspired by the core principles of the Preserve-or-Forget approach: synergetic preservation, managed forgetting and contextualized remembering.This document describes the final version of the Preserve-or-Forget (PoF) Reference Model and provides a guideline for the future adoption of ForgetIT concepts beyond the project scope.

Based on this, we have identified five characteristics for a PoF Reference Model, which considers Active System and Digital Preservation System (DPS) as a joint ecosystem: *integrative*, *value-driven*, *brain-inspired*, *forgetful* and *evolution-aware*. Those characteristics have driven the design of the functional part of the PoF Reference Model and of the associated Information Model.

The Functional Model is made up of three layers: *Core Layer*, *Remember & Forget Layer* and *Evolution Layer*. For each layer we describe the main functional entities and the representative workflows. The Information Model describes the main entities first from the perspective of the active system (user perspective) and then from the broader perspective of middlelayer, which mediates between actives system and DPS. For taking into account practical aspects and experiences from the projct, the Information Model has been defined in parallel with the development of the PoF Framework.

This deliverable is not limited to the PoF Reference Model. It also presents mappings and interactions of the model with other building blocks of an intelligent preservation solution. Firstly, the relationship with OAIS and other relevant digital preservation standards is discussed Moreover, we relate the PoF Framework developed in WP8 to the Reference Model showing its compliance to the model. The role of the Information Model entities in the implementation is also drafted here, further details are available in deliverable D8.6. Finally, it also discusses the extensions required in any Active System in order to follow the approach outlined by the PoF Reference Model.

Compared to the first version of the model described in deliverable D8.2, the final version includes an Information Model representing the main entities involved with the content preservation, re-activation, contextualization and evolution. Moreover, we also extended the functional part of the model, mainly for what concerns the evolution layer and the description of the existing functional entities and workflows. Mapping to other models has been extended from OAIS to further standards relevant in digital preservation, in order to better position the model in the current digital preservation research.

We decided to include in this document also other parts from D8.2 which were not affected by the model update, in order to provide a self-contained document.

1 Introduction

In this deliverable we present a model for the novel approach to intelligent preservation management defined by the ForgetIT project. The model aims to encapsulate the core ideas of the ForgetIT approach into a re-usable model. In the following we refer to this model as Preserve-or-Forget (PoF) Reference Model.

The PoF Reference Model presented in this deliverable is based on the work done for the PoF Middleware in WP8, the components and especially the concepts developed in WP3-WP7 as well as the foundational work in WP2 and WP5. Furthermore, the model builds upon the lessons learned in the ForgetIT project - including the experiences from the work with the interdisciplinary partners and with the application partners.

Since ForgetIT stresses the smooth interaction between information management and preservation management, the PoF Reference Model described here pays special attention to the functionality which bridges between the Active System and the Digital Preservation System (DPS) (see deliverable D8.1 [Gallo et al., 2013]). This includes the selection of content for preservation, the transfer of content between the systems, contextualization for easing long-term interpretation, processing during preservation time in reaction to changes, as well as access to the joint information space populated both by preserved content and content in active use.

The first version of the model has been described in deliverable D8.2 [Gallo et al., 2015b]. The PoF Reference Model has been revised and refined during the project lifetime. The final version of the model is presented in this document. Compared to the first version, the final model includes an Information Model (Section 4) based on the core data entities from different perspectives; moreover the functional part (Section 3) has been updated and completed. In this process, the Evolution Layer has been extended and now covers change workflows on different levels. Finally, the interoperability with other digital preservation standards has been outlined.

For making the document on the Reference Model self-contained, the existing (and revised) content from the first release of the Reference Model has also been included into this deliverable.

1.1 Purpose of the PoF Reference Model

The idea of the ForgetIT project is to follow a forgetful, focused approach to digital preservation, which is inspired by human forgetting and remembering. Its goal is to ease the adoption of preservation technology especially in the personal and organizational context and to ensure that important content is kept safe, useful, and understandable on the long run. For this purpose, concepts, technologies, and an entire framework (the PoF Framework) have been developed in the ForgetIT project.

The framework has been implemented in the form of the PoF Middleware (see deliverable

D8.6 [Gallo et al., 2016]). While the architecture and the implemented PoF Middleware ease adoption on the technology level, the PoF Reference Model leverages the ideas developed in the ForgetIT project to a conceptual level: required conceptual functionalities, core processes, and relevant concepts are collected in a systematic way and are related to each other.

The PoF Reference Model especially targets personal as well as organizational preservation settings, which are not covered by legal regulations such as deposit laws. The focus is not on supporting memory institutions, although parts of the ForgetIT ideas (e.g., contextualization) might be applicable in this area as well. The selected settings (personal and non-mandatory organizational preservation) can especially benefit from the ForgetIT approach, since a) there is a big gap to preservation adoption in those fields, b) there are explicit preservation choices to be made, and c) there is a need for automation of the processes, in order to reduce the amount of investment required for preservation.

Similar to the OAIS Reference Model [CCSDS, 2012], the PoF Reference Model defines the terminology and concepts for the approach, which can be used as a basis for the implementation of the ForgetIT approach as well as for the further discussion and development of a forgetful approach to preservation.

Recently, new models have been proposed in the digital preservation community, with the purpose of extending OAIS model. As ForgetIT, the model the University of California Curation Center (UC3) also propagates the idea that an interdisciplinary approach is useful for this purpose. In their building of an abstract model and curation foundations [University of California Curation Center (UC3), 2016] they, for example, information science, and semiotic theory, considering the content's underlying abstract cognitive meaning or emotional effect and evolution through time. The PoF Reference Model foundations are described in Section 2, where we identify five main characteristics, including the brain-inspired and evolution-aware features of the model.

In order to support a smooth transition between active information use in the respective information management system (environment), which we call Active System, and the DPS, in the PoF Reference Model we are considering active information use and preservation as part of such a *joint ecosystem*, which stresses the smooth transitions and the synergetic interactions rather than the system borders. This also is a core distinction from the OAIS Reference Model [CCSDS, 2012], which is restricted to the DPS.

It is worth noting that the ForgetIT model does not intend to replace the OAIS model and other preservation standards. As mentioned above, it has a different focus than OAIS and can interact with OAIS compatible approaches, as outlined in Section 6, where we extend our analysis to other relevant standards adopted by the digital preservation community.

In this deliverable, we also present a possible mapping of this joint ecosystem onto three separate systems, i.e. the Active System (in the form of adapters and/or system extensions), the DPS (typically OAIS based) and the PoF Middleware (which couples both systems in a flexible way)and we discuss the extensions of the Active System required by the model.

For the representation of the functional part (layers, workflows and entities) we provide simple graphical representations. A more formal approach has been used for the information model, using the Unified Modeling Language (UML) notation, in order to better support the design and implementation of PoF compatible systems.

1.2 Target Audience

The target audience of this deliverable are practitioners and developers in the area of content and information management, who consider to get started with content preservation. The deliverable is also targeting practitioners, developers, and researchers in the area of preservation management.

In order to serve this diverse audience, the deliverable considers the presented PoF Reference Model from different perspectives. In addition to the presentation of the model itself in terms of functional model and information model, the deliverable also presents a mapping to the OAIS model and the interaction of the PoF Reference Model with a content management system. Furthermore, the presented mapping to the PoF architecture is also meant to ease adoption for system developers.

1.3 Structure of the Deliverable

In Section 2, we discuss the foundations of the PoF Reference Model, covering the characteristics of the model as well as the requirements, which influenced its development. We also describe the model in the context of an integrated information and preservation management system, combined as a joint ecosystem.

The PoF Reference Model itself is described in two subsequent sections: the functional part in Section 3 and the information part in Section 4, respectively.

In Section 5, we relate the PoF Reference Model to the reference architecture of the PoF Framework, mapping the joint ecosystem of building blocks from the reference model to an architecture with three layers (Active System, DPS, PoF Middleware) [Gallo et al., 2016].

In the second part of the deliverable, we relate the PoF Reference Model to existing models and approaches, in order to ease adoption. This includes a discussion on how to map the relevant parts of the model to an OAIS based DPS and to other relevant digital preservation standards (Section 6. Furthermore, we required extensions to the Active System for becoming part of an information and preservation management ecosystem (Section 7).

Finally, we provide a summary of the main insights, with ideas for future work and an assessment of the results compared to the success indicators in the project proposal.

2 Foundations of the PoF Reference Model

The goal of the PoF Reference Model is to ease the use of preservation systems for personal as well as for organizational settings. In a nutshell, it should be easy to get things preserved, the right things should be preserved and the things should stay useful on the long run. Thus, we have to consider the full cycle of getting things into the DPS, managing them there for a considerable time and bringing them back into use.

This raises three questions: (1) how do we get content easily from the Active System into the preservation system? (2) which content should be put into the preservation system? (3) how can we keep the content useful on the long run and bring it back in a reasonable form, when needed?

For answering question (1) it is important that the gap, which actually exists between an Active System and a DPS has to be bridged. Thus the PoF Reference Model should support an **integrative** approach. For question (2) on which content to select, it makes sense to follow the idea of appraisal typically followed in an archive, i.e., having a decision process for which content to keep. Since we want to ease the use of preservation system, it is desirable that this process is automated, performing information value assessment for identifying important content to be preserved. This opts for a **value-driven** approach for the PoF Reference Model. Finally, question (3) implies that evolution, which takes place while the content is in the archival system has to be carefully considered, in order to keep it accessible and useful on the long run. Thus, the approach taken by the Reference Model should be **evolution-aware**. In addition, to those three characteristics, the ForgetIT approach takes inspiration from the effectiveness of humans in remembering important things and forgetting irrelevant things. Therefore, our Reference Model is also supposed to be **brain-inspired** and **forgetful** in its preservation decisions.

Thus, five main characteristics have been identified for the proposed PoF Reference Model, supporting the ForgetIT approach for a sustainable and smooth transition between information and preservation management. The model is:

1. **integrative** - bridging the gap between information and preservation management for easing the adoption of preservation technologies;

2. **value-driven** - acting upon short-term and long-term information value based on careful multi-faceted information value assessment;

3. **brain-inspired** - learning from the way humans forget and remember for a better more focused management of digital memories;

4. **forgetful** - using the idea of forgetting in the digital memory for staying focused and supporting preservation decisions;

5. **evolution-aware** - embracing the long-term perspective by dealing with change on various levels.

The five characteristics listed above are partially related one to the other. We discuss in more detail below their different facets as well as the requirements and concepts which inspired those characteristics. They provide the foundations for the PoF Reference Model discussed in Sections 3 and 4. The role of such characteristics for a forgetful digital memory is discussed also in [Niederee et al., 2015], where we envision a concept of managed forgetting for brain-inspired long-term data and information management, in order to systematically deal with information that progressively ceases in importance and also with redundant information.

2.1 Integrative: Bridging the Gap between Information and Preservation Management

It is the aim of the ForgetIT approach to preservation to create a smooth transition between active information use and preservation of information, which so far are some quite separate worlds. For this purpose, the PoF Reference Model should be *integrative*, bringing the Active System and the DPS closer together. Further details about foundations of synergetic preservation can be found in deliverable D5.1 [Nilsson et al., 2013] and subsequent WP5 deliverables.

However, due to the inherent long-term perspective of preservation-related solutions it is not the aim to build a strongly integrated, monolithic system. In the long run, it has to be foreseen that the Active System used as well as the employed DPS will change [Afrasiabi Rad et al., 2014]. Therefore, the idea is a flexible integration, which enables smooth bi-directional transition of information between the Active System and the DPS and, at the same time, is also prepared for major changes in the overall environment [Päivärinta et al., 2015].

A core part of integration is to enable the smooth transition of content to be preserved into the DPS and the sensible reactivation of content back into the Active System after a, possibly very long, period into the DPS. An integrative solution should also embrace the idea of a *joint information space*, where the information in the DPS stays conceptually accessible, e.g., visible in search results, even if the content is only available in the DPS.

One part of achieving a smooth transition is to act as a pre-ingest system (or pre-access) and prepare information packages for delivery in either way. With highly automated procedures for preparation of these packages according to agreements or requirements, preservation technology becomes easier to use and the quality of the information packages becomes more consistent, which alleviate the burden of information package handling on both sides (see [Heutelbeck et al., 2009] and D5.2 [Nilsson et al., 2014]). In addition, an integrative system should also support the decisions on what to preserve for easing the integration of preservation into the information management workflows, finally aiming for an integrated information and preservation management workflow (as presented, e.g., in D9.4 [Maus et al., 2015]). The automated selection process (appraisal) will be discussed in more detail in Section 2.4 and in the context of managed forgetting (see Section 3).

2.2 Value-driven: Acting upon Short-term and Long-term Information Value

One of the core ideas of the PoF Reference Model is to deviate from the general *keep it all* model, which makes the implicit assumption that all information has the same value with respect to being kept. In general the value of information is multifaceted and can be considered from different perspectives, e.g., the short-term value for current activities vs. long-term value of a resource for an organization. For the combined information and preservation management system, short-term value and long-term value of information has to be considered separately, since it is driven by different factors. For a review of short-term and long-term memory and forgetting see deliverables D2.2 [Logie et al., 2014] and D2.3 [Logie et al., 2014].

Short-term value The short-term value refers to the value of content for the current focus of activity, e.g., documents used for a task at hand are of high short-term value. In short-term value of information, we will see high dynamics in the information value (due to changing interests and tasks) and a high influence on interaction-based evidences on the information value. In terms of the human brain, this is roughly comparable to the working memory (see Section 2.3), although human working memory has an even higher change frequency. Further details about foundations of managed forgetting can also be found in deliverable D3.1 [Kanhabua et al., 2013] and subsequent WP3 deliverables.

Identifying the short-term value of a document is of high interest for creating immediate benefit in information management, e.g. by de-cluttering the desktop, which is one of the goals of the ForgetIT project and realized in the Personal Preservation Pilots described in D9.3 [Maus et al., 2014] and D9.4 [Maus et al., 2015]. The idea here is to give higher priority or visibility to resources with high short-term value. In the ForgetIT project the term *Memory Buoyancy* has been coined for this purpose. It is inspired by the idea that resources of decreasing importance to the user are sinking away from the user (decerasing Memory Buoyancy).

Long-term value The long-term value of a resource is obviously relevant in the context of preservation. It refers to the value a resource has on the long run. Such long-term value can be used to decide about the investment to be made into preservation for the respective resource [Andersson et al., 2014]. Long-term value is expected to be more difficult to compute, since it includes estimating future use of resources [Rauch et al., 2013]. Furthermore, long-term value is driven by (at least partially) different factors than short-term value. It is expected that more objective values such as diversity, coverage and quality will play a more important role here (see deliverable D3.4 [Zhu et al., 2016] for further details about the dimensions used for resource value assessment). In the ForgetIT project, we have coined the term *Preser-*

vation Value for the long-term value of a resource[1]. The long-term perspective is covered by the methods presented in in deliverables D3.3 [Kanhabua et al., 2015] and D3.4 [Zhu et al., 2016] and in the organizational application scenario in deliverable D10.4 [Dobberkau et al., 2016] and in the knowledge management view on the content value assessment in D9.5 [Maus et al., 2016].

2.3 Brain-inspired: What can we Learn from Human Forgetting and Remembering?

Humans are very effective in forgetting irrelevant details and things no longer relevant. This enables them to focus on the relevant things and to efficiently make decisions in their current life situations. This ability of the human brain inspired the idea of considering a form of forgetting for digital memories in the ForgetIT project. It is investigated, if similar mechanisms in digital memory might be helpful in better coping with the general situation of information overload, i.e., helping to focus on the really relevant digital resources. However, if we just copy human remembering and forgetting, the digital memory would just forget the same things as the human. Clearly, this is not the goal. Rather, the methods in digital memory should learn from and *complement* the human memory: it is desirable that a computer focuses on ”remembering” those things that the human might forget, but which are still useful or required later.

The idea of complementing human memory suggests to take an embracing perspective on human and digital memory, considering them as a joint system. In such a system, the two parts are not considered in isolation: their interactions and mutual influence are taken into account as well. Figure 1 shows models of the human memory and a digital system as well as interactions (joint system perspective). The high-level model of the functioning of the human brain depicted in Figure 1 in based on work described in D2.2 [Logie et al., 2014] and D2.3 [Logie et al., 2015]. The model in Figure 1 is focused on the synergy between human and digital memory. The human memory (on the left) and the digital memory (on the right) together contribute to a form of virtual memory a human can rely on. The idea of the joint model has been published in [Niederee et al., 2015].

On the human memory side, three main type of memory are distinguished: *working memory*, *episodic memory* and *semantic memory*. Together with the currently activated episodic and semantic memory, the *verbal short term memory* (things just heard) and the *visual short term memory* (things just seen) form the working memory, which frames the current situation. Knowledge is activated on demand from the semantic and episodic memory according to current needs via the so-called *executive functions*. *Perception* is one driver for such activation. It is worth noting that perceived signals do not directly become part of the verbal or visual short term memory, which are constantly updated, but they are rather filtered and interpreted by things already in the memory for making sense of the perceived signals.

[1]See deliverable [Kanhabua et al., 2015] for a definition of Preservation Value.

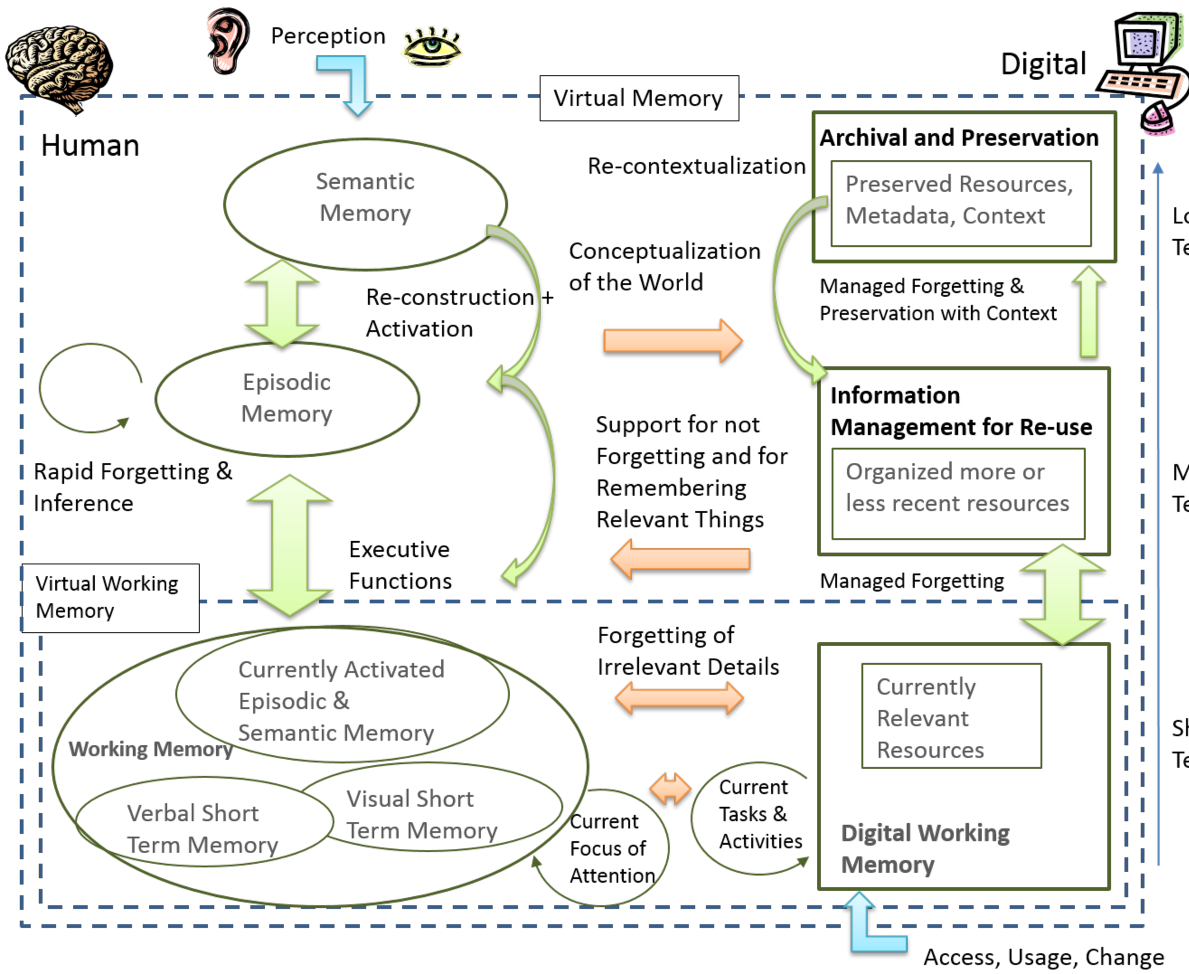

Figure 1: Joint Perspective on Forgetful, Interacting Human and Digital Memory (see also [Niederee et al., 2015]).

Similarly, we also foresee a working memory within the digital memory in our model. This is composed of the digital resources currently relevant, e.g., used or relevant for current tasks or activities. The idea here is to clearly distinguish those resources from the rest of the digital resources and to keep them as close as possible to the user and easily accessible as possible (see also Memory Buoyancy). In an automated digital working memory signals from resource usage, pattern of usage and change as well as relationships between resources will be used to update the digital working memory. For this purpose, in our model, we introduce managed forgetting functions which control the transitions between the different parts of the digital memory.

Together, the working memory and the digital working memory form the virtual working memory. Clearly, there is an influence between both of them. In the ideal case, the digital working memory would show to the user just all the information that the user needs in the current situation, but does not have in her working memory. Note that it is also possible that there is an influence of the way the digital memory works on the human (working) memory (joint system perspective). For example, with the easy storage of phone numbers in mobile phones, humans no longer have to remember those numbers.

Managed forgetting functions are also used to identify content that is of long-term value (see Section 2.2) and should, therefore, be preserved. In Figure 1, we distinguish (a) information management for re-use, as it is, e.g., done on a desktop computer or a server and (b) the system for archival and preservation (on top of the Figure). When content is transferred to *Archival and Preservation*, it makes sense to add context information to it (*contextualization*). This prepares the content for re-contextualization which is required when preserved content is brought back at a (much) later time. The idea of re-contextualization is to connect the re-activated content with the current environment or to - at least - make it understandable in the current environment. The idea of re-contextualization as an active situation-dependent process of bringing back things ”stored” in the memory is again inspired from human memory: when we as humans remember things this is also a re-construction process, which depends upon the current situation. Further details about foundations of contextualized remembering can be found in deliverable D6.1 [Greenwood et al., 2013] and following WP6 deliverables.

Episodic memory is a detailed storage of events and is typically subject to fast forgetting as well as blurring between the memory of similar events due to interference. Here, digital memory complementing human memory, e.g. via photos, can serve as a reminder of things that are forgotten, but that one might remember or refresh in a later point in time, e.g., reminiscing about past events (see also the discussion in D3.1 [Kanhabua et al., 2013]). For this purpose it is crucial to select the adequate content to preserve, so that it can help remembering, e.g. considering coverage and diversity of preserved content.

Semantic memory is a more conceptual storage of memory, which stresses on patterns, abstractions and lessons learned. Here, the strongest interaction between human memory and digital memory is that the organization of digital resources does or should reflect the conceptualization of the world of the user, which is linked to her semantic memory.

Referring to Figure 1 entities, an example of more explicit modeling of the *Conceptualization of the World* with a richer annotation of resources based on this knowledge in *Information Management for Re-use* and in the *Archival and Preservation* is offered by the Semantic Desktop approach (see WP9): by making search and navigation easier for the user, also re-finding things in the digital memory is improved.

2.4 Forgetful: Focus on the Important Things

The ForgetIT project introduces the idea of a forgetful approach to information and preservation management as an alternative to the dominating *keep it all* approach (see deliverables D3.1 [Kanhabua et al., 2013] and D3.2 [Kanhabua et al., 2014]). The forgetful approach opts for conscious decisions about what is important and thus should be kept (and preserved) replacing the often random form of forgetting (or losing) information as it can be often found with the *keep-it-all* approach (e.g. disk crashes, obsolescence of formats and technology, etc.).

Since preservation comes at a cost , it is important to make conscious decisions about what to preserve or how much to invest in the preservation of which part of the information space (see for example [Kejser et al., 2011], [Bote et al., 2012] and [Rauch et al., 2013]). For this need, a forgetful approach is a good fit.

A forgetful approach is based on *Information Value* assessment, i.e. computing and predicting the value of information resources (see deliverables D3.3 [Kanhabua et al., 2015] and D3.4 [Zhu et al., 2016] as well as [Tuan et al., 2016]). Value for different purposes can be considered. In the context of the PoF Reference Model, short-term and long-term values are important (see also Section 2.2). Effective information value assessment, especially for long-term information value, is a complex task involving a variety of parameters and heuristics. Based on such value, preservation decisions can be taken. On a high level, these decisions could include choice of preservation provider and/or service as well as decisions about redundancy and transformation.

2.5 Evolution-aware: Embracing the Long-term Perspective

Since we are targeting integrated information and preservation management systems, we are operating in a long-lived context, covering a time perspective of several decades. Even things that are considered relatively stable in the current setting of an information system - such as the type or class of content management system in use - will change over time [Afrasiabi Rad et al., 2014]. For sustainable operation it is important to be prepared for such changes. It is one of the core ideas of the ForgetIT approach to keep the important information accessible and usable even in case of large changes in the setting and context of operation. For incorporating evolution-awareness into our Reference Model, several types of evolution with different impacts on the reference model have to be considered:

1. **Changes in conceptual model of the Active System**: this could be due to, for example, changes in the organizational ontology underlying the content structuring as well as processes described in the content. This creates a semantic gap between the archived content (relying on the old implicit or explicit ontology) and the active content (structure by new ontology). This gap has to be bridged, at latest when preserved content is brought back into the active environment, in order to enable correct interpretation of the re-activated content;

2. **Active system evolution and exchange (Migration)**: the used Active System might be subject to major changes or might even be completely replaced by another system, if we look at time frames of several decades. In spite of such changes the content should stay accessible and usable [Afrasiabi Rad et al., 2014];

3. **DPS evolution or exchange**: in the same way, the chosen DPS might evolve or could be exchanged over time. This implies the migration of content into a new

DPS [Hutař, 2013]. In the ideal case this should have as little impact on the Active System as possible [Afrasiabi Rad et al., 2014];

4. **Change in best-practices and technology**: formats as well as employed technologies might become obsolete over time. This requires the identification of such changes as well as adequate actions to react to those changes, such as format transformations.

The last item in the list above is a classical issue of any DPS. It is, therefore, not covered in too much detail in our Reference Model, since we focus on the things that go beyond current best practices in digital preservation.

3 Functional Model

The Functional Model of the PoF Framework is concerned with the main workflows of the preservation approach introduced by ForgetIT and the functionalities required for realizing those workflows. Special focus is given to the novel concepts introduced, namely the aspect of using Managed Forgetting in support of preservation, the role of contextualization and the impact of evolution. The Functional Model is complemented by the Information model presented in the next section.

As in the case of the OAIS model, the function view of the PoF Reference Model, or the functional model for short, considers the main functional entities of the proposed reference model. Furthermore, we also describe the main workflows in the model and how the functional entities contribute to those workflows. Again the stress is on the parts which connect the two types of systems, Active System and DPS, with each other.

3.1 Layers, Workflows and Functional Entities

The proposed PoF Functional Model is made up of three layers, namely the *Core Layer*, the *Remember & Forget Layer* and the *Evolution Layer*:

- the **Core Layer** considers the basic functionalities required for connecting the Active System and the DPS;
- building upon this layer, the **Remember & Forget Layer** introduces the brain-inspired and forgetful aspects into the PoF Reference Model implementing more advanced functionalities for the preservation preparation and the re-activation workflow;
- finally, the **Evolution Layer**, is responsible for all types of functionalities dealing with long-term change and evolution.

The different workflows and functional entities in the PoF Functional Model are associated to the three model layers above, as summarized in Table 1 and depicted in Figure 2. The description of the layers, workflows and functional entities is provided in the next sections. Note, that some workflows are stepwise refined in the more complex layers.

An overview of the PoF Functional Model components (layers, workflows, functional entities) is depicted in Figure 2: within each layer box the relevant entities and workflows are shown. In the following Sections we provide a more detailed representation of each workflow, with the steps associated to each process and the involved entities. It is worth noting that Figure 2 already makes some assumptions about the functionalities implemented in the Active System and, especially, the DPS: those functionalities, which are parts of one of the respective systems, are not explicitly listed in the PoF Reference Model. For our purpose, we assume a OAIS compliant DPS implementing functionalities such as *Ingest*, *Data Management*, *Archival Storage* and *Access* of preserved content: in Figure 2 we omitted all OAIS compliant entities but the *Preservation Planning*, due to its relevance in

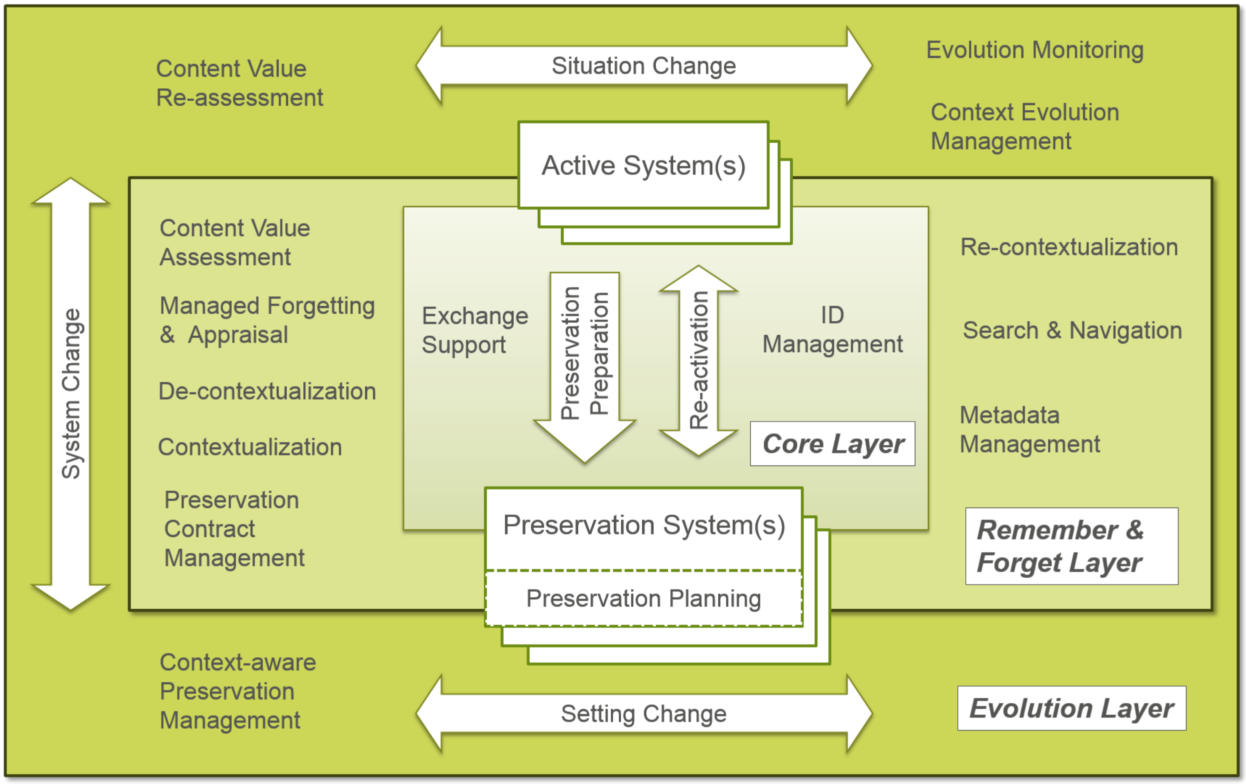

Figure 2: High Level Functional View of the PoF Reference Model

Model Layers	Workflows	Functional Entities
Core Layer	Preservation Preparation (*basic*) Re-activation (*basic*)	ID Management Exchange Support
Remember & Forget Layer	Preservation Preparation Re-activation	Content Value Assessment Managed Forgetting & Appraisal De-contextualization Contextualization Preservation Contract Management Re-contextualization Search & Navigation Metadata Management
Evolution Layer	Situation Change Setting Change System Change (Active System Change and Preservation System Change)	Content Value Re-assessment Evolution Monitoring Context Evolution Management Context-aware Preservation Management

Table 1: PoF Reference Model Components: Layers, Workflows and Functional Entities.

the context of the Evolution Layer, as discussed in the following. Additional details about OAIS functional entities can be found in [CCSDS, 2012].

The three layers are used in the following to describe the functional view of the PoF Reference Model in more detail.

3.2 Core Layer

The Core Layer embraces basic forms of two workflows connecting the Active System and the DPS, the **Preservation Preparation** workflow and the **Re-activation** workflow, and includes two functionalities, **ID Management** and **Exchange Support** (see Figure 3). The workflows and the functional entities are described in the following.

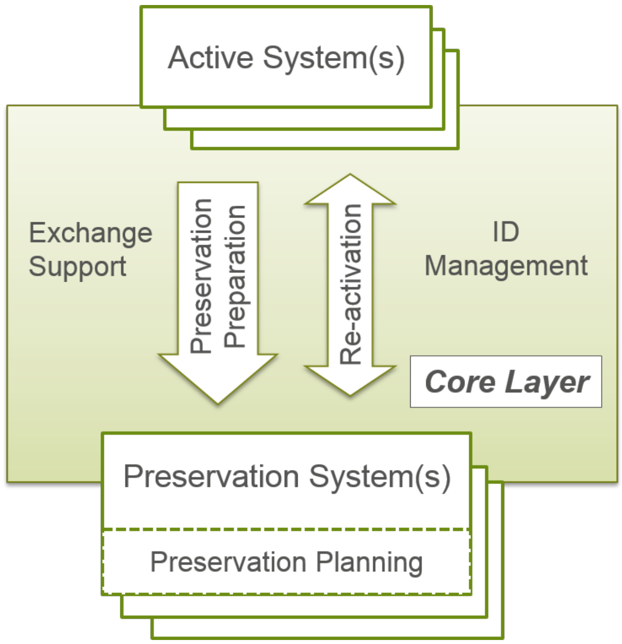

Figure 3: Core Layer of the PoF Reference Model

In support of these two core workflows in their basic form the Core Layer includes two types of further functionality, namely ID Management and Exchange Support:

ID Management The ID Management is mainly responsible for mapping between the IDs of the resources in the Active System and IDs that are used in the DPS for identifying (and locating) the respective content resources. Since several versions of the same resource can be put in the DPS, the ID Management also has to take care of resource versions and their mappings to archive IDs.

Exchange Support Exchange support is responsible for enabling the exchange of content and metadata between the Active System and the DPS. It adapts and maintains protocols for this purpose. The Exchange Support handles both outgoing information and incoming packages that should be put back into active use. This includes basic functional activities for preparation of information packages, such as automatic identification and extraction of technical metadata providing input to the enrich step, and the creation of packages according to the preservation contract. The Exchange Support can be considered as a client-side communication adapter and can, for example, be implemented in the form of a repository and related basic processes.

3.2.1 Basic Preservation Preparation

The Preservation Preparation workflow takes care of transferring content to be preserved between the Active System and the DPS. On an abstract level, this workflow - in its basic form - consists of five steps (see also Figure 4) (1) **select** the content to be archived, (2) **provide** the content to the archival process (3) **enrich** the content with context for preservation (4) **package** the content according to the expectations of the DPS (5) **transfer** the content into the DPS. In terms of preservation terminology, the preservation preparation workflow can be considered a pre-ingest workflow, which leads into the ingest functionality of the DPS. This interaction is discussed in more detail in Section 5. Furthermore, it is worth noticing that the enrich functionality is available on the Core Layer only in its basic form (e.g., to add technical information such as file format to the content to be archived). More advanced enrich functionality is discussed for the Remember & Forget Layer.

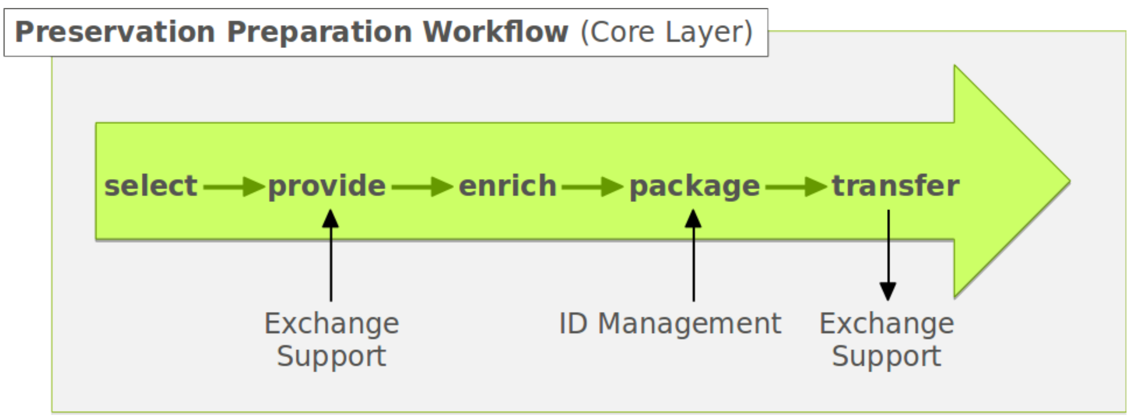

Figure 4: Preservation Preparation Workflow (Basic) in the Core Layer

3.2.2 Basic Re-activation

The *Re-activation* workflow takes care of enabling the Active System to retrieve and re-activate content, which has been transferred to the DPS. Again on an abstract level, this workflow - in its basic form - consists of five steps (see also Figure 5): (1) **request** the content to be retrieved from the PDS (here via its identifier), (2) **search** requested content thus translating the request into archival ID(s) (in the basic workflow just using the ID Manager) (3) **fetch** the respective content from the DPS (4) **prepare** the content for delivery to the Active System, (4) **deliver** content to the Active System. In the Core Layer the DPS is involved mainly in the fetch activity, which makes use of the access functionality offered by the DPS to retrieve the content (see also Section 6).

As in the case of the Preservation Preparation workflow, some of the steps in the Re-activation workflow are only included in a very basic form on the Core Layer and are extended with more advanced functionalities on the Remember & Forget Layer.

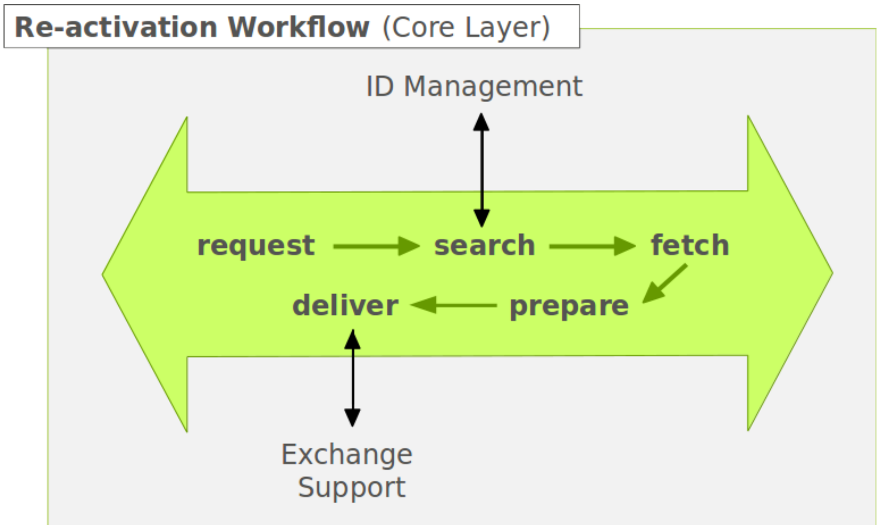

Figure 5: Re-activation Workflow (Basic) in the Core Layer

3.3 Remember & Forget Layer

The Remember & Forget Layer introduces brain-inspired functionality into the PoF Reference Model, which targets the concepts of managed forgetting and contextualized remembering. For this purpose, the Remember & Forget Layer extends the two workflows **Preservation Preparation** and **Re-activation** from the Core Layer with further, more advanced functionalities: **Content Value Assessment**, **Managed Forgetting & Appraisal**, **De-contextualization**, **Contextualization**, **Re-contextualization** and **Search & Navigation**, which are all described in the following. All of those listed functionalities create additional metadata, which have to be managed in a systematic way. Therefore, the Remember & Forget Layer also contains a functional entity for **Metadata Management**. The workflows mentioned above define activities that contain rules, constraints, and settings, that should be agreed upon and that could be defined in a Preservation Contract. This contract is handled by the functional entity **Preservation Contract Management** which potentially relate to many, if not all, entities and are not drawn in the figures to avoid clutter.

3.3.1 Preservation Preparation

In more detail, the Preservation Preparation workflow, which still consists of the five steps *select*, *provide*, *enrich*, *package* and *transfer* (see Figure 6), now uses the additional functionalities of Content Value Assessment and Managed Forgetting in the phase of selecting content for preservation:

Content Value Assessment Understanding the value of content is in the core of content appraisal for preservation and managed forgetting. Content value assessment aims to determine the value of a resource. This value may change over time and

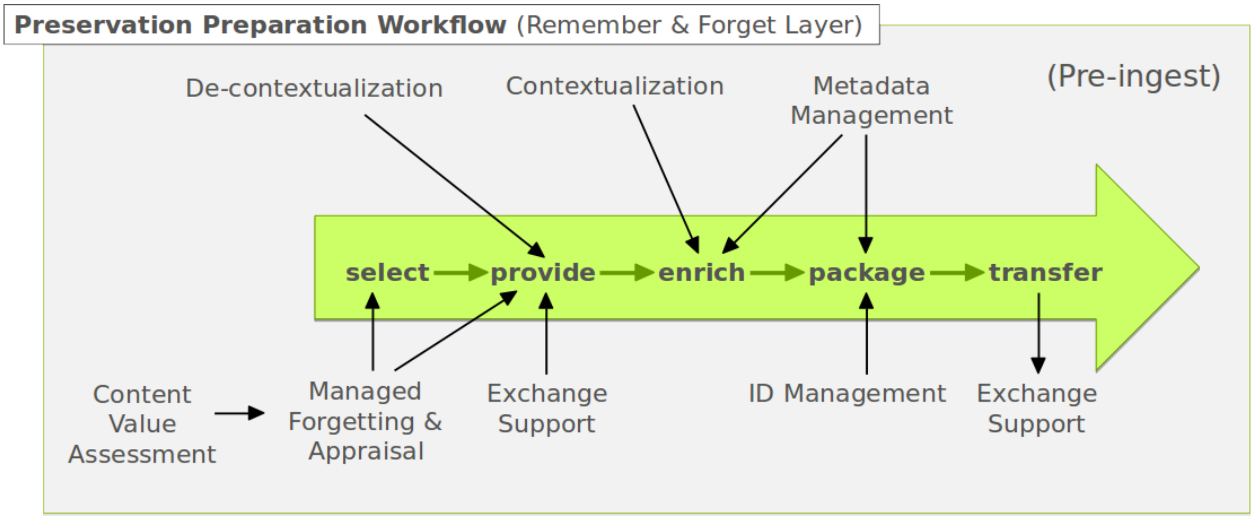

Figure 6: Preservation Preparation Workflow in the Remember & Forget Layer

there are different value dimensions, which reflect the value considering different purposes or perspectives and which may influence each other. There is, for example a value dimension reflecting current importance, e.g. Memory Buoyancy (MB), and a dimension reflecting the long term importance or relevance of a resource, the Preservation Value (PV). For assessing content value, the content value assessment component takes evidences from the Active System, e.g. about information use, content creation, and further knowledge about the role of resources in the Active System. Content value can be used as a basis for making preservation decisions, e.g. if a resource should be preserved or how much should be invested in the preservation of a resource. Content value can also be used in the Active System, e.g., for especially highlighting resources with high content value.

Managed Forgetting & Appraisal With the dramatic growth of the amount of content, nowadays it becomes more and more important to make conscious decisions about preservation. Clear decisions on what to put into the DPS and explicit content appraisal have always been part of the processes of an archive [Harvey, 2006], although not always as much in personal archiving [Marshall, 2011]. The component for appraisal and managed forgetting aims to help in automating such decisions, a need that has been identified earlier [Harvey and Thompson, 2010], for both personal archiving as well as organizational settings. This is encapsulated in the concept of managed forgetting, which uses the results of content value assessment for deciding about preservation and forgetting actions. The effects of managed forgetting functionality is not restricted to the preservation functionality. It can also be used in the Active System for improved information access.

Furthermore, the workflow steps provide and enrich are extended with De-Contextualization and Contextualization functionality, respectively:

De-Contextualization De-Contextualization refers to the extraction of an object from its Active System context in preparation of packaging it for archiving. Decoupling the object under preservation from its Active System context is non-trivial, since it has to be decided how much of its current context has to be taken for its future contextualization and where a cutting can and should be made. De-contextualization and contextualization (see below) are conceptually closely related.

Contextualization Contextualization consists of providing sufficient additional information for the content to be preserved, in order to allow archived items to be fully and correctly interpreted at some undefined future date. This entity is responsible for defining and assigning the appropriate context to content to be archived. Contextualization can leverage other processes (similarity analysis, concept detection) to explicate context. Contextualization provides the basis for the management of context evolution over time (see Evolution Layer in Section 3.4) and Re-contextualization (see Re-activation workflow in Figure 7).

The Preservation Preparation workflow is linked to the **Pre-Ingest** functionality as it is described for Preservation Systems, e.g. in the PAIMAS model. In order to facilitate easy (seamless) ingest into the DPS and make sure that the packages contain metadata needed for both the DPS as well as for access, pre-ingest aids the Active Systems as well as the DPS systems adhering to standard protocols and metadata, as defined in the Preservation Contract. This also means that the Pre-ingest function puts up some requirements on the Active Systems to follow certain protocols (which can/should be domain specific). Our Preservation Preparation workflow, from the perspective of the DPS serves as a pre-ingest function.

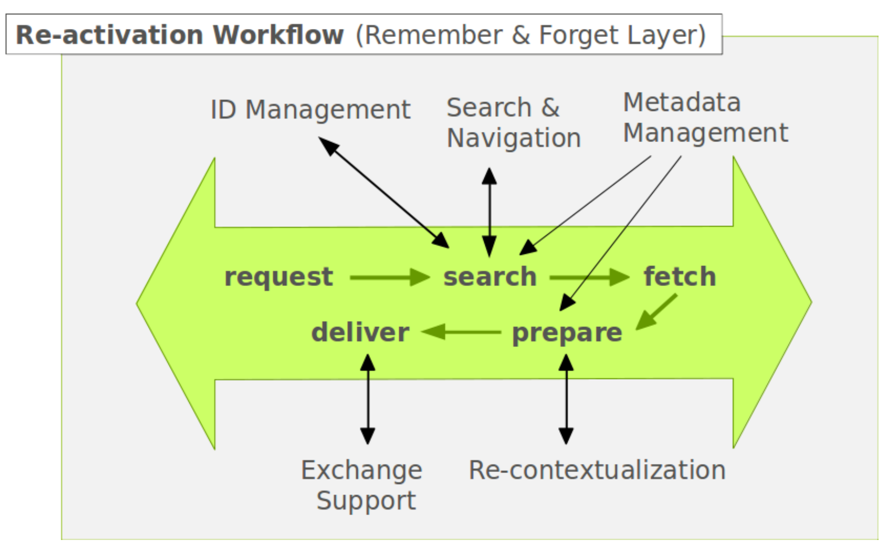

Figure 7: Re-activation Workflow in the Remember & Forget Layer

3.3.2 Re-activation

For the Re-activation workflow, two types of additional functionality are used, with respect to the Core Layer, namely Re-contextualization and Search & Navigation:

Re-contextualization The purpose of the Re-contextualization functional entity is to support the interpretation of a content object at the time of access (which might be a considerable time after archival). Re-contextualization occurs when a document is retrieved from the DPS at some future date. Once retrieved from the DPS and before it is put back into active use, the context information, which has been provided by the contextualization functionality, stored together with the content object and possibly updated or extended over time is retrieved. This context information is used for Re-contextualization, i.e., to relate the content object to the current usage context. Re-contextualization can also include the re-construction or extension of context information for content archived with no or insufficient original context.

Search & Navigation The Search & Navigation functional entity is responsible to enable finding things that have been preserved. Various types of search and navigation can be supported here. This includes search in the metadata, full-text search in the content (or more general content-based search also including non-textual content), search in the context information and in other types of annotation, exploratory search for understanding the archive content, etc. For our integrative and forgetful approach it is crucial (a) to manage the interaction between the search in the Active System and the search in the DPS and (b) to understand how the forgetful approach and search support interact. Since we are following an integrative approach, it makes sense for (a) to consider the information in the Active System and the DPS as a type of a *joint virtual information space*, which are both considered for search in the Active System. However, it might still make sense to differentiate the two types of content taking into account the cost that might be attached to accessing content from the DPS. Archive content might, for example, only be considered on demand or if nothing can be found in the Active System. Furthermore, content stemming from the preservation store might be marked in result lists. For aspect (b), the influence of the forgetful approach, the results of content value assessment, namely MB and PV, can be considered in result ranking (or even indexing): this would prefer results with higher content value balancing content value and relevance as it is for example done in diversification approaches. Furthermore, this includes adequate filtering and ranking approaches for handling versioned archive content. Situation search as it id explained in the section on System change can also be used here. However, since it is especially relevant for Active System Change it is discussed in Section 3.4.

3.3.3 Metadata and Preservation Contract

Finally, in both workflows metadata are generated and used for different purposes. This metadata is taken care of by the functional entity for Metadata Management:

Metadata Management The Metadata Management functional entity is responsible for the different kinds of metadata that are created and exchanged by the aforementioned functional entities. This includes a variety of different types of metadata such as current and past values for memory buoyancy and preservation value, information on the context of a resource, information extracted from resources for further processing as well as indexing information for search and navigation. Some of the metadata, which is collected here just remains in this middle layer for supporting its operation. Other parts of the metadata such as the preservation value and the context information will be stored as part of the archived object in the preservation system. The Metadata Management of the Remember & Forget Layer interacts with the respective components of the preservation system by (a) providing input for enriching the metadata in the preservation system for improved preservation management, (b) incorporating information coming from the preservation system (e.g. for the joint indexing) and (c) by - as mentioned before - storing some of the metadata as part of the resources to be preserved.

Preservation Contract Management In general, before transferring any digital items or collections from Active System to DPS, a submission agreement has to be established between the participating actors, preferably following a standard approach, e.g. as described in the Producer-Archive Interface Methodology Abstract Standards (PAIMAS) [CCSDS, 2004]. In ForgetIT, we extend this to also include aspects of relevance to the re-activation of content and, therefore, label it *Preservation Contract* to signify that it does not only concern the submission phase.

The preservation contract should contain accurate information about type of package content, structure, and metadata. It should also include requirements for security and privacy mechanisms at transfer and storage. Furthermore, it should be stated in the agreement if there is a need or requirement for migration at ingest. Other examples are a specification to what extent metadata should be obtained and generated during the pre-ingest process, if there is specific demand on storage, or rules regarding management of objects in the DPS including different preservation levels as forgetting options. There is also room for agreements about e.g. privacy requirements and exploitation rights.

3.4 Evolution Layer

A preservation framework per definition exists in a long-lived environment aimed to survive at least decades. Clearly, most of the involved parts widely ranging from formats, technology and systems to semantics, and relevant real world situations (represented by the application) will not remain stable (i.e. unchanged) over such long period of times. This implies that adequately dealing with changes is a core property that is required from such a framework.

The purpose of the Evolution Layer is to adequately deal with the upcoming changes, such that the preserved content remains accessible and understandable. Clearly this means different things for different types of changes. Other types of actions are, for example, required in case a media format gets out of use as compared to the situation that an organization is re-organized.

The Evolution Layer, therefore, groups the changes considered in three classes, each handled by its own type of change workflow. Thus, the Evolution Layer contains three new workflows: the **Situation Change** workflow is responsible for monitoring semantic changes in the Active System and propagating them into the DPS, in order to keep the preserved content understandable; the **Setting Change** workflow deals with changes in practices, formats and technology in the environment of the preservation (setting); finally, the **System Change** workflow is responsible for situations, where one of the involved systems changes.

In support of the aforementioned workflows, the Evolution Layer includes additional functionalities: **Content Value Re-assessment**, **Evolution Monitoring**, **Context Evolution Management** and **Context-aware Preservation Management**.

3.4.1 Situation Change

Active Systems such as the semantic desktop reflect real world processes and situations and thus the content and structures in such systems are subject to change, which we capture under the notion of ”Situation Change”. Since the preserved content co-exist with such changes, this raises the question, if such changes have implications for the preserved content. Even if the content is not directly affected, changes such as re-structuring in an organization or change in life situation of an individual might have implications for the interpretation and contextualization of preserved content.

For this purpose a Situation Change Workflow has been defined. This workflow consists of four steps (as depicted in Figure 8): (1) **change monitoring**, (2) **change assessment** (assessment of detected changes), (3) **change notification** (notification of involved components as well as the DPS on relevant changes) and (4) **change propagation**, which performs different types of actions depending on the observed change and the chosen change propagation strategy.

Change monitoring is responsible for detecting changes in the content and the content structuring of the Active System. This is typically best realized by an extension of the active system. An important and more demanding step in the workflow is change assessment. The idea here is that changes are analysed with respect to their potential impact to the preserved content. It does not make sense to consider all the small changes that happen in the Active System. Rather Situation Change is only interested in the ”larger” high

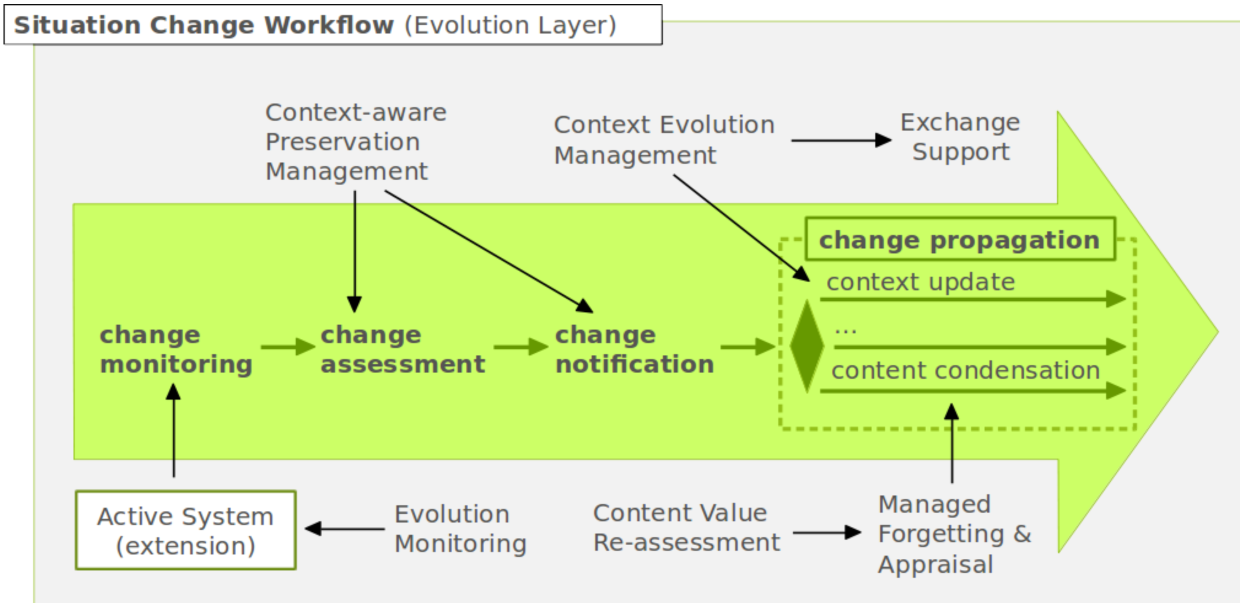

Figure 8: Situation Change Workflow in the Evolution Layer

impact changes. One way of dealing with this idea is to distinguish more and less important concepts, instances and relationships in the content structures of the active system and to propagate this importance level to the changes affecting such entities. Alternatively or in addition, it is also possible to give different impact levels to different types of change operation, e.g. the deletion of a department will have higher impact than a smaller change of name. Change assessment is the basis for filtering out irrelevant changes and for performing change notification, i.e., informing relevant system components about the change. This will lead to change propagation, which can take different forms depending on the type of the change and the selected change strategy. A change might for example imply that the preserved context information is modified or extended, in order to capture the change and to be able to reflect it, when re-contextualization is performed. Another action that can be triggered by a change is content value re-assessment: Due to a change, e.g., a change in employment, an entity such as the former employer might cease in importance, which would lead to a decreased preservation value of content items related to such an entity.

For monitoring, assessing the changes and deciding about the consequences, in support to the Situation Change workflow, the functionalities Evolution Monitoring (mainly part of Active System), Context Evolution Management and Content Value Re-assessment have been introduced in the Evolution Layer:

Evolution Monitoring The Evolution Monitoring functionality is required for the change monitoring step in the Situation Change workflow. This is mainly performed in the Active System (extension), because this is the main place, where evolution in the conceptual model of the application becomes visible. Such changes might, for example, be changes in the ontology (implicitly) underlying the organization of the information triggered, e.g., by a major re-organization. Evolution Monitoring has to observe changes in the explicit representations of the conceptual model (such as taxonomies, information structures) in the Active System as well as more implicit signals of context evolution (e.g. newly upcoming topics, tags getting out of use, department sites no longer updated).

Context Evolution Management The Context Evolution Management is responsible for keeping up-to-date the context information which has been stored along with the archived content. This may include the storage of further context information, in case of larger changes in the Active System (e.g. major restructuring of an organization). This also includes keeping track of such larger changes, which have happened in the Active System on a conceptual level. Such a change history can be used in Re-contextualization for making the content understandable in the changed new context.

Content Value Re-assessment The functionality Content Value Re-assessment serves a very similar purpose as the functionality Content Value Assessment described for the Remember & Forget layer. It re-visits the value assessments originally provided by the Content Value Assessment based on observed evolution in context. It is considered separately in our model, since the resources that it works on are now already preserved, which has implications on the computation of the value (e.g. the role of usage data) as well as on where this functionality is performed. One option, which is followed in the ForgetIT project is to map it to in storage computation.

3.4.2 Setting Change

Although the DPS should have its own support for typical OAIS functionalities, such as Preservation Planning and Administration, the many-to-many relationship in the PoF Reference Model and the long-term perspective of those functions mean that at least parts of those functionalities need to be shared and communicated over several systems. The OAIS **Preservation Planning** function could for example benefit from Technology Watch residing in the middleware, thereby gathering and aggregating information about, e.g., usage of file formats in all systems connected to the same middleware. Another example would be when Preservation Planning in a DPS declare a file format as obsolete, this information could then be shared with the PoF Middleware which propagates this to Active Systems and other DPS.

The PoF Middleware acts as a man-in-the-middle (broker) between Active Systems and DPS and therefore has a suitable position for capturing these bi-directional interactions. Based on such evidences, it provides additional possibilities to summarize and analyse the usage and, e.g., storage quotas over several systems, thereby giving an overview of the holdings for a particular customer. Since a customer might have several Active Systems, as well as preserved content in several DPS, such an overview is beneficial in locating objects that need preservation actions. This process is part of what is labelled ”Setting Change”.

The Setting Change workflow consists of four different phases with two different starting points (as depicted in Figure 9), involving the Context-aware Preservation Management (CaPM) entity (described below): (1) **activity monitoring**, which logs the bi-directional communication between the Active System and DPS including process activities, systems in action, and digital objects passing through, (2) **change assessment** that detects and propagates change in usage, (3) **change estimation** suggests suitable change recommendations based on rules defined in Preservation Contract including, e.g., Preservation Value, and use statistics, (4) **change recommendation**, which propagates recommended actions to DPS, which could be of different types, such as transformation of content or change of physical and logical content structure.

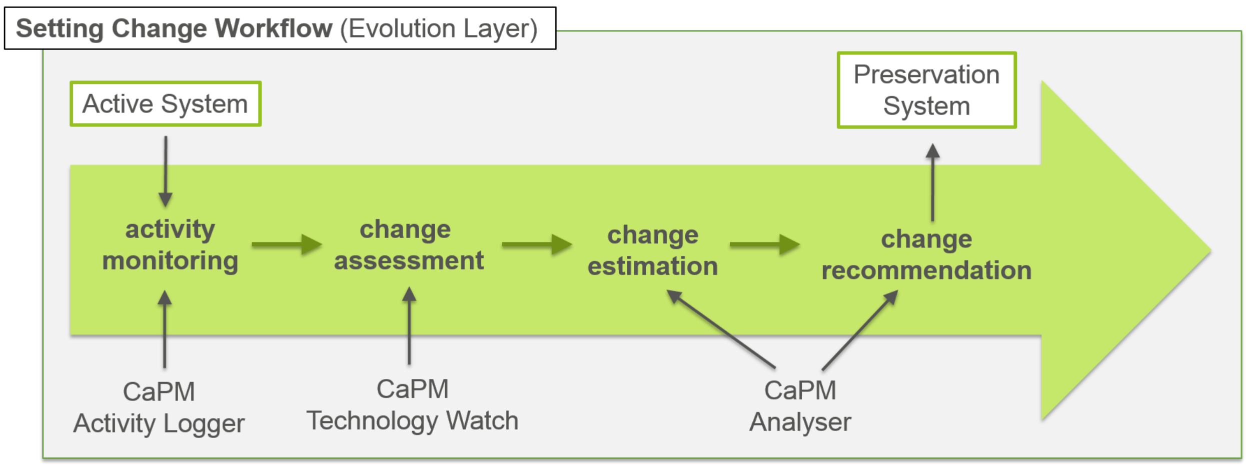

Figure 9: Setting Change Workflow in the Evolution Layer

The Evolution Layer includes the functional entity Context-aware Preservation Management (CaPM), with three supporting functional entities, namely: CaPM Activity Logger; CaPM Technology Watch; and CaPM Analyser. Together they support the workflow described above:

Context-aware Preservation Management The Context-aware Preservation Management functional entity (CaPM) externalises and extends parts of OAIS Preservation Planning and OAIS Administration by keeping track of Active Systems as well as the (several) DPS that are involved. The main idea here is that preserved information should be easy to seamlessly put back into active use in either the same system as it originates from, but even more importantly into other information systems (of the same type). Even the same system might have evolved to newer versions with different standards, and maybe even a different information structure or ontology. This must also be tracked by the Context-aware Preservation Management.

CaPM Activity Logger The Activity Logger provides support for monitoring of activities in PoF workflows. A typical example is to keep track of the execution of tasks executed by components in the workflows. This data will then serve as one input to the change assessment.

CaPM Technology Watch The Technology Watch gathers data on objects handled by the PoF workflows. This includes information on Active System and DPS, as well as technical metadata on the object, e.g., file format. This information combined with input from Activity Logger form the basis for a change assessment, which, if needed, triggers a change estimation.

CaPM Analyser Based on an initial assessment of the need for a setting change, the change estimation process is supported by the Analyser, which aggregates and processes data provided from earlier functions and should provide a graphical user interface for human interpretation and intervention. The Analyser may calculate a preferred setting change based on thresholds and rules, provided by, e.g., a Preservation Contract, but it can be interactively overridden by a human operator. The decision, be it automatic or manual, will then become a change recommendation that is handed over to the DPS for implementation of the recommendation.

3.4.3 System Change

Within the Evolution Layer, System Change refers to the case, where one of the participating systems is replaced. This might be the Active System (e.g. because a new type of active system is adopted for the same task) or the DPS (e.g. because a preservation provider goes out of business or the use wants to change the provider). Of course, it is also possible that the middleware is replaced with a middleware solution based on a different approach or technology. However, this case is out of the scope of this reference model.

The two cases, change of Active System and change of Preservation System have very different implications and are, therefore, considered separately.

Active System Change

The most interesting aspect for the change of Active System is how the content preserved with Active System A can still be accessed, if the system no longer exists or is no longer in use. This might mean to access the preserved content through a new Active System B or to access it via a lightweight access infrastructure such as a Web Interface. For covering those cases we have created a modified Re-activation workflow (see Figure 10).

A prerequisite for accessing the content of Active System A is some type of an identifier (and credentials) for getting access to the content that has been connected to a user via Active System A. After (1) **authentication**, the owner or another user on behalf of the owner can access the content. For not overwhelming the user with the full content collection, which might have a considerable size, a high-level and user-friendly structuring principle is required. For this purpose, the concept of *situations* has been introduced into the reference model (see also section on Information Model). Therefore, after authentication, the user can perform a (2) **situation search**, which returns to the user a list of

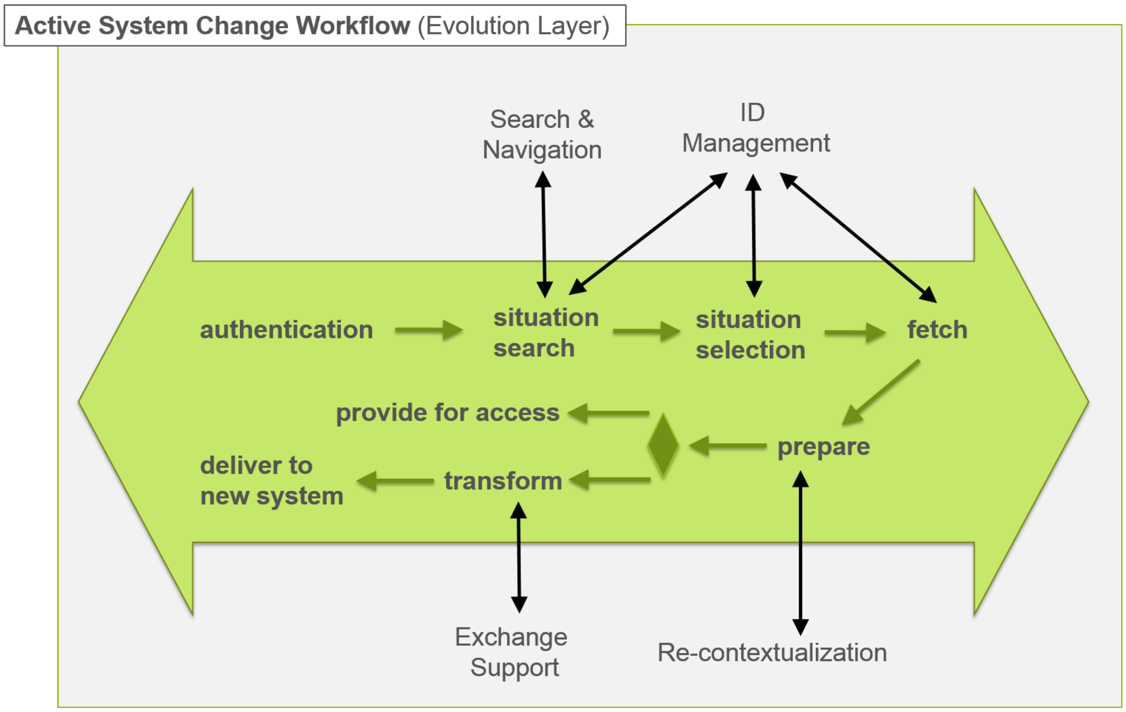

Figure 10: Re-activation Workflow for Active System Change

situations, for which there is content in the archive (or the list of situations fulfilling the user query). Search can be either offered as an integrated part of the new Active System B or separately (e.g, via a Web Interface, see above). For each of the situations also a short description in the form of a profile is given (see Situation Profile in Section 4.1). The (3) **situation selection** step enables the user to select relevant situations, which subsequently during the (4) **fetch** step can be retrieved by the system via the mapping from situation to content objects provided by the ID manager. During the (5) **prepare** step, as in the normal case of Re-activation (see Remember & Forget Layer), the content is prepared applying e.g. Re-contextualization and made available via the Exchange Support (in the PoF Framework the CMIS standard is used for this purpose). At this point, the (6) **transform** step might be necessary to transform the content and the context into a format that can be digested by the Active System B before the (7) **deliver to new system** step takes place. Alternatively, if no content transformation is required, the (6) **provide for access** step can be given to the content as it is provided by Exchange Support, e.g., for browsing the content.

Preservation System Change

A change of preservation system can happen for various reasons including technology change, preservation provider change, etc. In each case it will become necessary to get the preserved content out of the preservation system to be discarded and to import it into a new preservation system. In addition to dealing with the content it is also necessary to deal with the preservation contract associated with the content, as can be seen in the workflow shown in Figure 11.

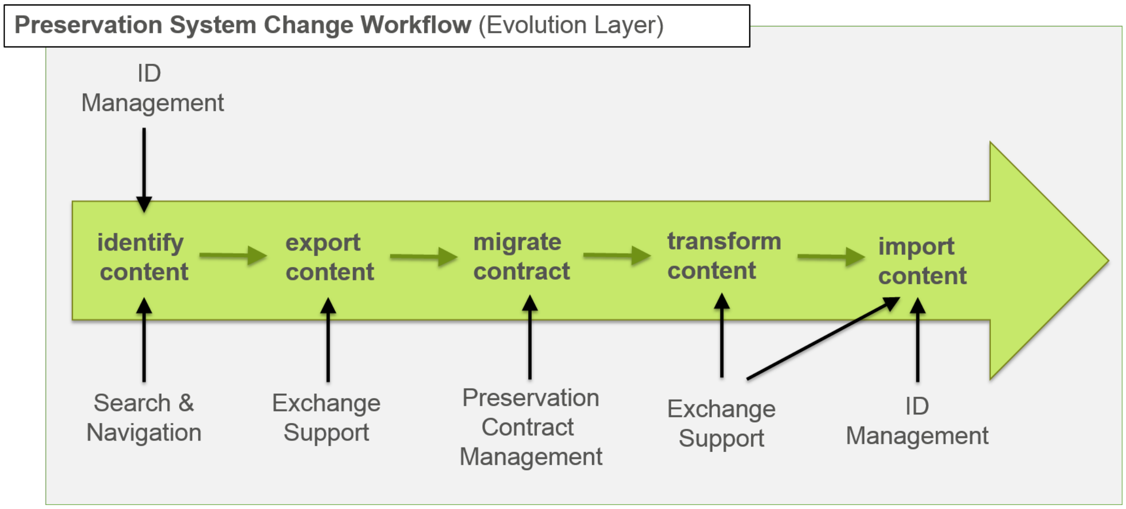

Figure 11: Simple Migration Workflow for Preservation System Change

The workflow shows a high-level Preservation System Change process. During the (1) **identify content** step, all the content that belongs to the system instance or the user that wants to change the preservation system is identified. In case of a technology change, this must be done for all the users or a bulk export function can be used; then the (2) **export content** step requires that all the content is fetched and exported together with the necessary metadata, e.g., linking content to situations or linking content to context. Also, since the middleware has information on what metadata and the format of metadata that is required from the new preservation system it is possible to request the export to be in a suitable format; in addition, during the (3) **migrate contract** step, the Preservation Contract is migrated to meet the service offer, possibilities and practices of the new preservation system; the migration might have implications that require re-negotiations with the user, therefore it is expected that contract migration is no fully automated process; it may be then necessary to (4) **transform content** (and metadata) in order to comply with the newly negotiated contract, depending on what was possible to achieve already during the export content phase. This would also lead to an update of the provenance metadata for the objects, since they have been handled by different actors during this process. Subsequently, content is imported, possibly after re-packaging according to the new Preservation Contract: this takes place during the (5) **import content** step, when the ID Management functionality is used to link the new archival IDs with the IDs known to the Active System and possibly internal IDs of the Middleware.

Under certain circumstances, a change of preservation system might just concern a subset of the digital objects in the current preservation system. In that case, it could be useful to utilise the CaPM Analyser and CaPM Technology Watch to support decision making on whether to make that change or not, based on, e.g., the number of objects concerned or the volume of those objects.

4 Information Model

In this section, we describe the information model as part of the PoF (Preserve or Forget) Reference Model which bridges the gap between the Active System and the DPS.

In order to meet the requirements of a variety of systems on the site of the Active System as well as on the site of the DPS, the information model has been designed to be flexible, extensible and inter-operable.

The target of the PoF Information Model is twofold: a) it should reflect the perspective of the User, i.e., the active system, thus enabling the active system to interact with the PoF compliant system and b) it should reflect the perspective of the Middleware in order to support all the needed information for storing content in the archive and for retieving it from the archive. Therefore, we present two perspectives of the information model, the user perspective (see Figure 12) and the Middleware perspective (see Figure 13). The information model presented in this deliverable has been developed based on the very preliminary information model presented in deliverable D8.2 [Gallo et al., 2015b].

The three core elements of the ForgetIT Information Model are:

- Content
- Context
- Situation

Content, obviously, represents the content to be preserved. **Context** provides additional information helping in the interpretation of the content, as required when considering the long-term perspective. **Situation** provides a high level structuring concept for archival content. The meaning of Situation is discussed in more detail in the next section, before the information model as a whole is presented.

4.1 Situations as Units of Preservation

Users as well as organizations have to be able to incrementally store things and retrieve units of content from the archive even after long time periods. This should also be possible in the case, when the original active system no longer exists. To this end we have chosen a *situation-driven* approach: typically, content is created for (or in) a situation such as a holiday trip for personal context or is created in a project for organizational context and is also associated with this situation in memory, e.g. my photos of my trip to Paris. Therefore, we have decided to introduce Situations as an important high level structuring concept into the Reference Model. According to our definition, a **Situation** represents an event, a life situation or an experience from the perspective of a user or an organization. For enabling a high flexibility, which fits many preservation settings a rather wide definition is chosen here. Examples for situations are: a holiday trip or a life event such a wedding in the personal context, a project or a project meeting in the organizational context. In the archive the content is associated with the situation (or rather a representation of it). This does not mean that all resources such as images belonging to the same situation have to be archived at the same time or have to be stored at the same place in the archive.

The purpose of introducing situations is to provide the user with a high-level notion for perceiving the structure of the potentially large and growing amount of content that is put into the archive over the time. Archive content will be associated with situations and can be accessed in terms of situations. In some way, situations are similar to the concept of collections as they are used in digital archives. However, we see an important difference. It is the purpose of the archive to enable the memorization of situations by storing content that enables remembering relevant aspects of the situation. The preserved content thus illustrates the preserved situations.

Similar to an event, a situation is associated with a time span, which describes the temporal dimension of the situation and with location information. Due to the wide definition of situation the considered time frame can vary widely. This is driven by the user's understanding of situation granularity. A Situation can be very short, but also can span several years. Nested Situations are also possible, allowing situations to be part of larger ones.

Table 2 summarizes the core attributes of the Situation Profile, which is the representation of a situation in the Information Model. It provides a core set of metadata for describing the situation.

Attribute	**Description**
Id	Unique ID for the Situation
Title	Name of the Situation
Type	Type of Situation
Time	Time range related to the Situations
Locations	Location or locations related to the Situation
Persons	Important persons related to the Situation
Further entities	Important entities that are not persons, such as pets, buildings or organizations
Memory Cues	Information that is expected to be useful in memorizing and retrieving the Situation.
Description	Optional short description of the Situation.
Preservation Value	Preservation value of the Situation

Table 2: Core attributes of the Situation Profile.

In order to enable interoperability among different active systems (if we need to migrate from one to another or in case of changes), an agreement upon or standardization of situation profiles would be desirable. This would ease the access of situations, which have been archive on behalf of one active system and are supposed to be brought back in the context of another active system.

An approach similar to Dublin Core (DC) can be adopted, by identifying a core set of elements to be used as descriptive metadata associated to a given situation. Such attributes could be standardized, as done by The Dublin Core Metadata Initiative (DCMI), and further specified in order to cover a wide variety of situations. Subsequently, further attributes can be agreed upon for specific types of situations such as holidays, projects, etc., resulting in a more specific set of attributes comparable to the qualified DC elements.

Completing this Section describing the Situations, we introduce some aspects related to the human memory behavior. The notion of the situations is related to concepts of cognitive science in two ways. First of all, the situation can be related to episodic memory. Following the current understanding of human memory behaviour (see for example [Baddeley et al., 2009] and [Baddeley, 1999]) and more precisely focusing the attention on the *Episodic Memory* (as discussed in [Conway and Loveday, 2010]), our brain and *human archival system* has to store "*Situations*" with the associated and related *contents*. A *Situation* in some aspects can be compared to a *Episodic Memory Item* (EMI) (rather than being translated to a rigid OAIS-like *Information Package*). An EMI is:

- *complex*, because it is made up of several parts and concepts (or even sub-parts);
- *heterogeneous*, because its parts are of different kinds: we have associations and links, contexts, concepts, images, texts, and the like;
- *dynamic*, because an EMI can change over the time due the changes of associated resources;
- *unstructured*, because even if some parts can be structured more or less rigidly (i.e. by concept hierarchies and semantics), many other parts can be completely unstructured such as images, contextual information (feelings, moods or environmental cues) and generally not classified information;
- *loosely coupled* with other EMIs, for example when events are related to other events, we may capture memory items linked each other but with some.

Events can be related to other events creating arches in the brain's associative model, arches [Allasia and Palumbo, 2015], that can be overwritten or reinforced as well as forgotten over time [Palumbo and Allasia, 2015]. Situations in our PoF Information Model have to be considered in a similar way, in order to support and complement human memory with an archival system [Logie et al., 2015].

A second aspect of Situations that is inspired by cognitive processes is the inclusion of memory cues into the Situation Profiles.

4.2 Core Information Model: User Perspective

From the point of view of the user or of the Active System, we can summarize the Information Model as shown in Figure 12.

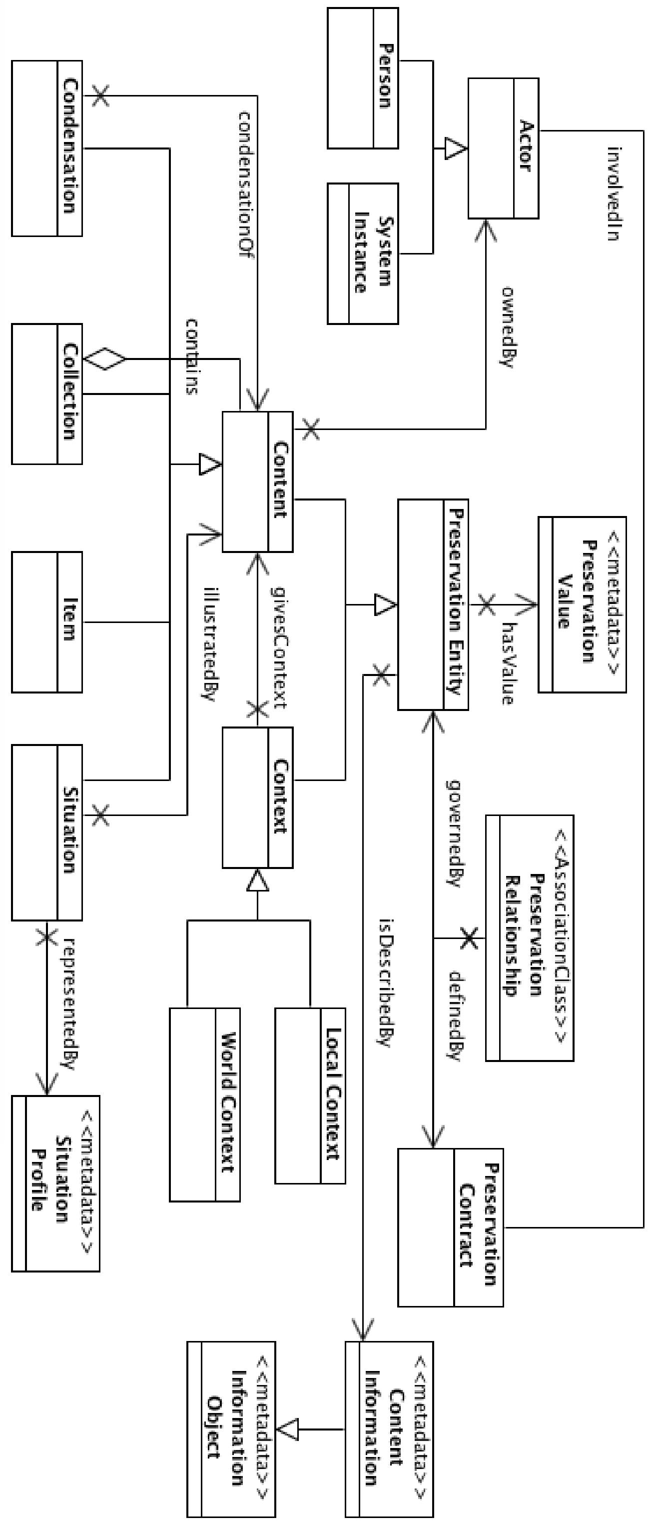

Figure 12: Information Model from User Perspective

As stated above in previous Section, the three classes **Content**, **Context**, and **Situation** are the core elements of the Information Model. Both, **Context** and **Content** are **Preservation Entities**, i.e. they are elements, that the user can provide to the archival system for preservation. Content can be an individual content **Item** or a **Collection**, which will again contain content objects.

As another child of **Content**, the Information Model contains the concept **Condensation**, which is an object created from a content object such as a collection or an content item by condensation operations such as *text summarization*. *Information reduction* is an important concept in the *forgetting* environment and is one of the possible forgetting actions resulting in condensed objects that can replace or augment and improve the original projects.

Information on **Context**, the second core concept of the PoF Information Model, is kept in the archive for helping in the interpretation of content objects. Context can be provided in many different forms and for content on all levels of granularity, including individual items, collections as well as situations. In addition, we distinguish **Local Context** and **World Context** depending upon the scope of the respective information: this distinguishes information that is only known in the local scope and information considered as *world* knowledge.

The third core element, **Situation**, is also a child of **Content**. This provides a high flexibility allowing nesting of situations as well as for nested collections to be assigned to a *Situation*. A Situation is *represented by* a **Situation Profile**, an associated Profile storing the properties of the *Situation* (details of the Situation Profile are described in Table 2). As shown in Figure 12, the Situations in Reference Model are linked by the association *representedBy* to the **Situation Profile**. As explained above, *Situation* and *SituationProfile* are *going beyond* the classic and more rigid OAIS [CCSDS, 2012] specifications.

A further important concept in the Information Model is ownership of the preserved content. This is modeled by introducing the concept of an **Actor**, which can be a **Person** or a **System Instance** of the **Active system**. For modelling ownership, the **Actor** is linked to the **Content** element via the relationship *owned by*. This enables for flexible assignment of ownership.

Another important concept of the user's perspective of the information model, is the relationship to a preservation provider. This is modeled in the Information model by the associative class of a **Preservation Relationship**, which is attached (associated) to both the **Preservation Entity** and the **PreservationContract**. Such a relationship is defined by a **Preservation Contract**, in which the preservation actions for the respective Preservation Entities are defined (see also [CCSDS, 2004]).

In addition, Preservation Entities are also associated with a **Preservation Value**, which reflects the expected benefit of preserving the respective entity and acts as a parameter for the preservation processes (mainly assessments and preservation planning activities).

All the information related to **Content** (such as title, type, etc.) is represented by **Content Information** which can be considered as child of an **Information Object** according to

OAIS.

The individual Model Entities are detailed in Table 3.

4.3 Extended Information Model: Middleware Perspective

In the previous section we have described the PoF Middleware from the point of view of the *User*. Here we describe the point of view of the Middleware. The Middleware has to manage the mapping between the Information Model as seen by the user (or organization) and the Information Model of the underlying preservation system (or systems).

Figure 13 shows the Extended information model, which reflects the perspective of the Middleware. The elements (classes) coming from the *User Perspective* have been highlighted (yellow background color) and can be considered as *core elements* of the model.

This model introduces a number of concepts that are relevant for the preservation system, extending the user experience and adding the needed elements for managing and supporting the *digital preservation*.

The Middleware Perspective in Figure 13 introduces the so called *Detachment Elements* [Borghoff et al., 2006] that are the low level elements such as **File** and **Bitstream** (children of **Component**, aggregated by **Item**) as well as the detailed metadata information represented by **Descriptors** (that are all children of **Information Object**). These **Detachments** have been designed in order to be compliant to the most widely adopted preservation metadata standards, as described in Section 6.

A core aspect of the Middleware perspective is shown at the right corner of the figure, the **ID Mappinfg Table**. Since the preservation entities as they are seen by the active system are not mapped 1:1 to the DPS, it is necessary that the Middleware manages a mapping between the Preservation Entities and the Archival Resources in the respective repositories. For maximal flexibility an ID mapping is used for this puprpose. The class **Preservation Entity Identifier**, which is a type of **Reference**, is the identifier of a preserved entity. This can be identifiers of different kinds. Therfore, there are at least three children, the **CMIS ID** is the identifier used for the resource in the Active System, **PoF ID** is an identifier used for internal purposes and **Repository ID** is the ID of the Resource in the Archive. All these IDs are related to each other by the **ID Mapping Table** that can be considered as their *aggregator* (represented by the black diamond in the diagram). This mapping table is also use to link archival resource to Situations.

A class specifically useful for the Middleware and not for the User, is the **Archival Resource**, a parent class of **Preservation Entity**. Actually the User has to deal only with **Preservation Entity**, while the Middleware has to store into archives the related resource as top level. Hence a further abstraction layer was needed in order to associate it to a specific child of **Information Object**, the **Repository**. In our PoF Middleware we have to manage **CMIS** repositories, i.e. every repository exporting a CMIS interface [OASIS, 2013].

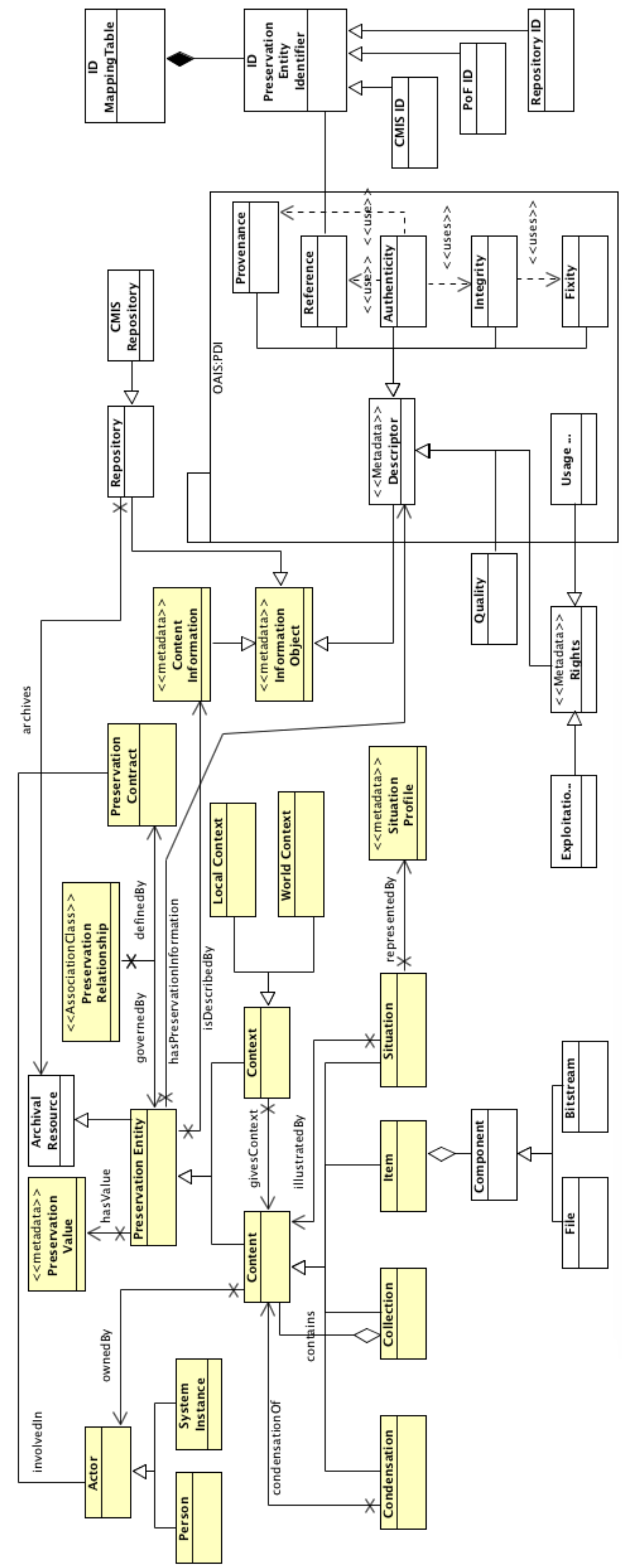

Figure 13: Information Model from Middleware Perspective (User Perspective in Yellow)

Furthermore, the **OAIS:PDI** information package has been added to the Middleware Perspective, which embodies the *Preservation Description Information* according to the OAIS model [CCSDS, 2012]. They are mostly **Descriptors** needed for expressing digital preservation information. We have identified a **Descriptor** stereotyped as *Metadata* that inherits the more generic **Information Object**. The latter is inherited by many low level descriptive information elements. Among others we have reported here the **Provenance**, the **Reference**, the **Authenticity**, **Integrity** and **Fixity**. As described in the Figure 13 in order to assess the Authenticity we need to make use of the other descriptors, expressed by the *uses* dependency dashed arrows: **Fixity** is needed for guaranteeing the **Integrity** which is needed for assuring the **Authenticity** that needs also **Provenance** information [Gladney, 2007] and related resources and references.

Together with the OAIS Preservation Description Information we have added some other *Elements* coming from other standards and best practices in digital preservation (see Section 6).

The OAIS Descriptor expressing the *Rights* has been improved: the Descriptor **OAIS Rights** has quite general purposes. The experience matured in professional environments (e.g. broadcasting) where archived digital contents have strict rules for their government has induced us to split the **Rights Descriptor** into two main elements: **Exploitation Rights** and **Usage Rights**. Without this distinction it may happen that many *preservation actions* can be blocked and/or forbidden: there are many cases where the lack of *usage rights* are practically freezing the digital preservation system, leading to the impossibility to preserve and curate the stored contents. Even if the preservation system has a copy of the contents (and is ideally able to preserve it), if it is not sure about the exact rights and permission on their modifications and changes it cannot undertake any preservation action. To this end we need to clearly state as **preservation description information** (PDI) this difference and clear it. We split exploitation and usage. Separating the Rights about **Exploitation** (eventually described into **Contracts**) from the **Usage** allows the digital preservation systems to execute the needed preservation actions without infringing the copyright laws. As good practice in digital preservation, during the submission and ingestion of digital contents into preservation storage systems, it must be explicitly granted the usage rights, especially if the digital archive has not the ownership of the contents and cannot commercially exploit them. We have expressed the *rights* to perform these actions with the *hasRights* association between **Actor** and the generic **Rights**. As already discussed above, an **Actor** can be a **Person** as well as a **System Instance**.

A **Descriptor** not planned in the OAIS but vitally important especially for media contents, is the **Quality** (represented as child of **Descriptor** as well) and implementing all the digital properties related to the quality of a digital item. An image can have, for instance, a quality property expressing the blurriness level and a video can have a property for the *lossy* or *lossless* compression adopted (e.g. YUV 4:2:2).

The complete list of **Elements** introduced in the Information Model has been provided in Table 3. It is worth noticing that due to the close correspondence to the MP-AF standard some elements have been grabbed accordingly to [ISO/IEC, 2016].

Table 3: Information Model Entities for User and for Middleware Perspective (see Figure 13)

Entity Name	**Description**
	User Perspective
Situation	A **Situation** is an event, a life situation or an experience from the perspective of a user or an organization. It is introduced as a user-oriented structuring principle for the preserved content. A user can store, retrieve and access content in terms of situations. Situations are illustrated by content objects, typically collections of preserved content items. An example is the situation ”My Holidays in Crete” illustrated by a selected set of photos from this trip with high preservation value.
Situation Profile	**Situation Profile** is a metadata record describing a situation. In its core it consist of a set of attribute value pairs as described in Table 2. The information in the situation profile is kept simple and system independent, in order to ease finding and recognizing situations, even if the active system no longer exists.
Content	**Content** is an abstract concept for referring to different types of content that can be provided for preservation and can be part of content collections. This can be individual items (Item), collections of other content items (Collection) as well as condensed information objects (Condensation). Furthermore, also Situations are considered as Content.
Context	**Context** is extra information, which is intended to help in the interpretation understanding the preserved content objects, especially, when re-accessed after a long time. Context can be collected in different ways e.g. provided by the user, extracted or collected from external sources and it can come in different forms e.g. structured semantic information, extra images, text, links to external knowledge basis, etc.
Local Context	**Local context** is context information associated with a content object based on local knowledge. In contrast to world knowledge, this is information, which is (mainly) only known in the environment of the content object (e.g. within the organization, or by a person and his/her family and friends.
World Context	**World context** is context information associated with a content object based on ”world knowledge”. It is assumed that such world knowledge is generally known (not only in the local environment) and, thus, can be retrieved from external sources at the time of preservation, but also retrospectively at the time of later access. This property makes local context more important for preservation than world context.

Table 3: Information Model Entities (continued)

Entity Name	Description
Collection	A **Collection** is a content object consisting of a set of content objects. Those contained content objects can either be individual content items such as an image or they can be again collections, thus allowing nested collections.
Condensation	A **Condensation** is an content object, which has been obtained by a condensation operation from one or a set of other content object. This can for example a text summary obtained by summarizing a text document or a set of text documents. Another example is a low-resolution image obtained by a transformation from a high-resolution image or a reduced image collection obtained by using a near-duplicate detection method. Condensation plays an important role in creating additional forgetting options beyond keep-or-delete.
Item	An **Item** is an individual content object, such as an image, a text file or a video. An item may also consist of groups of subitems and/or components that are bound to relevant Descriptors. The Item descriptor contains information about the Item. An Item that contains no sub-items can be considered a whole. An Item that does contain sub-items can be considered a compilation. Items may also contain annotations to their sub-parts [ISO/IEC, 2016]
Preservation Entity	A **Preservation Entity** is an object to be preserved coming from the active system. This concept provides an umbrella over the two core concepts Context and Context in the PoF Information Model capturing them both as preservable objects. Furthermore, it links such objects to be preserved with preservation-related information: It links such object to a preservation relationship, which defines the way the respective object is preserved, and to a preservation value, which act as a parameter to the preservation process.
Preservation Value	**Preservation value** is a value reflecting the expected benefit from keeping the related Preservation on the long run. Here numerical values as well as more user-friendly preservation categories such as gold, silver, bronze, etc. are possible. Preservation value can be used both for deciding if something is preserved or not and for actually deciding about preservation options, e.g., the redundancy level.

Table 3: Information Model Entities (continued)

Entity Name	Description
Preservation Relationship	A **Preservation Relationship** is a relationship between an Actor and a preservation service provider, who takes care of the preservation actions. The Preservation actions and constraints are defined in a Preservation contract. The actions taken for a Preservation entity are governed by this Preservation Relationship and the associated Contract.
Preservation Contract	A **Preservation Contract** is an agreed upon contract between a preservation provider and a preservation client, which defines rules for digital preservation actions and communications to be undertaken within the middleware as well as within the preservation system (see [CCSDS, 2004])
Actor	An **Actor** is an entity that takes an active role as a subject with respect to content to be preserved. This can for example be as the owner of a content object or as a preservation contract holder. An Actor is not restricted to be a person. It can also be a system instance acting on behalf of a person or an organization in the interaction with a preservation middleware. Indirectly, this will typically boil down to a person or a role in an organization, which for example has the ownership of the content.
Person	A **Person** is a real person or a role within an organization that can be taken by several persons at the same time or over time. In the Information Model, persons come into play as owners of content objects and as actors involved in preservation contracts.
System Instance	A **System Instance** is an instance of the Active System, which interacts with the Preservation Middleware on behalf of a user or a group of users. In this role, it might act as an intermediary between the user(s) and the Preservation System. However, since the active system may go out of operation, it is important to define processes on how to deal with relationships such as content ownership in this case.
Information Object	**Information Object** is defined according to OAIS specification and provides a set of attributes defining the semantics (meaning) of a content. In our model it can be considered as the abstract element for **descriptors**
Content Information	**Content Information** extends the Information Object and represents all the information related to a Preservation Entity, it may contain editorial information (such as title, author, editor, series, year, ID code, keywords, etc.), technical (such as format, length, size, etc.) and other descriptive attributes.

Table 3: Information Model Entities (continued)

Entity Name	Description
	Middleware Perspective
Archival Resource Component	An **Archival Resource** is a resource as it is archived in a DPS. A **Component** is the binding of a digital resource to a set of Descriptors, i.e. the information concerning all or part of the specific resource instance. A Component itself is not an Item; Components are building blocks of Items [ISO/IEC, 2016].
File	A **File** is a Component materialized as a unit recognized by a computer system, subsystem, or application [ISO/IEC, 2016].
BitStream	A **BitStream** is a Component recorded as contiguous or non-contiguous data within a File. If metadata are specific to streams or tracks (e.g. audio and video tracks of a file), Bitstream shall be used and descriptors shall be added on Bitstream level [ISO/IEC, 2016].
Rights	**Rights** represent information concerning legal, regulatory or contractual provisions that affect ownership, control, access or use of resources insofar as they impact the long term preservation (e.g. intellectual property, copyrights, privacy, etc.). Actions or events in the preservation of resources need to respect such rights [ISO/IEC, 2016].
Exploitation Rights	**Exploitation Rights** represent information specifically related to the ownership and commercial exploitation of the digital resource.
ID Mapping Table	The **ID Mapping Table** provides a mapping among the different types of entities based on their IDs. The user only needs to know the ID generated in the user application and the preservation system only makes use of a repository ID. The middleware bridges the gap between Active System and Preservation System by linking such identifiers, also using a PoF identifier for internal purposes. Note that the mappings can also be 1:N mappings. The mapping table can be implemented in different ways.
Preservation Entity Identifier	A **Preservation Entity Identifier** represents the different identifiers associated to a given Preservation Entity in the different systems or for the different perspectives: the user perspectives requires a Active System identifier (**CMIS ID**), the middleware perspective makes use of a **PoF ID**, while a **Repository ID** represents the content identifier in the preservation system. The way these identifiers are implemented and managed depends on the particular implementation. For the Active System identifier we assume an ID based on the CMIS standard.

Table 3: Information Model Entities (continued)

Entity Name	**Description**
Repository	A **Repository** is a system used to store digital content and its associated metadata, providing methods to import, update, search and access content, exposing standard and application-independent interfaces. A digital repository can be part of a Content Management System used by a user application or be included in a Digital Preservation solution, where it is extended with long term preservation functionalities.
CMIS Repository	A **CMIS Repository** exposes interfaces based on the OASIS CMIS standard, to enable interoperability among different content management systems and to provide functionalities to import, update, search and access the content using CMIS standard.
Usage Rights	**Usage Rights** represent information related to the usage of the digital resource. They define if preservation actions can be undertaken by preservation systems.
Descriptor	A **Descriptor** associates information with the enclosing entity. This information may be a Component (e.g. image) or a textual statement [ISO/IEC, 2016].
Quality	**Quality** provides information related to the description of the technical condition of preserved Items and resources. This information can at least partly be automatically extracted from content with specialized tools but often requires manual revision and validation. This manual work causes considerable costs, which is an additional reason for preserving it. Quality information includes digital defects (such as audio and visual) and characteristics, their collocation in time and space and their severity. Additionally, structural information and technical metadata of resources in relation to relevant standards are considered. It is worth to highlight that it is needed to preserve as wells the description of the hardware devices, tools and agents used for extracting and reviewing that quality information [ISO/IEC, 2016]
Provenance	**Provenance** documents the chronology of events regarding the creation, modification, ownership and custody of a resource, such as who produced it and who has had custody since its origination; it provides information on the history of the multimedia content (including processing history) [ISO/IEC, 2016].

Table 3: Information Model Entities (continued)

Entity Name	Description
Reference	**Reference** provides information that is used for identifying the digital resources. It provides one or more identifiers, or systems of identifiers, by which the resources may be uniquely and persistently identified. Reference information supports the linkage of identical or related resources that might be stored in separate repositories. These repositories may use different mechanisms for identifying resources (e.g. using different standards for representing local identifiers) [ISO/IEC, 2016].
Authenticity	**Authenticity** provides information to enable any User to verify that an object is correctly identified and free from (intentional or accidental) corruption (i.e. capable of delivering its original message). Authenticity encompasses identity and integrity (as well as Provenance and Reference). Identity comprises all those attributes necessary to determine what a thing is (e.g. the original recording of a Work). Integrity asserts that none of those essential attributes have changed, i.e. there are no significant differences neither in the same resource over time nor between two resources thought to be copies of the same asset. While identical copies are authentic, authenticity does not require complete equivalence. Thus, a digital version of an analog original may be an authentic copy of the Work if it can be shown that the differences between the two versions are not significant, e.g. all of the content is present and is structured the same way, and all important elements or attributes, such as title, creator, performer, remain the same [ISO/IEC, 2016]
Integrity	**Integrity** represents the state of a Digital Item indicating the fact of being complete and unaltered. It can be proven by verifying the presence of all required parts in an unaltered (i.e. not modified) state [ISO/IEC, 2016]
Fixity	**Fixity** encompasses the information ensuring that resources (as described by their properties) are not altered in an undocumented manner. This information is also used to verify the integrity of Digital Items. Thus, if the fixity information for an Item changes over time, the Item has changed [ISO/IEC, 2016]

5 Mapping to ForgetIT Architecture

In the following, we describe how the PoF Reference Model can be mapped to the architecture of the PoF Framework, described in deliverables D8.1 [Gallo et al., 2013] and D8.6 [Gallo et al., 2016].

5.1 PoF Framework Architecture

The architecture of the PoF Framework, described in detail in deliverable D8.1, is made up of three layers: *Active Systems*, *PoF Middleware* and *Digital Preservation System (DPS)*, as depicted in Figure 14.

The Active Systems represent user applications, while the PoF Middleware is intended to enable seamless transition from Active Systems to the DPS (and vice versa) for the synergetic preservation, and to provide the necessary functionality for supporting managed forgetting and contextualized remembering. The PoF Middleware provides also the integration framework for all components developed in WP3-WP6. The DPS is composed by two sub-systems (Digital Repository and Preservation-aware Storage, including a Cloud Storage Service) and provides both content management and typical archive functionalities required for the synergetic preservation.

The deliverable D8.1 also contains a preliminary discussion about the role of the OAIS Reference Model in the overall architecture: according to the project proposal, since OAIS nowadays is the most recognizable conceptualization of a DPS, it was considered as one of the building blocks of ForgetIT approach. However, the model described in Sections 3 and 4 complements and supersedes this initial approach towards the OAIS model. We further describe the relationship with OAIS and other digital preservation standards in Section 6.

5.2 Relationship with PoF Architecture

We provide the mapping between each functional entity in the PoF Reference Model and the architectural components of the PoF Middleware in Table 4. The list of components is taken from deliverable D8.1.

It is worth noting that for some components the mapping with model entities is not one-to-one, because more than one component can participate in the implementation of a given functional entity of the model. The `Scheduler` component is not explicitly mentioned in Table 4, because it mainly provides process management functionalities supporting the different workflows across the three layers described before.

Other preservation related functionalities, which are typically supported by a DPS, also benefit from PoF Middleware components. For example, in terms of OAIS entities, two functionalities which are relevant for the Evolution Layer, such as Preservation Planning and Administration, are partially supported by the Context-aware Preservation Manager. Another typical DPS-related functionality, such as the Pre-ingest, is supported by the Archiver (and a dedicated workflow for Preservation Preparation).

Functional Entity	**Model Layers**	**PoF Middleware Components**
ID Management	*Core*, *Remember & Forget*	`ID Manager`
Exchange Support	*Core*, *Remember & Forget*	`Collector`, `Archiver`, `Metadata Repository`
Content Value Assessment	*Remember & Forget*	`Forgettor`
Managed Forgetting & Appraisal	*Remember & Forget*	`Forgettor`
De-contextualization	*Remember & Forget*	`Contextualizer`
Contextualization	*Remember & Forget*	`Contextualizer`, `Extractor`, `Condensator`
Preservation Contract Management	*Remember & Forget*	`Context-aware Preservation Manager`
Re-contextualization	*Remember & Forget*	`Contextualizer`, `Archiver`
Search & Navigation	*Remember & Forget*	`Navigator`
Metadata Management	*Remember & Forget*	`Forgettor`, `Extractor`, `Condensator`, `Contextualizer`, `Metadata Repository`, `Collector`, `Archiver`
Content Value Re-assessment	*Remember & Forget*	`Forgettor`, `Contextualizer`
Context-aware Preservation Management	*Evolution*	`Context-aware Preservation Manager`
Evolution Monitoring	*Evolution*	`Context-aware Preservation Manager`
Context Evolution Management	*Evolution*	`Context-aware Preservation Manager`, `Contextualizer`

Table 4: Mapping between PoF Reference Model Functional Entities and the PoF Middleware Components.

The PoF Functional Model described in Section 3 includes several workflows spanning across the three model layers:

- the Preservation Preparation workflow (Figure 4 and Figure 6) for the Core Layer and for the Remember & Forget Layer;

- the Re-activation workflow (Figure 5 and Figure 7) for the Core Layer and for the Remember & Forget Layer;
- the Situation Change workflow (Figure 8) for the Evolution Layer;
- the Setting Change workflow (Figure 9) for the Evolution layer;
- the two System Change workflows for the Evolution layer: the Active System (Figure 10) and the Preservation System (Figure 11).

The different steps of such workflows involve different PoF Framework components, mainly for what concerns the PoF Middleware. We provide a representation of the activation of the different components in each workflow in Figure 15 and Figure 16 for the Remember & Forget Layer and in Figure 17, Figure 18, Figure 19 and Figure 20 for the Evolution Layer.

It is worth noting that some components are involved only in one of the layers, e.g. the Remember & Forget Layer or the Evolution Layer (see Table 4) and that Figure 18 about the Setting Change workflow includes also the Active System and Preservation System components, while the others involve only the components within the PoF Middleware.

5.3 Model Implementation in PoF Architecture

An implementation perspective of the Information Model described in Section 4 is described in deliverable D8.6 [Gallo et al., 2016].

The main challenges in implementing the PoF Reference Model are associated to (a) the integration of the components within the PoF Middleware, (b) the integration of the Active Systems and of the Preservation Systems with the PoF Middleware, and (c) the implementation of some specific components which are crucial for the new ForgetIT approach to digital preservation described in this model. Examples of such components are the `Forgettor`, the `Contextualizer` and the `Context-aware Preservation Manager`, just to name a few.

The integration of the components in the PoF Middleware leverages the outcomes of WP5, detailed in deliverables D5.1 [Nilsson et al., 2013] and D5.2 [Nilsson et al., 2014], which described the foundations of synergetic preservation and a workflow model and prototype for the transition between Active System and DPS. The communication among the components and the business logic for the different workflows are implemented in the PoF Middleware using a Message Oriented Middleware (MOM) approach [Chappell, 2004], powered by a rule-based engine which activates the different components according to specific Enterprise Integration Patterns (EIP) [Hohpe and Woolf, 2003], based on best practices in the field of Enterprise Application Integration (EAI). The choice of the most suitable solution is performed according to ForgetIT requirements and also taking into account the future adoption of the PoF Framework (see deliverable D8.6 [Gallo et al., 2016]).

The adoption of any middleware technology requires some effort by software architects and engineers before it can be used in a effective way. The integration of the Active Systems and the Preservation Systems should be based on standard technologies and robust APIs, to enable the integration with different preservation solutions and user applications.

In the previous Section we described how the different components of the PoF Framework are related to the PoF Reference Model functional entities. The final release of the PoF Framework is fully compliant to the model described here. The first release of the prototype described in deliverable D8.3 already integrated many components and implemented two priority workflows for basic synergetic preservation and managed forgetting support, as described in Section 4.2 of deliverable D8.1. The second release of the PoF Framework, described in deliverable D8.4 [Gallo et al., 2015a], was further improved according to the model and provided an implementation of the Preservation Preparation and Re-activation workflow, integrating all components. Finally, the final release described in deliverable D8.6 [Gallo et al., 2016] implements the PoF Information Model with an improved support for the functional model workflows.

The main challenge in implementing PoF Framework components is related to the novelty of the approach and to the lack of similar technologies or applications. Examples of such components are the `Context-aware Preservation Manager` (synergetic preservation), the `Forgettor` (managed forgetting) and the `Contextualizer` (contextualized remembering), just to name a few which are closely related to the core ForgetIT principles. The development of such components was based on a iterative approach, with new requirements identified within the project as a result of the analysis and discussion among all partners, resulting in the identification of the main processes and the relevant scenarios to be implemented. Moreover, the development of the Active Systems for the personal and organizational scenarios has to provide a proof of the ForgetIT approach validity through the implementation of application pilots. Other components in the PoF Middleware which have not been mentioned above are crucial for the implementation of the relevant workflows, providing the required input for preserve-or-forget decisions, dealing with the core preservation objects in the information model, managing metadata, logging processes and other related tasks.

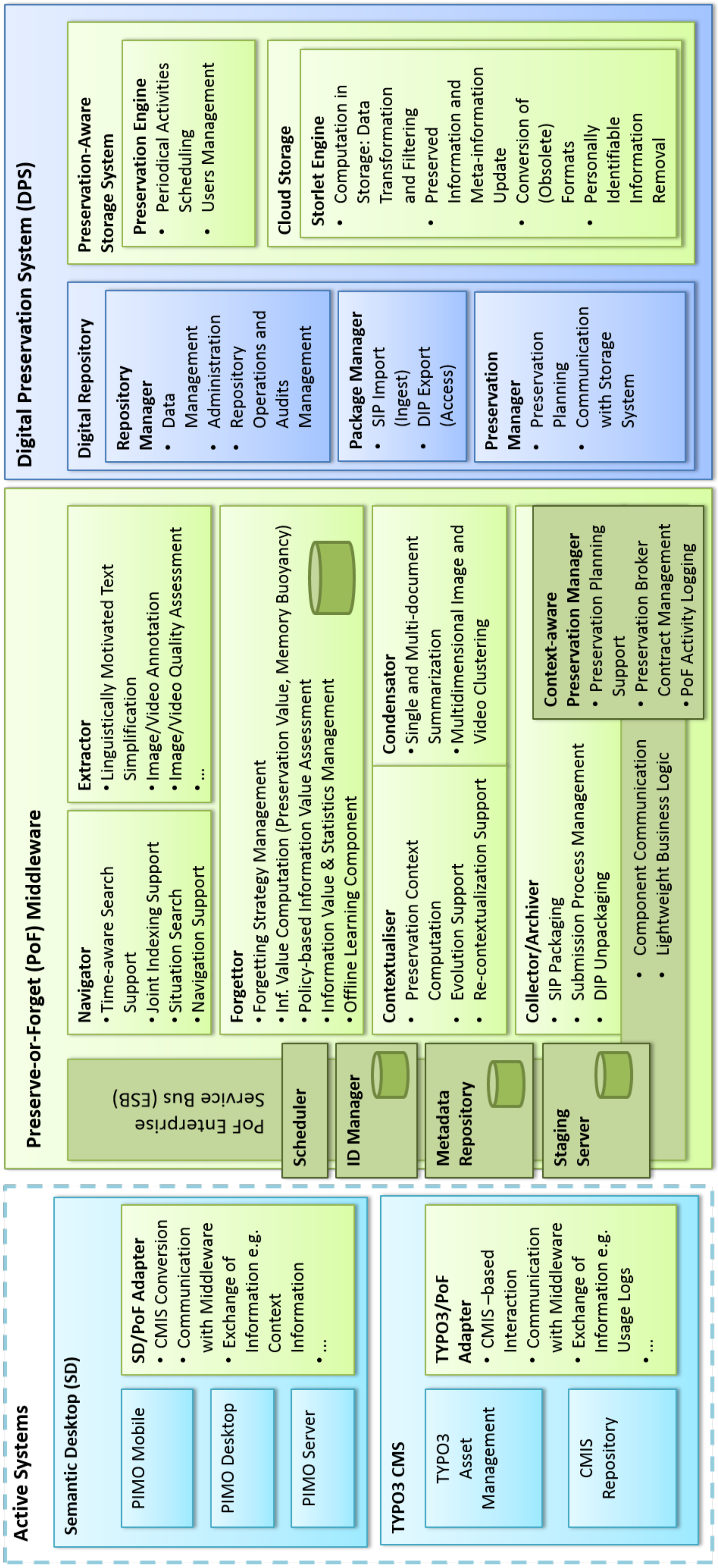

Figure 14: Architecture Diagram of the Preserve-or-Forget (PoF) Framework.

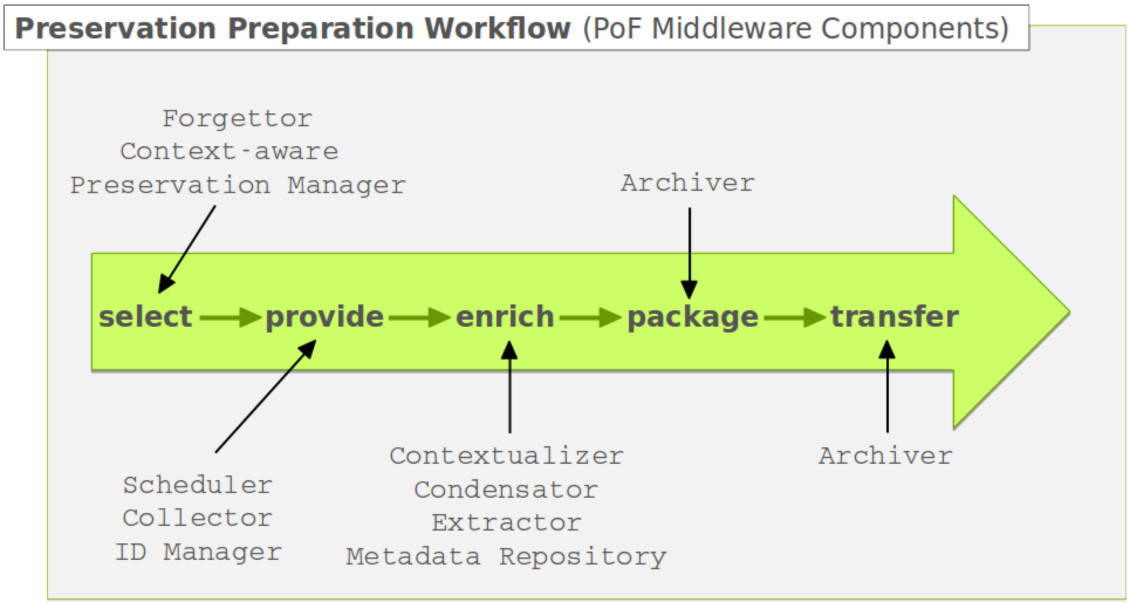

Figure 15: Mapping between the PoF Middleware Components and the Preservation Preparation Workflow.

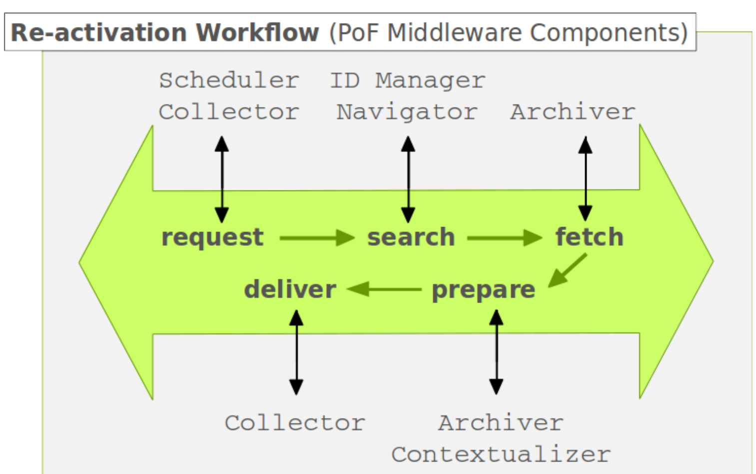

Figure 16: Mapping between the PoF Middleware Components and the Re-activation Workflow.

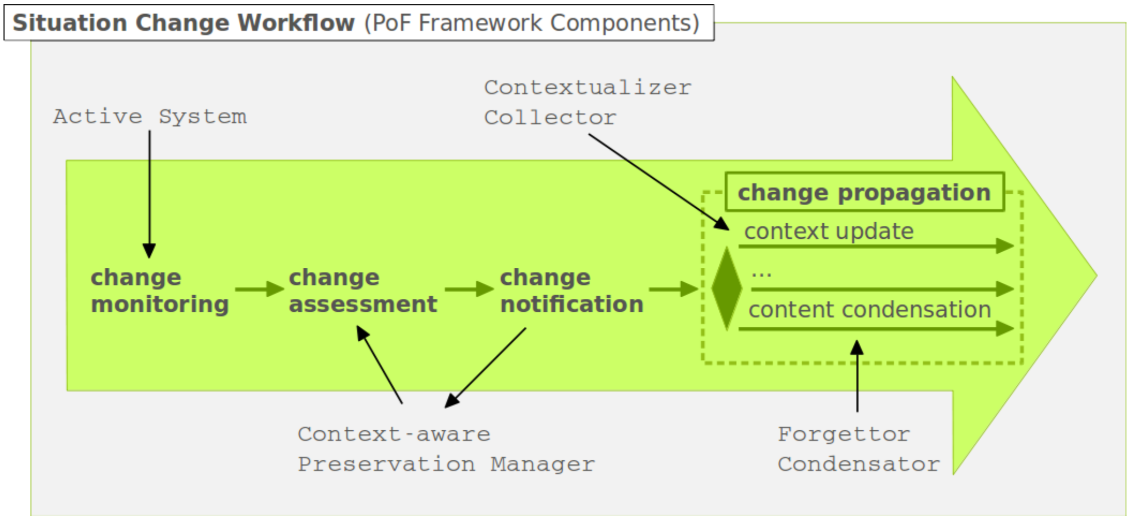

Figure 17: Mapping between the PoF Framework Components and the Situation Change Workflow.

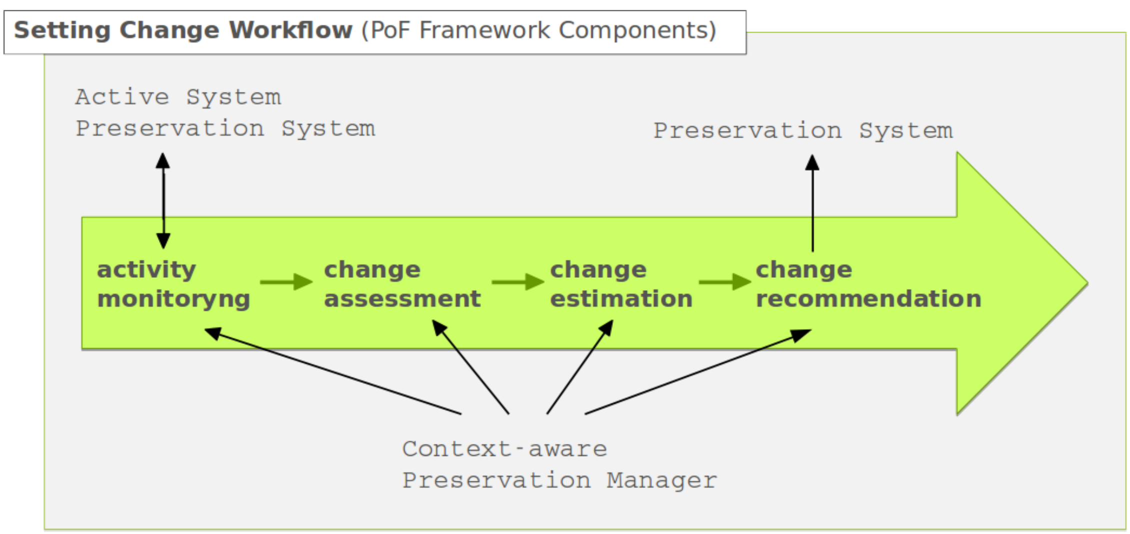

Figure 18: Mapping between the PoF Framework Components and the Setting Change Workflow.

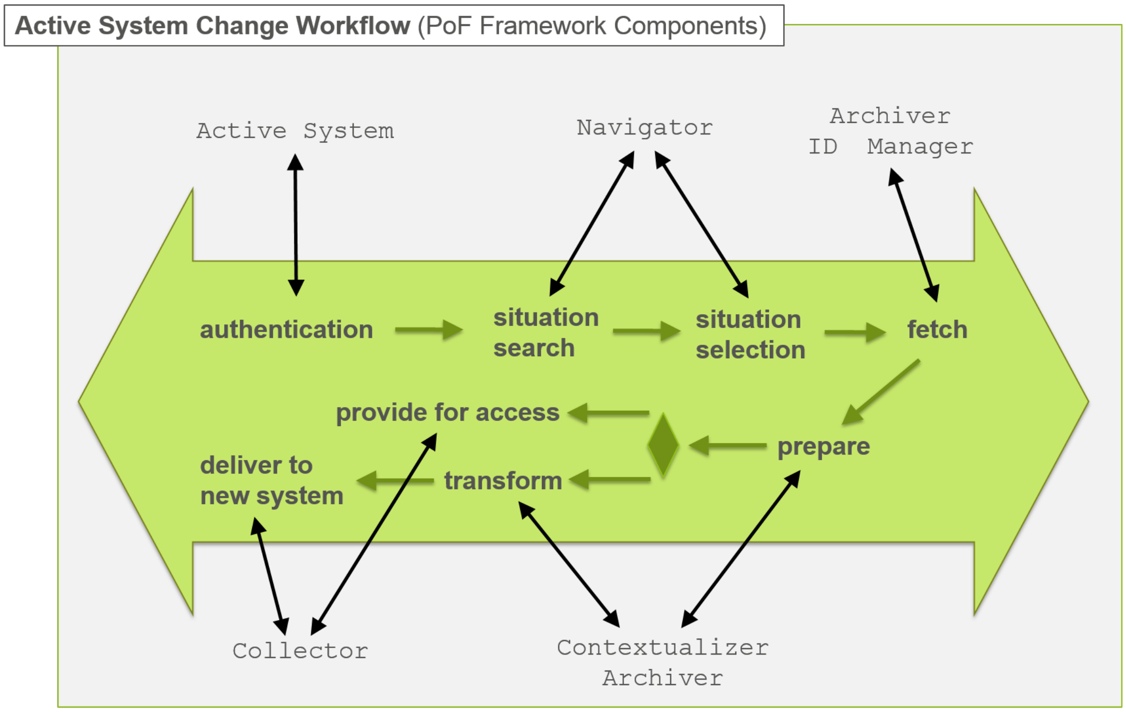

Figure 19: Mapping between the PoF Framework Components and the Active System Change Workflow.

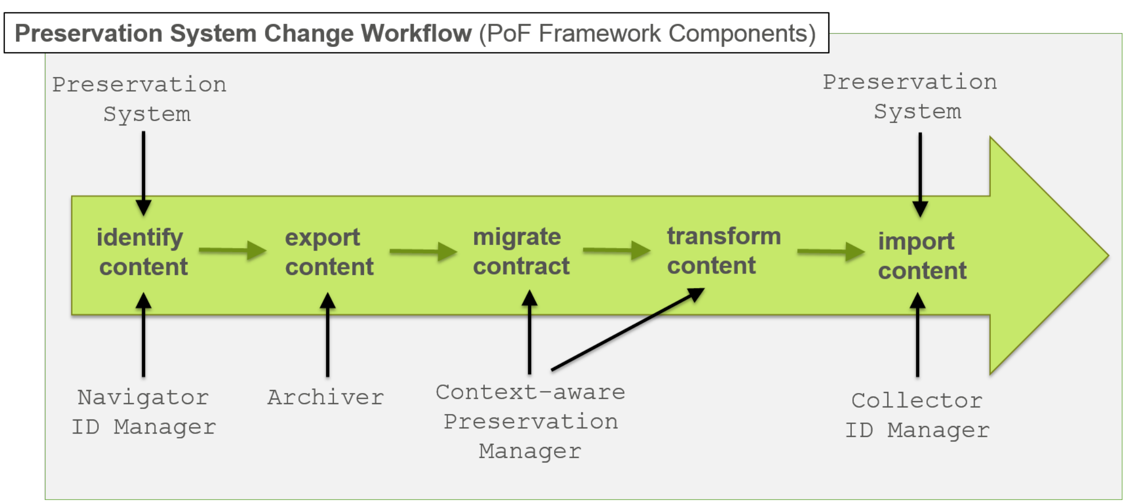

Figure 20: Mapping between the PoF Framework Components and the Preservation System Change Workflow.

6 Mapping to Digital Preservation Standards

In this Section we discuss the interoperability of the Information Model proposed in Section 4 and represented in Figure 21 with other well established metadata information models that are currently widely used in digital preservation.

The scope is to demonstrate that the proposed Information Model is compliant to the main standards adopted for digital preservation metadata representation in order to be able to submit/import or extract/export digital items from digital archives/libraries already up and running. **Digital Preservation** makes use of information related to the content and according to identified preservation strategies the contents can be checked periodically, migrated to different digital formats, enriched or reorganized and the like. Less often we have to migrate also the digital preservation system or the digital library where the contents are stored and saved. Hence a mandatory requirement for digital preservation is to keep safe all the information related to preservation such as history of the actions performed on the contents, by whom and by which digital devices, etc. ForgetIT doesn't want to introduce new issues for digital preservation, then aims to allow the full interoperability and interchangeability between different digital repositories i.e. digital preservation systems.

This Section attempts to verify that at least some of the most widely used metadata models can be *translated* from/into our Information Model. It is a matter of defining the appropriated ontology mapping the Elements from one model to the others, but the core concepts must be supported.

First of all we checked the compliance to the OAIS specifications [CCSDS, 2012], because many preservation metadata are already coming out from OAIS guidelines.

Then we selected the following models[2] as mostly adopted in professional environments:

- PREMIS [Library of Congress, 2015], the Preservation Metadata Implementation Strategies from Library of Congress (LoC)
  The PREMIS Data Dictionary and its supporting documentation is a comprehensive, practical resource for implementing preservation metadata in digital archiving systems. The Data Dictionary is built on a data model that defines five entities: Intellectual Entities, Objects, Events, Rights, and Agents. Each semantic unit defined in the Data Dictionary is a property of one of the entities in the data model. The PREMIS Data Dictionary is a comprehensive, practical resource for implementing preservation metadata in digital preservation systems. The Data Dictionary defines preservation metadata that: Supports the viability, renderability, understandability, authenticity, and identity of digital objects in a preservation context; Represents the information most preservation repositories need to know to preserve digital materials over the long term; Emphasizes implementable metadata: rigorously defined, supported by guidelines for creation, management, and use, and oriented toward automated workflows; and, Embodies technical neutrality: no assumptions made

[2] we reported few lines describing the models, taken from their website respectively

about preservation technologies, strategies, metadata storage and management, etc.

- PROV-O [W3C, 2013], the Provenance Ontology from World Wide Web Consortium (W3C)
  PROV-O is a lightweight ontology that can be adopted in a wide range of applications. The PROV Ontology classes and properties are defined such that they can not only be used directly to represent provenance information, but also can be specialized for modeling application-specific provenance details in a variety of domains. Thus, the PROV Ontology is expected to be both directly usable in applications as well as serve as a reference model for creating domain-specific provenance ontologies and thereby facilitates interoperable provenance modeling.

- MP-AF [ISO/IEC, 2016], the Multimedia Application Preservation Format from ISO IEC SC29 WG11 Moving Picture Expert Group (MPEG)
  The MP-AF (ISO/IEC 23000-15) defines the Multimedia Preservation Description Information (MPDI), extending the concept of Preservation Description Information (PDI), providing metadata addressing the specific requirements for preserving multimedia content. MP-AF defines a metadata format that enables users to effectively exchange information (metadata) related to multimedia preservation operations and their outcomes. Typical examples include the description of integrity checking and related results, content migration from one system to another, replication of subparts or entire contents, content quality evaluation and related quality report, relationships between the source and output of any transformation process, etc. At the core of MP-AF is its data model definition provided through UML diagrams and formal descriptions and a normative XML-Schema implementation. The model has been harmonized with MPEG-21 Digital Item Declaration and the schema reuses considerable parts of existing MPEG technologies, most notably MPEG-21 and MPEG-7.

- CCDM [EBU, 2016], the Conceptual Data Model, from European Broadcasting Union (EBU).
  The EBU Class CCDM is an ontology defining a basic set of classes and properties as a common vocabulary to describe business objects, e.g. programmes, articles and other types of content, and their relations in the business processes of media enterprises. Examples are programmes in their different phases of creation from commissioning to delivery, their associated rights or publication events, etc. CCDM is a common framework and users are invited to, and should, further enrich the model with classes and properties fitting their needs more specifically. Properties for describing each of the objects can be found in EBUCore, or you are welcome to define your own. The CCDM has been purposefully designed as a minimum and flexible set of classes for a wide range of broadcasting applications, including archives, exchange and media service oriented production, semantic web and linked data. The CCDM specification combines several aspects from existing models and specifications into a common framework. It has been built over several EBU attempts to represents broadcasting as a simple logical model. It has benefited from EBU work in metadata modelling (P-META and EBUCore) and semantic web

developments. The distribution part has been designed to seek maximum mapping to TV-Anytime and the ”BBC Programmes Ontology”.

Figure 21 shows the attempt to map the aforementioned metadata information models onto our information model described in Section 4.

The presented mapping must not be considered exhaustive nor the only possible one. Other candidates can be defined as well as further data models can be selected. Nevertheless, in order to guarantee a good level of interoperability, the presented mapping can be considered a proof that most of the core Elements available in other preservation metadata models have here their corresponding Element.

The mapping has been described by UML [OMG, 2015] `notes` attached to the **Elements** with a dashed line.

These `notes` have different color according to the related model. A *Legend* with the meaning of the colors is reported in Figure 21 as follow:

- Core OAIS Elements are *pale rose*
- EBU CCDM Elements are *pink*
- MPEG MP-AF Elements are *pale green*
- W3C PROVenance Elements are *violet*
- LoC PREMIS core Elements are *pale yellow*

Together with the color we also reported the name of the metadata model as `namespace`, i.e. with the syntax commonly adopted in programming and markup languages, having the *name* then a colon *:* followed by the Element name. Some Elements can probably be mapped onto more than one Element of our Information Model. Actually it could be possible to create an intermediate **ontology** placed in the middle, as proposed in [Höffernig et al., 2011], that fit better the possible correspondences. If our information model presented in Section 4 will be interesting and exploited by a standardization body such as OASIS for the CMIS model [OASIS, 2013] we will work in defining the intermediate ontology for easing the mapping between metadata models. Figure 21 should be considered as a candidate mapping (a proposed one) that can be refined and extended. As the reader can see, we can summarise that the proposed PoF information model (the overall picture from the Middleware point of view) is fully compliant to the most relevant standard models currently adopted in digital preservation. Almost every Element of other schemes has a corresponding Element in our model. That guarantees that preservation descriptive information will not be lost. Preservation descriptive information will be kept and stored in the appropriate Element whatever changes of digital preservation system will be required. Migration from one archive to another will not create a preservation issue.

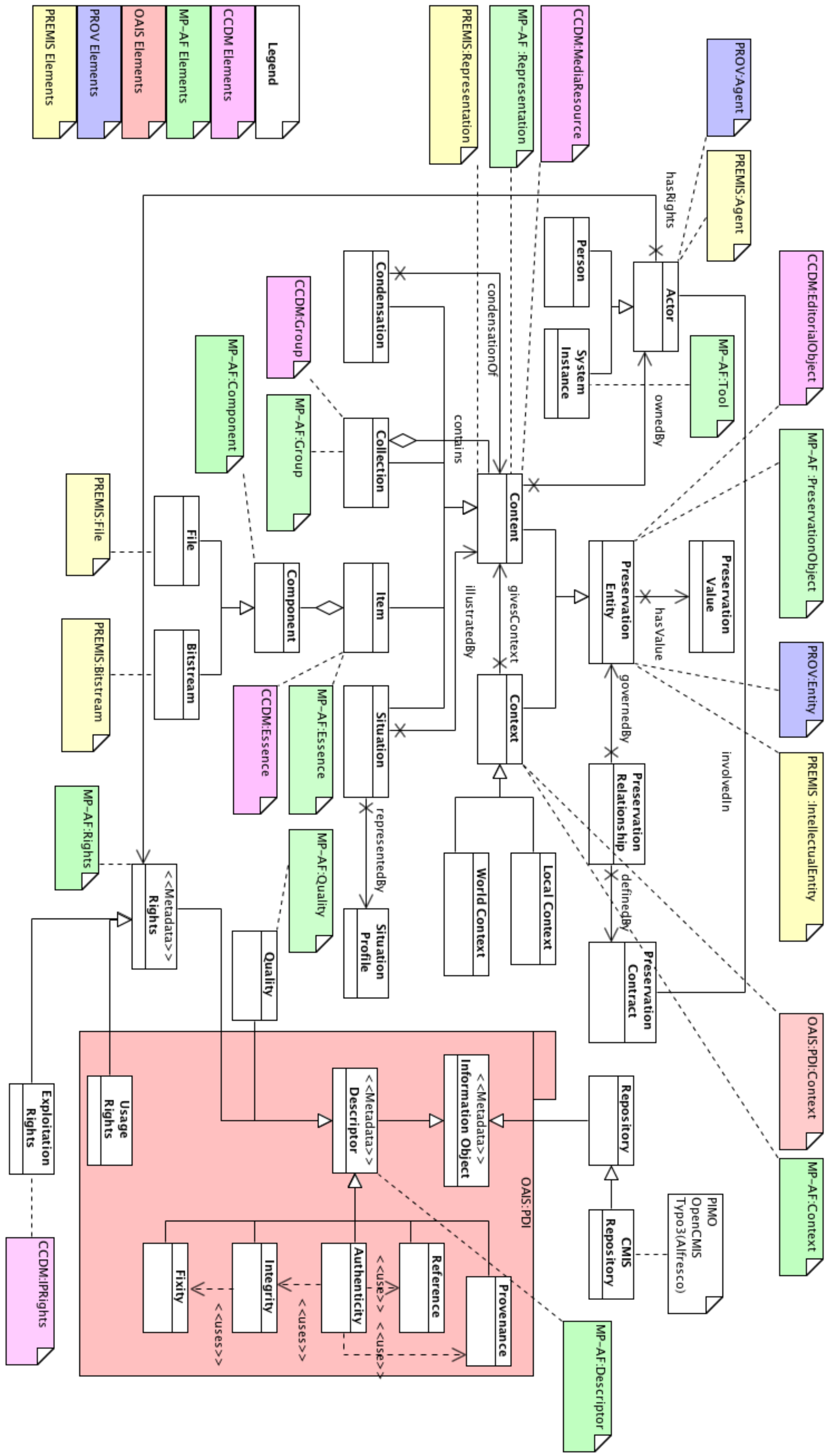

Figure 21: Information Model of the PoF Reference Model: mapping to well established and adopted digital preservation metadata information models. The namespaces adopted are described in the *Legend* with specific colors associated

7 Information Management Systems Extensions

The goal of the ForgetIT approach is to keep the impact of introducing preservation into the information management workflow as small as possible. Besides the long-term preservation of selected content - which is the aim - the approach also has the potential to introduce other more immediate benefits into active information management.

Both for the basic functionality of supporting preservation as well as for leveraging the benefit enabled by the approach, some extensions are required in the Active System.

These extensions are discussed in more detail in Section 7.1, while Section 7.2 elaborates on the benefit that can be created in the Active System. Finally, Section 7.3 focuses on preservation strategies, discussing different interactions of Active System and DPS.

7.1 Extensions of the Active System

Extensions to the Active System are required, where it has to interact with a DPS (possibly via a middleware as in the case of ForgetIT architecture) and where information has to be provided for the targeted intelligent preservation processes. This includes the collection of evidences for information value assessment and the collection of information in support of contextualization.

7.1.1 Supporting Information Exchanges

A core functionality, which needs to be enabled for synergetic preservation is *information exchange* between the Active System and the DPS.

Information to be exchanged includes the content to be preserved as well as metadata and context information describing this content. Furthermore, it has to be possible to bring content from the DPS back into the Active System (see Re-activation in Section 3.2). Thus, bi-directional information exchange has to be enabled.

Bi-directional exchange can be enabled for example by a repository used by both sides for making content available to the respective other system (plus possibly a notification channel). This approach is investigated in WP5 deliverables (see for example deliverables D5.1 [Nilsson et al., 2013] and D5.2 [Nilsson et al., 2014]).

We mention here two different approaches adopted in ForgetIT to exchange information with the Preserve-or-Forget (PoF) Middleware in order to show that different strategies are possible with the actual implementation. As a first example, we consider TYPO3 CMS, the Active System chosen in WP10 to implement the organization preservation use case. WP10 makes use of a standard-based repository leveraging the content exchange standard CMIS [OASIS, 2013], which enables the exchange between TYPO3

CMS and the PoF Middleware using a CMIS repository as intermediate (see also deliverable D10.3 [Dobberkau et al., 2015]). Besides these asynchronous channels, more synchronized forms of information exchange are also possible, such as direct service calls. This is done in the second approach, where we consider as a second example the PIMO Server, the Active System chosen in WP9 to implement the personal preservation use case. WP9 uses direct service calls between PoF Middleware and the Active System Semantic Desktop. There, CMIS is used as an exchange format for content objects which enables the PoF Middleware to retrieve content directly from the PIMO Server (see also deliverable D9.4 [Maus et al., 2015]). Nevertheless, both approaches use the PoF interfaces for communicating with the PoF Middleware such as registering content, preservation value updates, restore requests, etc.

7.1.2 Information Value Evidences

The Managed Forgetting & Appraisal function described in Section 3.3 heavily depends on the idea of content value assessment, which is discussed mostly in WP3 deliverables (see deliverables D3.3 [Kanhabua et al., 2015] and D3.4 [Zhu et al., 2016]). This is true for assessing short-term importance as it is done for computing Memory Buoyancy (MB) as well as for assessing the long-term value, named Preservation Value (PV).

For substantial content value assessment, evidences have to be collected from the Active System. For the short-term importance these are for example information about the usage pattern of a resource as well as information about the relationship between resources. In order to provide such evidences, specific interfaces as well as protocols are required defining which evidences are provided, in which format and in which frequency. Furthermore, methods for collecting such evidences have to be implemented in the Active System. In the opposite direction, the Active System might also profit from the computed content value information and use it for advanced functionalities and generate short-term benefits (see Section 7.2).

Because of these benefits, the Active System in WP9 does these assessments on its own which is detailed in Section 7.2. Therefore, the exchange is reduced to the calculated values such as the preservation value categories (see D8.6 [Gallo et al., 2016] for details of the interface).

7.1.3 Supporting Contextualization

Context information added to the preserved content is meant to ease the interpretation of such content in case of re-activation. The information contextualization is investigated in WP6 (see deliverables D6.1 [Greenwood et al., 2013], D6.2 [Greenwood et al., 2014], D6.3 [Greenwood et al., 2015], and D6.4 [Greenwood et al., 2016]).

Context information can also be gained in different ways, since (a) it can be provided by the Active System at the time content is sent to the DPS, (b) the PoF Middleware can automatically extract information from the provided content and other sources such as e.g. a domain specific ontology or external knowledge sources (e.g. Wikipedia) and (c) it can be a mix of the previous ones.

If this is possible in the considered Active System, harvesting context information which is already explicated in the Active System (option (a) above) looks more promising. In this way a richer and more quality-controlled form of context can be provided, with respect to what can be automatically extracted in the preservation process.

For example, in the WP9 Semantic Desktop, content is already annotated using an ontology, i.e. the Personal Information MOdel (PIMO). This annotation, obviously, is a good source of context information for preservation (see D9.4 [Maus et al., 2015]). Among those tools used, especially the Semantic Editor developed in WP4 allows annotation of textual content during writing which supports early contextualization by a user. The annotation vocabulary consist of above mentioned personal as well as external knowledge.

However, option (a) also puts higher requirements on the Active System: (1) explicated context has to be available (or it has to be explicated for this purpose) and (2) the Active System has to be extended with a functionality that is able to attach context information to the content information sent for preservation.

7.2 Creating Benefit in the Active System

Investment in preservation is typically paying back on the long run only. In order to foster the adoption of preservation technology, it is one of the goals of the ForgetIT approach to also create short-term benefit in the Active Systems - as a kind of positive side-effect of introducing preservation technology. This Section summarizes some ideas on how such side-effects can be created in the Active System based on the ForgetIT approach.

7.2.1 Forgetful Information Presentation

The functionality for Managed Forgetting & Appraisal and the related Content Value Assessment (see Section 3.3) can be used to distinguish between content, which is of current importance from other content (see discussion above for MB). The values of MB can be used in the Active System to bring the content currently important “closer to the user”, e.g. by showing it on the desktop or in special lists, having it on mobile devices, or by preferring search results with high MB, the so-called *forgetful search*. The core idea is to ease access to the things that are currently important: this is related to one of the five characteristics of the PoF Reference Model (*brain-inspired*), as discussed in Section 2.3 where we describe the digital working memory. This would be very similar to the human working memory, thus helping the user to focus on current activities. Examples for implementations of such forgetful information presentation can be seen in the Personal Preservation Pilots of WP9 (see deliverables D9.3 [Maus et al., 2014] and D9.4 [Maus et al., 2015]).

7.2.2 Forgetful and Archive-aware Search

Search is one of the core content access methods. Search & Navigation functionality (see Section 3.3) in the Active System can be affected in two ways by the introduction of preservation technology, as described in the following. The most obvious way is to smoothly integrate the archived content into the Active System search functionality, i.e. archive-aware search. This idea has already been discussed in more detail in Section 3.3, within the Active System extensions in support of archive-aware search have to be implemented. As a second way of modifying search in order to benefit from the ForgetIT approach, forgetful search (see above) can be introduced. The idea here is to take MB into account in the ranking function, thus preferring resources of relevance for the current task in the search result list (see different forgetful searches in D6.4 [Greenwood et al., 2016] and D9.3 [Maus et al., 2014]).

7.2.3 Creating Awareness for Content Value

On a more conceptual level, the idea of content value assessment as it is used for computing MB and PV in managed forgetting can also be used to raise the awareness for the value of content assets. An improved understanding of the value of content, which is based on a variety of factors such as investment, usage, popularity, etc. can become an important building block for next generation Content Management System (CMS) applications. This idea is discussed in more detail in the deliverables of WP10, most notably in D10.4 [Dobberkau et al., 2016]. In the lessons learned of WP9 (see D9.5 [Maus et al., 2016]) the idea to use the content value assessment also for evolutionary knowledge management is discussed.

7.2.4 Content Value Assessment in the Active System

The functional entity *Content Value Assessment* (CVA) is responsible for the assessment of resources in the PoF Framework as pointed out in Section 3.3. Considering the role of the functional entity CVA in the PoF Framework, the Semantic Desktop as Active System is an example of the situation where the Active System is capable of providing the PV for the preservation decision as well as the MB for Managed Forgetting in the Active System. In the light of the benefits discussed above, WP9 decided to embed the functional entity CVA into the Active System. In contrast to that, the implementation chosen for WP10's application scenario is an example of the situation where the Active System does not calculate the values by itself but rather delivers the evidences to the PoF Framework where the computation will take place.

Considerations on the decision taken in WP9

The design decision in WP9 was made because of the beneficial usage of both MB and PV in the Semantic Desktop infrastructure. The rich semantic model of the Personal Information MOdel (PIMO) and the usage statistics of the Semantic Desktop allow for a comprehensive view on the resources with respect to MB and PV. Furthermore, the nature of the PIM application scenario implies a lot of access, usage, and changes to resources and the PIMO resulting in a lot of traffic as well as content assessment in the PIMO as a knowledge base. Therefore, both values are computed in the Semantic Desktop and stored directly in the PIMO to be easily accessed by its components and thus, making them an integral part of the PIMO. To enable the PoF Middleware to make decisions based on the PV, the values are reported and updated in certain time intervals to the PoF Middleware (for details see D9.4 [Maus et al., 2015] and D8.6 [Gallo et al., 2016]).

7.3 Preservation Strategies

When preservation is introduced into the content management life-cycle of an Active System, a variety of decisions have to be taken defining the preservation strategy to be used. This includes decisions about when to preserve and about the granularity of preservation. Furthermore, the interaction between resource versioning and the preservation of a resource has to be defined.

Preservation actions can be triggered in different ways. They can for example be activated by the content management life-cycle: resources might be considered for preservation, when they go out of active use (low MB) or already upon creation or import into the system (e.g. for very valuable resources). Furthermore, preservation can be scheduled, for example, by queuing all resources above a predefined PV threshold for preservation on a regular basis. Finally, it is of course also possible to manually trigger preservation actions for individual resources or resource collections.

The choice of strategy is also dependent on the type of resources considered as well as on the level of control the user wants (or needs) over the preservation process. This could include enabling identification and removal of duplicate objects, or removal of objects with poor quality, as well as if there should be (or has to be) any transformation of objects already at ingest. Decisions has to be made on which of the options are best suited for an Active System under consideration. The options chosen influence the way the preservation process is integrated into the Active System beyond enabling the transfer of content to be archived.

Besides deciding *when* to preserve, it is also necessary to decide *what* to preserve. This can be considered along two related dimensions. First, it is possible to either preserve individual resources or entire collections of resources (or other types of complex objects such as sets of related resources) as one unit of archival. Second, resources can be preserved in isolation or together with context, which describes them. This second point is closely related to the work on contextualization in WP6 and the results affect the definition of the archival objects, the basic units of the PoF Information Model (see Section 4).

The choices with respect to granularity of preservation has consequences for the transfer protocols between Active System and DPS. In addition, it might require methods for selecting (extracting/collecting) relevant context information for a resource to be preserved.

One further aspect of granularity is the possibility to use different service providers for different types of objects. Image objects could for example go to a specialised image preservation provider, while regular documents go to a general preservation service, and moving images (and sound) to a media archive. This is supported by the Preservation Contract, Exchange Support, and ID Management, running in the PoF Middleware which gives the Active System one single point of contact for interacting with several DPS.

An interesting further aspect of the preservation strategy is to think about the model of co-existence between the copy of the resource in the archive and the resource in the Active System (if the strategy allows for such a co-existence). This has some implications, when the resource is changed in the Active System, after a copy has been archived. Typically, newer versions will not overwrite the archived version. Rather, the changed version will be archived as a new version. There are, however, decisions to be made about, if and when the updated version is put (automatically) into the archive. In any event, there should be a possibility to define how to handle copies in the DPS and if/when the version in the archive should be overwritten/updated, or versioned.

Last but not least, the preservation strategy is also to be considered for a service provider when defining the business model for a preservation service (e.g., also discussed in D11.4 [Akşener et al., 2015]). The strategy will influence service costs including traffic and storage as well as quality of service offered to the customer.

8 Conclusion

In this deliverable we described the final version of the PoF Reference Model, based on the ForgetIT novel approach to digital preservation, inspired by human forgetting and remembering. The model described here aims to provide the basic terminology, concepts, an core processes inspired by the three core principles: synergetic preservation, managed forgetting and contextualized remembering. The model also supports the implementation of a forgetful approach to preservation in the PoF Framework.

We identified five main characteristics for the model: integrative, value-driven, brain-inspired, forgetful and evolution-aware. For each characteristic we discussed the relationship with other project outcomes. Following the OAIS approach, we have defined a functional and a information part, identifying the relevant workflows, the functional entities and the information objects and discussing the relationship with the PoF architecture.

The functional part of the PoF Reference Model is made up of three layers: Core Layer, Remember & Forget Layer and Evolution Layer. For the information part we described the main entities (Situation, Collection, Item) and their relationship with other entities from a user and middleware perspective. For the information model we also discussed the relationship with other relevant standards used in digital preservation, which guarantees the interoperability of our model with other models adopted by potential adopters of the project results.

The final release of the framework implements the reference model described here and was used for validating it. In this document we also provided a mapping between the functional entities and the PoF Framework components.

Finally, we included a preliminary discussion about the extensions of an information management system (Active System) implied by the reference model and the possible impact of introducing the preservation into such an information system. According to the principle of synergetic preservation, this should enable a smooth transition between active use and preservation by making intelligent preservation processes an integral part of the content life-cycle in information management and by developing solutions for smooth bi-directional transitions. This is also related to creating immediate benefit from adopting preservation (not only on the long run) and can drive the preservation strategies when choosing the appropriate preservation objects and managing the co-existence of the resources in the DPS and in the Active System.

In the following sections, we briefly discuss the assessment of the results presented here, according to the WP8 performance indicators in the project proposal and then describe the lessons learned and the vision for the future.

8.1 Assessment of Performance Indicators

The expected WP8 outcomes, reported in the project proposal, are:

- the Preserve-or-Forget (PoF) Reference Model
- the PoF Framework

The results described here refer to the first outcome, for which the following success/progress indicators have been identified in the project proposal:

1. *availability of reference models and degree of adoption in the different contexts covered by each work package*
2. *availability of adequate "formats" for content and metadata*

The model described here represents the final release of the PoF Reference Model. We consider the results achieved so far compatible with the expected progress and success indicators for WP8, although further investigation and dissemination in the digital preservation community would be beneficial for the validation of the model and is also planned by publishing it.

Indicator 1: Reference Model

The PoF Reference Model has been created from scratch collecting the inputs from all work packages while taking into account the recommendations after each project review, mainly for what concerns the role of OAIS in the PoF Framework and the need to go beyond such model to support the novel ForgetIT approach to preservation.

After the first year only some preliminary ideas and a draft approach were available. During the second year the model foundations were identified and the functional part was specified. During the third year the functional part was improved (although the core ideas of the three layers were not affected) and a new information part was introduced and implemented in the framework, providing a complete and coherent definition of the preservation objects, their interaction and evolution within the PoF Framework. The PoF Framework architecture described in D8.1 is compliant to the model.

The PoF Reference Model is available as expected in this indicator and is based on the contributions from all work packages, as described for example in the model foundations.

Indicator 2: Content and Metadata Formats

The information model provides a representation of the main preservation entities and their relationships from the user and middleware perspective, providing a guideline for the choice of the appropriate content and metadata formats used to preserve the content generated in the different user applications.

In deliverable D8.6, when describing the different PoF architecture components, we also mentioned the different content and metadata formats supported by each component. As a result, we also have identified a set of reference technologies used for the implementation of the different information model entities. For example the adoption of the CMIS standard enables the content and metadata exchange between the user applications and the framework, while the integration with the middleware is based on typical RESTful services protocols, such as XML and JSON. Other standards used for content packaging and archival: examples include METS, DublinCore, MODS and PREMIS. The preservation entities (including situations, collections and items defined in the model) are represented using these formats. The context representation leverages different standards used for the ontologies, such as RDF and its serializations (such as Turtle) or in general custom XML representations. Other examples are available in deliverable D8.6. The aforementioned formats are suitable to support the representation of contents, their exchange between user applications and preservation systems, and their evolution over time.

Finally, we also discussed the interoperability of the information model with relevant digital preservation standards and other models, which enables the adoption of the PoF model with existing formats and technologies.

8.2 Lessons Learned

The definition of the PoF Reference Model required deep discussion among all partners during the whole project lifetime: the new developments and discoveries from each WP provided new insights into the ForgetIT approach, while the feedback collected when disseminating the project and during the project reviews were helpful in improving the model specification.

One of the main challenges was defining an information model which could be potentially adopted by the different digital preservation stakeholders, including end users (producers and consumers), archivists, and in general technologists implementing digital preservation solutions. The application pilots (WP9 and WP10) and the analysis conducted in WP2 shed light on the challenges related to personal and organizational preservation and influenced the design of the model.

Moreover, integrating the PoF approach with existing standards and approaches adopted in the digital preservation community was not easy, due to the innovative approach to digital preservation defined in ForgetIT and also to the variety of technologies and solutions adopted in each specific community.

Finally, compared to other established models such as OAIS, the effort and lifetime of the project were to limited to achieve a fully validated model. However, we believe that the PoF Reference Model developed by the project is a good starting point for future research in digital preservation.

8.3 Vision for the Future

The final version of the PoF Reference Model includes a new information model which provides a representation of the archival entities in the context of ForgetIT. This addition has been done in parallel with the design and development of the reference platform and the availability of the prototype deliverables from technical WPs. The final release of the PoF Framework, described in D8.6 [Gallo et al., 2016], implements the information model and many of the reference workflows defined in the functional model.

The project partners investigated various opportunities to disseminate the model in the digital preservation community and, following the recommendations of the project reviewers, also proposed the PoF Reference Model to different working groups in two standardization bodies, namely OASIS and MPEG (ISO/IEC JTC 1/SC 29/WG 11). Such activities are still in progress at the moment of writing since the standardization process could not be completed within the limited project lifetime. However, different partners in the consortium are interested in supporting such standardization efforts beyond the end of the project. Concerning PoF model dissemination, further publication opportunities in relevant scientific journals related to digital preservation have been considered by the partners. The submission of long papers describing the model and the ForgetIT approach is planned.

We believe that the PoF Reference Model together with the PoF framework as a model implementation, provides a strong basis for implementing and further developing the Forgetful approach to preservation. The feedback received at various occasions such as the CeBIT 2015 and CeBIT 2016 sugests that there is actually a strong need for solutions based on the approach, e.g., in the personal preservation setting.

9 References

[Afrasiabi Rad et al., 2014] Afrasiabi Rad, P., Nilsson, J., and Päivärinta, T. (2014). Administration of Digital Preservation Services in the Cloud Over Time: Design Issues and Challenges for Organizations. In Endicott-Popovsky, B., editor, *The Proceedings of the 2nd International Conference on Cloud Security Management*, Proceedings of the international conference on cloud security management. Academic Conferences and Publishing International Ltd.

[Akşener et al., 2015] Akşener, F., Sözen, S., Niederée, C., Maus, H., and Nilsson, J. (2015). ForgetIT Deliverable D11.4: Personal Preservation as a Service: Opportunities, Challenges and Exploitation Plan.

[Allasia and Palumbo, 2015] Allasia, W. and Palumbo, E. (2015). A Complex Network Model of Semantic Memory Impairments. In *2015 IEEE International Conference on Multimedia & Expo Workshops, ICME Workshops 2015, Turin, Italy, June 29 - July 3, 2015*, pages 1–6.

[Andersson et al., 2014] Andersson, I., Lindqvist, G., and Randers, F. (2014). Preservation Services Planning: a Decision Support Framework. In *The Proceedings of the 2nd International Conference on Cloud Security Management*.

[Baddeley, 1999] Baddeley, A. (1999). Essentials of human memory. *Psychology Press*, pages 139–144.

[Baddeley et al., 2009] Baddeley, A., Eysenck, M., and Anderson, M. (2009). Memory. *Psychology Press*.

[Borghoff et al., 2006] Borghoff, U., Rdig, P., and Scheffczyk, J. (2006). *Long-Term Preservation of Digital Documents, Principles and Practices*. Springer-Verlag Berlin Heidelberg, online-ausg. edition. by Uwe M. Borghoff, Peter Rdig, Lothar Schmitz, Jan Scheffczyk.

[Bote et al., 2012] Bote, J., Fernandez-Feijoo, B., and Ruiz, S. (2012). The Cost of Digital Preservation: a Methodological Analysis. *Procedia Technology*, 5:103–111.

[CCSDS, 2004] CCSDS (2004). Producer-Archive Interface Methodology Abstract Standard (PAIMAS) - Recommended Practice, CCSDS 651.0-M-1 (Magenta Book) Issue 1. Equivalent to ISO 20652:2006. `http://public.ccsds.org/publications/archive/651x0m1.pdf`. Retrieved March 2016.

[CCSDS, 2012] CCSDS (2012). Reference Model for an Open Archival Information System (OAIS) - Recommended Practice, CCSDS 650.0-M-2 (Magenta Book) Issue 2. Equivalent to ISO 14721:2012. `http://public.ccsds.org/publications/archive/650x0m2.pdf`. Retrieved March 2016.

[Chappell, 2004] Chappell, D. (2004). *Enterprise Service Bus*. O'Reilly Media, Inc.

[Conway and Loveday, 2010] Conway, M. A. and Loveday, C. (2010). Accessing Autobiographical Memories, the Act of Remembering: Toward an Understanding of How We Recall the Past. *Wiley Blackwell*, pages 56–70.

[Dobberkau et al., 2015] Dobberkau, O., Goslar, J., Gsedl, I., and Wolters, M. (2015). ForgetIT Deliverable D10.3: Organizational Preservation Pilot Application V2.

[Dobberkau et al., 2016] Dobberkau, O., Goslar, J., Gsedl, I., and Wolters, M. (2016). ForgetIT Deliverable D10.4: Organizational Preservation Report.

[EBU, 2016] EBU (2016). Class Conceptual Data Model (CCDM) - TECH 3351. `https://tech.ebu.ch/metadata`. Retrieved March 2016.

[Gallo et al., 2016] Gallo, F., Ceroni, A., Chen, T. T. D., Andersson, I., Greenwood, M. A., Maus, H., Lauer, A., Schwarz, S., Papadopoulou, O., Apostolidis, E., Pournaras, A., Collyda, C., Mezaris, V., Goslar, J., and Fogliati, M. (2016). Deliverable D8.6: Preserve-or-Forget Framework - Final Release.

[Gallo et al., 2015a] Gallo, F., Ceroni, A., Tran, T., Chen, D., Andersson, I., Greenwood, M. A., Maus, H., Lauer, A., Schwarz, S., Papadopoulou, V. S. O., Apostolidis, E., Pournaras, A., Mezaris, V., and Goslar, J. (2015a). ForgetIT Deliverable D8.4: The Preserve-or-Forget Framework - Second Release.

[Gallo et al., 2013] Gallo, F., Kanhabua, N., Djafari-Naini, K., Niederée, C., Rad, P. A., Andersson, I., Lindqvist, G., Nilsson, J., Henis, E., Rabinovici-Cohen, S., Maus, H., Steinmann, F., Mezaris, V., Papadopoulou, O., Solachidis, V., Dobberkau, O., Doan, P., Greenwood, M., Allasia, W., and Pellegrino, J. (2013). ForgetIT Deliverable D8.1: Integration Plan and Architectural Approach.

[Gallo et al., 2015b] Gallo, F., Niederée, C., Andersson, I., Nilsson, J., Chen, D., Maus, H., Greenwood, M., and Logie, R. (2015b). ForgetIT Deliverable D8.2: The Preserve-or-Forget Reference Model - Initial Model.

[Gladney, 2007] Gladney, H. M. (2007). *Preserving Digital Information*. Springer.

[Greenwood et al., 2013] Greenwood, M. A., Ceroni, A., Kanhabua, N., Mezaris, V., Niederée, C., Nilsson, J., and Papadopoulou, O. (2013). ForgetIT Deliverable D6.1: State of the Art and Approach for Contextualization.

[Greenwood et al., 2014] Greenwood, M. A., Ceroni, A., Mezaris, V., Niederée, C., Papadopoulou, O., and Solachidis, V. (2014). ForgetIT Deliverable D6.2: Contextualization Tools - First Release.

[Greenwood et al., 2015] Greenwood, M. A., Ceroni, A., Petrak, J., Gorrell, G., Mezaris, V., Papadopoulou, O., Solachidis, V., Eldesouky, B., and Maus, H. (2015). ForgetIT Deliverable D6.3: Contextualisation Tools - Second Release.

[Greenwood et al., 2016] Greenwood, M. A., Solachidis, V., Papadopoulou, O., Apostoliditis, K., Galanopoulos, D., Tastzoglou, D., Mezaris, V., Eldesouky, B., Maus, H., Tran, N. K., Hube, C., Niederée, C., Petrak, J., and Gorrell, G. (2016). ForgetIT Deliverable D6.4: Contextualisation Framework and Evaluation.

[Harvey, 2006] Harvey, R. (2006). Appraisal and Selection. In Ross, S. and Day, M., editors, *DCC Digital Curation Manual*.

[Harvey and Thompson, 2010] Harvey, R. and Thompson, D. (2010). Automating the Appraisal of Digital Materials. *Library Hi Tech*, 28(2):313–322.

[Heutelbeck et al., 2009] Heutelbeck, D., Brunsmann, J., Wilkes, W., and Hundsdörfer, A. (2009). Motivations and Challenges for Digital Preservation in Design and Engineering InDP 2009. In *Proc. of the First International Workshop on Innovation in Digital Preservation in conjunction with JCDL 2009*, Austin TX, USA.

[Höffernig et al., 2011] Höffernig, M., Bailer, W., Nagler, G., and Mülner, H. (2011). *Semantic Multimedia: 5th International Conference on Semantic and Digital Media Technologies, SAMT 2010. Saarbrücken, Germany, December 1-3, 2010. Revised Selected Papers*, chapter Mapping Audiovisual Metadata Formats Using Formal Semantics, pages 80–94. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Hohpe and Woolf, 2003] Hohpe, G. and Woolf, B. (2003). *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Hutař, 2013] Hutař, J. (2013). Archives New Zealand Migration from Fedora Commons to the Rosetta Digital Preservation System. *iPRES2013 Proceedings*.

[ISO/IEC, 2016] ISO/IEC (2016). FDIS 23000-15 - Information technology – Multimedia application format (MPEG-A) – Part 15: Multimedia preservation application. `http://www.iso.org/iso/catalogue_detail.htm?csnumber=66430`. Retrieved March 2016.

[Kanhabua et al., 2015] Kanhabua, N., Niederée, C., Ceroni, A., Naini, K. D., Kawase, R., Tran, T., Maus, H., and Schwarz, S. (2015). ForgetIT Deliverable D3.3: Strategies and Components for Managed Forgetting - Second Release.

[Kanhabua et al., 2013] Kanhabua, N., Niederée, C., Loggie, R., Tran, T., Djafari-Naini, K., Maus, H., and Schwarz, S. (2013). ForgetIT Deliverable D3.1: Report on Foundations of Managed Forgetting.

[Kanhabua et al., 2014] Kanhabua, N., Niederée, C., Tran, T., Nguyen, T. N., Djafari-Naini, K., Kawase, R., Schwarz, S., and Maus, H. (2014). ForgetIT Deliverable D3.2: Components for Managed Forgetting - First Release.

[Kejser et al., 2011] Kejser, U. B., Nielsen, A. B., and Thirifays, A. (2011). Cost Model for Digital Preservation: Cost of Digital Migration. *IJDC*, 6(1):255–267.

[Library of Congress, 2015] Library of Congress, P. E. C. (2015). PREMIS Data Dictionary for Preservation Metadata, Version 3.0. http://www.loc.gov/standards/premis/v3/index.html. Retrieved March 2016.

[Logie et al., 2014] Logie, R., Niven, E., Wolters, M., and Mayer-Schönberger, V. (2014). ForgetIT Deliverable D2.2: Foundations of Forgetting and Remembering - Preliminary Report.

[Logie et al., 2015] Logie, R., Niven, E., Wolters, M., Mayer-Schönberger, V., and Allasia, W. (2015). ForgetIT Deliverable D2.3: Foundations of Forgetting and Remembering - Intermediary Report.

[Marshall, 2011] Marshall, C. C. (2011). Challenges and Opportunities for Personal Digital Archiving. *Digital: Personal Collections in the Digital Era*, pages 90–114.

[Maus et al., 2014] Maus, H., Schwarz, S., Eldesouky, B., Jilek, C., Wolters, M., and Loğoğlu, B. (2014). ForgetIT Deliverable D9.3: Personal Preservation Pilot I: Concise Preserving Personal Desktop.

[Maus et al., 2015] Maus, H., Schwarz, S., Jilek, C., and Gallo, F. (2015). ForgetIT Deliverable D9.4: Personal Preservation Pilot II: Concise Preserving Mobile Information Assistant.

[Maus et al., 2016] Maus, H., Schwarz, S., Jilek, C., Wolters, M., Rhodes, S., Ceroni, A., and Gür, G. (2016). ForgetIT Deliverable D9.5: Personal Preservation Report.

[Niederee et al., 2015] Niederee, C., Kanhabua, N., Gallo, F., and Logie, R. H. (2015). Forgetful Digital Memory: Towards Brain-Inspired Long-Term Data and Information Management. *SIGMOD Rec.*, 44(2):41–46.

[Nilsson et al., 2014] Nilsson, J., Andersson, I., Rad, P. A., Lindqvist, G., Päivärinta, T., Rabinovici-Cohen, S., Maus, H., Dobberkau, O., Allasia, W., and Gallo, F. (2014). ForgetIT Deliverable D5.2: Workflow Model and Prototype for Transition between Active System and AIS - First Release.

[Nilsson et al., 2013] Nilsson, J., Päivärinta, T., Rad, P. A., Maus, H., and Dobberkau, O. (2013). ForgetIT Deliverable D5.1: Foundations of Synergetic Preservation.

[OASIS, 2013] OASIS (2013). Content Management Interoperability Services (CMIS) Version 1.1. OASIS Standard. http://docs.oasis-open.org/cmis/CMIS/v1.1/CMIS-v1.1.html. Retrieved March 2016.

[OMG, 2015] OMG (2015). UML - Unified Modeling Language, formal/2015-03-01. http://www.omg.org/spec/UML/2.5. Retrieved March 2016.

[Päivärinta et al., 2015] Päivärinta, T., Westerlund, P., and Nilsson, J. (2015). Integrating Contemporary Content Management and Long-term Digital Preservation: a Design Problem. volume 223 of *Lecture Notes in Business Information Processing*, Luleå University of Technology, Information Systems. Springer Verlag.

[Palumbo and Allasia, 2015] Palumbo, E. and Allasia, W. (2015). Semantic Similarity Between Images: a Novel Approach Based on a Complex Network of Free Word Associations. In *Similarity Search and Applications - 8th International Conference, SISAP 2015, Glasgow, UK, October 12-14, 2015, Proceedings*, pages 170–175.

[Rauch et al., 2013] Rauch, J., Braud, M., Edelstein, O., Rabinovici-Cohen, S., Nagin, K., Marberg, J., Voets, D., Sanya, I., Badawy, M., Shebab, E., Randers, F., Droppert, J., and Klecha, M. (2013). ENSURE: Long Term Digital Preservation of Health Care, Clinical Trial and Financial data. In Borbinha, J., Nelson, M., and Knight, S., editors, *iPRES 2013: Proceedings of the 10th International Conference on Preservation of Digital Objects*.

[Tuan et al., 2016] Tuan, T., Schwarz, S., Niederée, C., Maus, H., and Kanhabua, N. (2016). The Forgotten Needle in My Collections: Task-Aware Ranking of Documents in Semantic Information Space. In *Proceedings of the 1st ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR)*. ACM Press.

[University of California Curation Center (UC3), 2016] University of California Curation Center (UC3) (2016). Digital Curation FOundations. `https://wiki.ucop.edu/display/Curation/Foundations`. Retrieved March 2016.

[W3C, 2013] W3C (2013). PROV-O: The PROV Ontology - Technical Report. `http://www.w3.org/TR/prov-o/`. Retrieved March 2016.

[Zhu et al., 2016] Zhu, X., Niederée, C., Tran, T., Ceroni, A., Naini, K. D., Tran, N. K., , Maus, H., and Jilek, C. (2016). ForgetIT Deliverable D3.4: Strategies and Components for Managed Forgetting Final Release.

Glossary

CMS Content Management System. 64

DC Dublin Core. 36

DCMI The Dublin Core Metadata Initiative. 36

DPS Digital Preservation System. 7–12, 17–19, 21, 22, 24–26, 28, 30–32, 35, 49, 61, 62, 66, 67

EAI Enterprise Application Integration. 51

EIP Enterprise Integration Patterns. 51

MB Memory Buoyancy. 24, 26, 62–65

MOM Message Oriented Middleware. 51

PAIMAS Producer-Archive Interface Methodology Abstract Standards. 27

PIM Personal Information Management. 65

PIMO Personal Information MOdel. 62, 63, 65

PoF Preserve-or-Forget. 7–13, 17, 19, 20, 23, 30–33, 35, 37, 39, 40, 44, 46, 49–53, 61–70

PV Preservation Value. 24, 26, 62, 64, 65

Semantic Desktop is an approach which introduces a knowledge representation layer (the PIMO) by semantic technologies on top of information sources on the desktop and mobile devices and opening up the PIMO to applications and plug-ins of 3rd party applications. 62–65

UML Unified Modeling Language. 10, 58