SEVENTH FRAMEWORK PROGRAMME

ForgetIT

www.forgetit-project.eu

ForgetIT

Concise Preservation by Combining Managed Forgetting and Contextualized Remembering

Grant Agreement No. 600826

Deliverable D9.3

Work-package	WP9: Personal Preservation
Deliverable	D9.3: Personal Preservation Pilot I: Concise Preserving Personal Desktop
Deliverable Leader	Heiko Maus, DFKI
Quality Assessor	Doron Chen, IBM
Dissemination level	RE
Delivery date in Annex I	M23
Actual delivery date	M24
Revisions	1
Status	final
Keywords	Personal Preservation Pilot, Semantic Desktop, PIMO, Evaluation

Disclaimer

Revision History

Version	Major changes	Authors
0.1	initial version	Heiko Maus
0.2	added section on evaluation	Maria Wolters
0.3	outline of pilot documentation,	Heiko Maus
0.4	more on the pilot & photo organization scenario	Heiko Maus, Berker Loğoğlu
0.5	added section about PIMO Diary	Christian Jilek
0.6	finalizing structure, indicators, ex. summary; added Seed	Heiko Maus, Bahaa Eldesouky
0.7	added use case assessment for success indicators; final version for QA	Heiko Maus
0.8	included QA comments	Heiko Maus, Maria Wolters, Berker Loğoğlu
0.9	included 2nd round of QA comments	Heiko Maus
1.0	finalized with last QA comments	Heiko Maus

List of Authors

Partner Acronym	Authors
DFKI	Heiko Maus, Sven Schwarz, Bahaa Eldesouky, Christian Jilek
UEDIN	Maria Wolters
TT Argela	Berker Loğoğlu

Table of Contents

Executive summary

This deliverable documents the *Personal Preservation Pilot I* which is an outcome of ForgetIT's application scenario in work package WP9. The pilot implements personal preservation in a personal desktop setting based on the Semantic Desktop approach. It contributes to realize the ForgetIT goals of managed forgetting, contextual remembering, and synergetic preservation as well as takes advantages of ForgetIT's innovative methodological approach for information systems in the Semantic Desktop itself.

This deliverable gives an overview of the components of the pilot and is accompanied by a more detailed web-based documentation with explanatory text and examples such as videos of the pilot components.

The pilot is deployed in the ForgetIT testbed environment as a running system connected to the ForgetIT Preserve-or-Forget (PoF) Middleware and ultimately also to the preservation system.

With the pilot, the personal preservation application scenario is realized by connecting to the PoF Framework as well as using components from the technical work packages to realize the pilot's functionality. Most notably, these are memory buoyancy calculation and initial preservation policy (from WP3), semantic writing editor (from WP4), contextualization and condensation (from WP6), interfacing with preservation workflows and functionality (from WP5 and WP7), and PoF Middleware interaction (from WP8).

Furthermore, we introduce another application which supports users in the selection process for preserving photos without the requirement of an existing Semantic Desktop which is realized in close cooperation with WP3.

This deliverable also includes intermediate results of an evaluation of the long-term usage of the Semantic Desktop prototype by a group of 4 students. The students used the Semantic Desktop for their daily life. The evaluation is done in cooperation with WP2.

1 Introduction

This deliverable documents the *Personal Preservation Pilot I* which is an outcome of ForgetIT's application scenario work package WP9. This deliverable and pilot documentation builds on WP9 deliverables D9.1 [ForgetIT, 2013b] and implements mockups and functionalities described in D9.2 [ForgetIT, 2014e].

Based on the Semantic Desktop approach for Personal Information Management, this pilot implements personal preservation in a desktop computer setting. It contributes to realize the ForgetIT goals of managed forgetting, contextual remembering, and synergetic preservation. It also takes advantage of ForgetIT's innovative methodological approach for information systems in the Semantic Desktop itself.

This deliverable provides in Section 2 an overview of the components of the pilot supporting Personal Information Management of a user and shows how forgetting and preservation is realized in this pilot.

Furthermore, dedicated applications introduced for the personal preservation application scenario are explained: the PIMO photo organization for getting hold of photo collections (i.e., acquire[1] the photos for preservation) and richer contextual information for preservation in Section 2.5.1, the PIMO Diary for contextual remembering in Section 2.5.2, the semantic editor Seed for supporting semantic writing and early contextualization of texts throughout the whole Semantic Desktop in Section 2.5.3. Finally, an application which supports users in the selection process for preserving photos without the requirement of an existing Semantic Desktop in Section 3.

The Personal Preservation Pilot I is an implementation of the PoF Framework (see Figure 1; the technical details are explained in deliverable D8.1 [ForgetIT, 2014d]) with the Semantic Desktop as Active System and is built in accordance with the Preserve-or-Forget Reference Model as explained in deliverable D8.2 [ForgetIT, 2015d].

The pilot was made possible by the close cooperation of WP9 with all ForgetIT work packages. This resulted in successfully deploying and running of the pilot in the ForgetIT testbed environment connected to ForgetIT's Preserve-or-Forget (PoF) Middleware components and the Digital Preservation System which resembles the architecture depicted in Figure 1. Furthermore, several components from other ForgetIT work packages are used in the pilot. This allows for preservation of content on the computer with connection to the Semantic Desktop infrastructure as well as restoring from the archive.

Section 4 details intermediate evaluation results of a long-term usage of the Semantic

[1] As ForgetIT's goal is not to directly send material to the archive – as a standard preservation system would do – but rather get to know which resources are available, learn more about their value for preservation as well as their context, the Semantic Desktop first tries to 'get hold' of resources. This means technically, we acquire the following information: where the resources are located, what their content and context is, and how the resources are used and when the usage ceases. Therefore, the Semantic Desktop first acquires such information about a resource to be able to retrieve the resource and hand it over to a preservation system if it was decided to preserve that resource.

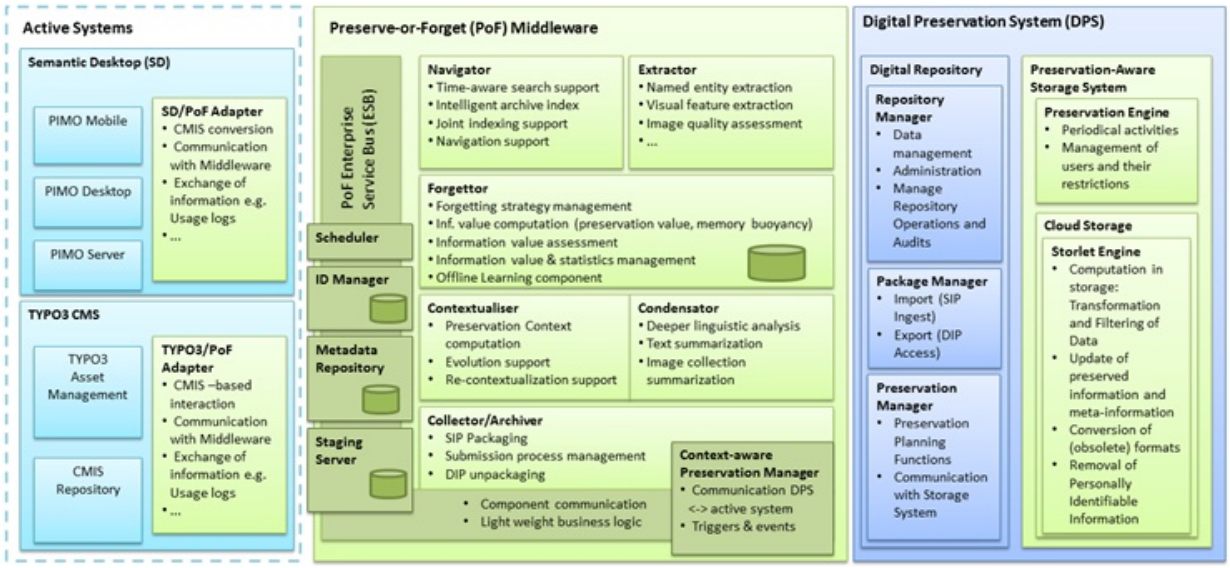

Figure 1: The Personal Preservation Pilot I is comprised of the Semantic Desktop as Active System and connects to the PoF Middleware and the Digital Preservation System.

Desktop prototype of a student group which used the prototype of the Semantic Desktop for their daily life, which was done in cooperation with WP2.

1.1 Target Audience

The target audience are all readers interested in how ForgetIT goals are realized in the personal preservation application scenario. Researchers from the Semantic Desktop field would get an insight on how the Semantic Desktop approach can contribute to the goals of ForgetIT as well as how forgetting, remembering and preservation can be beneficial for the Semantic Desktop itself. In addition, application developers could get an idea on how applications could contribute to an infrastructure which supports preservation. Finally, the online documentation allows the interested audience to get an insight into the ideas of ForgetIT with real examples.

1.2 Structure of the Deliverable

The deliverable is structured as follows: Section 2 gives an overview of the Personal Preservation Pilot I and explains the accompanying material for the pilot's documentation.

Section 3 details a dedicated application for selecting photos for preservation.

Section 4 addresses first results of a user evaluation and details next steps.

Section 5 provides the assessment of the performance indicators and points out next steps.

2 The Personal Preservation Pilot I

The pilot documentation in this section gives an overview of components of the pilot and their contribution to ForgetIT. To allow a more detailed explanation, this deliverable is accompanied by a web-based documentation of the pilot guiding the user through the pilot's functionalities by explanatory text and videos (screencasts of the pilot components) showing functions of the pilot.

The videos were taken from two installations of the pilot: First, the one on the ForgetIT testbed using the Stainer scenario and data set introduced in deliverable D9.1 [ForgetIT, 2013b] Section 2.4. Second, several real examples (such as forgetting and diary) were taken from the Semantic Desktop installation of the DFKI Knowledge Management department used in daily work.

This web-based documentation is delivered together with this deliverable as a zip-file containing html-pages and the videos[2]. However, in order to make this documentation publicly available, the documentation is available at a dedicated web page as shown in Figure 2. At the current time it is available at `https://pimo.opendfki.de/wp9-pilot/`. It will also be made available on the ForgetIT results page `http://www.forgetit-project.eu/en/project-results/`.

To allow to reference videos in this deliverable, the following naming convention is used: As each video is discussed in a separate section in the Pilot Documentation (PD), each video gets a number which derives from the succession of the section-numbers it is presented in:

PD.<section>.<subsection>.<subsubsection>

To navigate to the respective video start from the pilot documentation start page. On the start page, all sections and subsections are listed as presented in Figure 2. Click on the subsection, this will open a new page, and then scroll to the respective sub-subsection there. This works for the online as well as the zipped documentation.

In the upcoming deliverable D9.4, this online documentation will be extended with the components of the second release of the pilot. In the meantime, the online documentation will be extended if new features are introduced or if interfaces change.

With the pilot, the personal preservation application scenario is realized by connecting to the PoF Framework as well as using components from the technical work packages to realize the pilot's functionality. Most notably, these are memory buoyancy calculations and initial preservation policy from WP3, semantic writing editor (from WP4), contextualization and condensation (from WP6), interfacing with preservation workflows and functionality (from WP5 and WP7), and communication and PoF Middleware (from WP8). The technical details of their realization are explained in the respective WP deliverables.

[2] If extracted, please open the file `index.html` in a browser.

Figure 2: Online documentation of the Personal Preservation Pilot.

Several features were already introduced as mock-ups or early prototypes in deliverable D9.2 [ForgetIT, 2014e]. D9.3 shall be a comprehensive explanation of the personal preservation pilot, therefore, we build on explanations from D9.2 and extend them where appropriate. The online documentation presents further material to explain the motivations for the Semantic Desktop approach for ForgetIT.

The remainder of this section will start by providing a short introduction to the Semantic Desktop, then we explain the features of the pilot for Personal Information Management, afterwards we address the features especially dedicated to ForgetIT such as forgetting and preservation, and finally detail some dedicated applications addressing the ForgetIT use cases. Finally, an assessment of the Personal Preservation Pilot I along the use cases for personal preservation application scenario as identified in deliverable D9.2 is given.

2.1 Semantic Desktop in a Nutshell

In the following, we want to introduce the Semantic Desktop approach and the concept behind the Personal Information Model. The introduction is based on [Maus et al., 2013].

The Semantic Desktop approach was motivated by the observation that users, especially knowledge workers, would benefit from a support of their personal knowledge manage-

ment (for details on history[3] and motivation see [Sauermann et al., 2005] or [Dengel, 2012]) as the modern working environment places high requirements on knowledge workers: they are confronted with various applications, are involved in several projects and processes, work in changing teams, are on the road with a mobile office, and finally, face an ever increasing flow of information. The resulting knowledge spaces[4] are complex, dynamic, distributed over several applications, and use different vocabulary (i.e., set of names for, e.g., folders or tags). In this situation, it is hard to handle the complexity of the resulting personal knowledge space.

Therefore, the approach of the Semantic Desktop addresses this challenge. It follows the strategy to embed the mental model[5] of the knowledge worker in the daily work by means of a *Personal Information Model* ('PIMO', [Sauermann et al., 2007]). The user's mental model in the PIMO consists of concepts (called 'Things'[6] such as specific topics, projects, persons, tasks, etc.), associations between them (persons are member of projects, a task has topic 'Semantic Desktop', etc.), and finally, associated resources (documents, emails, web pages, pictures, etc.). Such resources are semantically represented in the PIMO (the Semantic Desktop community denotes this as 'rebirth': the resource is introduced with its semantic representation into the PIMO).

The PIMO serves as an easy to understand conceptualization of the knowledge worker's mental model, which can be used as a common vocabulary across different applications. Therefore, the PIMO provides the means required for a multi-criterial document classification considering the user's subjective view. Figure 3 shows an example of a representation of resources in the real world within the PIMO. The person Peter Stainer, his holiday trip as an event in a calendar, and the photo on his computer are resources in the real world.

These resources are represented in the PIMO as things which have different types (in knowledge representation terms these are instances of classes). The relations between the things can also be represented, for instance, that Peter attended the event and a photo has been taken on the trip. Which classes and relations are possible is defined in the PIMO ontology, which is a model of classes, their hierarchy, and allowed relations between them (as shown in the upper layer of Figure 3).

The resulting graph connects various resources, e.g., from the file system with items of a calendar application, with notes written by the user, with web pages about the holiday. The resources can further be associated with topics, locations, tasks, etc. In various applications, this graph then allows to find and access resources or things, to annotate, and to relate them.

[3]Most notably, the EU Integrated Project Nepomuk (The Social Semantic Desktop; Grant No FP6-027705) was a predecessor for the Semantic Desktop work in ForgetIT.

[4]A metaphor for the distribution of information and knowledge known by the user, contained in different documents, applications, and in the structuring of folders on the file system, as well as their interconnections which need to be handled by the user.

[5]The mental model is the representation of the world in the user's cognitive system as the user understands it. Therefore, the mental model differs btw. users.

[6]We keep this naming convention from the Semantic Desktop area as it is one step closer to an end-user when explaining that all their things are represented in the PIMO instead of talking about concepts.

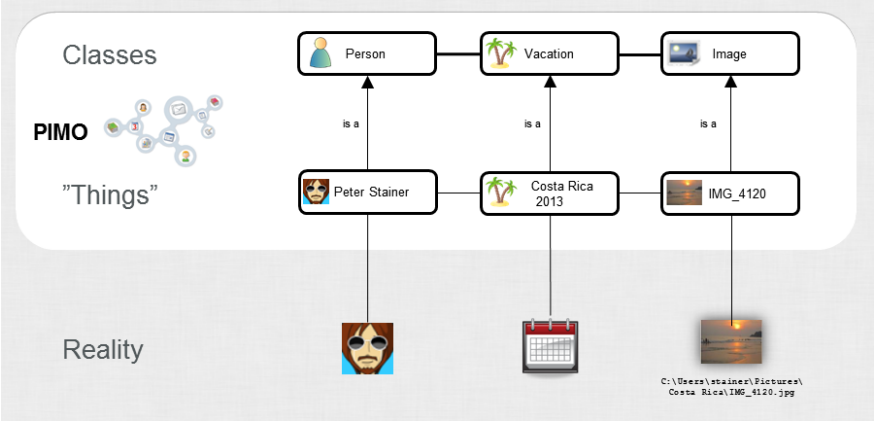

Figure 3: The layers of the Personal Information Model (PIMO) and their connection to the real world.

The PIMO uses the semantic power of the formal representation of the PIMO as an ontology [Sauermann et al., 2007], thus, introducing a knowledge representation layer on the user's computer. Besides enabling to annotate and interconnect resources over application borders, further semantic services are possible which make use of the semantic representation of the user's mental model in the PIMO.

This support for a user's Personal Information Management (resp. Personal Knowledge Management) – across various applications, embracing various resources, and connecting them in the PIMO, providing contextual information for value-added services – lets the Semantic Desktop approach be ideal for supporting the ForgetIT goals:

First, the Personal Preservation Pilot presents an advanced infrastructure for the Semantic Desktop enabling a plug-in architecture for in-situ PIMO access across different applications, thus, providing users one vocabulary for their work, regardless of the application or location (e.g., for tagging emails and files). This architecture allows ubiquitous access by storing PIMO data in the cloud, thus, allowing to apply sharing to group members such as in the office or in a family.

Second, the Personal Preservation Pilot embeds the Semantic Desktop as Active System in the Preserve-or-Forget Framework. By doing this the Semantic Desktop contributes to the ForgetIT goals, namely managed forgetting, contextual remembering, and synergetic preservation as well as takes benefits from this.

2.2 Personal Information Management in the Pilot

This section provides an overview of the features of the Semantic Desktop infrastructure for Personal Information Management contained in the pilot and their contribution to ForgetIT goals.

The Semantic Desktop infrastructure as part of the Personal Preservation Pilot consists of various applications and plug-ins for standard programs that provide support for the users

in every day life, and thus, in various scenarios identified for WP9. Furthermore, the resources, concepts, and the interaction with them provide evidences for services in the Preserve-or-Forget Framework in order to calculate memory buoyancy and preservation value (see deliverable D3.3 [ForgetIT, 2015a] for the technical details of the calculation). Furthermore, the ability of the PoF Framework to get insights on the user's relation to resources helps in the reduction of effort for actually preserving resources whenever this is embedded and supported by PIM applications.

Therefore, the Personal Preservation Pilot consists of components which support user's Personal Information Management. The support addresses the following areas and functionalities:

annotation support The PIMO provides with its contained concepts (such as topics, persons, events, projects, etc.) a vocabulary for users which can be used for annotating resp. tagging of various resources. By providing access to this vocabulary as a set of tags for third party applications, the pilot enables tagging with one vocabulary across different applications (instead of what is common today, where each application or service has its own set of tags). The pilot covers the following standard applications: web browser and email client (with the add-on FireTag for Mozilla Firefox and Thunderbird, see **PD.2.1.1**; also MS Internet Explorer), file system (SemanticFileExplorer for Windows, see **PD.3.1.1**), and annotation embedded in special-purpose applications such as photo organization (see **PD.4.1.2**, see also PIMORE in D9.2 [ForgetIT, 2014e] Section 3.3) or Seed (see Section 2.5.3 and **PD.7.1.1**).

file organization With the SemanticFileExplorer (SFE) it is possible to annotate, access, filter, and find files using the PIMO embedded in a user's normal desktop environment (see **PD.3.1.1**; currently only MS Windows is supported). With PIMOCloud[7] this is extended to cloud storage with file synchronisation across different devices, versioning, and sharing (see **PD.3.2.1**). Embedded in the SFE are various convenience methods such as browse the representation of a file in the PIMO (**PD.3.1.2**) or creating a photo collection from a folder (**PD.4.1.1**).

task management The Semantic Desktop infrastructure supports tasks management by introducing tasks as dedicated concepts with dedicated applications such as the Task Management application in PIMO5 (see **PD.5.1.1** and **PD.5.2.2**) which supports reminders, due dates, notes, annotations of concepts and resources (across all connected applications and plug-ins such as tagging an email with a task), etc. as well as connection and syncing capabilities with third party task management tools (see **PD.5.1.1** and **PD.5.2.1**).

calendar The Semantic Desktop also has events as specific concepts which allows to manage events and reminders within the PIMO and organize these with everything

[7]This is a service similar to DropBox or Google Drive, but embedded in the PIMO. Each file in the PIMOCloud is semantically represented as a resource in the PIMO.

else in the PIMO such as topics, persons, resources, or notes. By offering a CalDAV[8] endpoint, events can be synced and managed in third party calendar tools and devices supporting CalDAV such as Mozilla Thunderbird (with Lightning calendar; see **PD.6.1.1**) or the iPhone (see for tasks as part of CalDAV **PD.5.2.1**) without losing the power of the richer representation of events in the PIMO.

contacts Like with the previous *calendar*, the Semantic Desktop also offers a CardDAV[9] endpoint, thus allowing to access and exchange contacts in the PIMO (see **PD.6.2.1**), i.e., persons in the PIMO will be available via CardDAV resp. the Semantic Desktop can extract contact information from address books supporting CardDAV.

writing Taking notes, writing descriptive texts or whole documents is supported by a semantic text editor (Seed, see Section 2.5.3) which is embedded throughout the infrastructure if text has to be written such as notes (see **PD.7.1.1**), descriptions of photos (**PD.4.1.2**), or task notes.

collaboration The Semantic Desktop allows to have multiple users on one PIMO server (see, e.g., the student group in **PD.11.1.1**; the home screen with activities from other users in **PD.1.3.2**). Users have their own PIMO with private concepts and resources which can be shared with other users. This supports teams in an organization as well as a family setting. The infrastructure allows to share and reuse concepts and any kind of resources from photos to notes, calendar events to tasks. Users can subscribe to interesting things in a PIMO such as topics and get informed about activities of other users (see **PD.7.1.1**). This social aspect of the PIMO fosters its usage and the benefits of actually introducing material. Likewise, sharing of resources and interaction with shared resources of other users provide further evidences for preservation value.

2.3 Synergetic Preservation in the Pilot

The Personal Preservation Pilot is connected to the PoF Framework and allows to preserve files if either the PoF Framework decides to do so or preservation is triggered manually by the user. The pilot allows this preservation functionality, embedded in the file system of the user and supported by the Semantic Desktop, to easily connect a user's files to the PoF Framework's preservation workflow.

The PoF Framework accesses the CMIS[10] interface of the Semantic Desktop server to retrieve resources for preservation if it decides to preserve a resource (see for these details deliverable D8.3 [ForgetIT, 2014d]), the actual preservation process is not directly

[8] CalDAV is a standard internet protocol for accessing scheduling information on a server including events, alarms, and tasks.

[9] CardDAV is a standard internet protocol for accessing address book information on a server.

[10] Content Management Interoperability Service

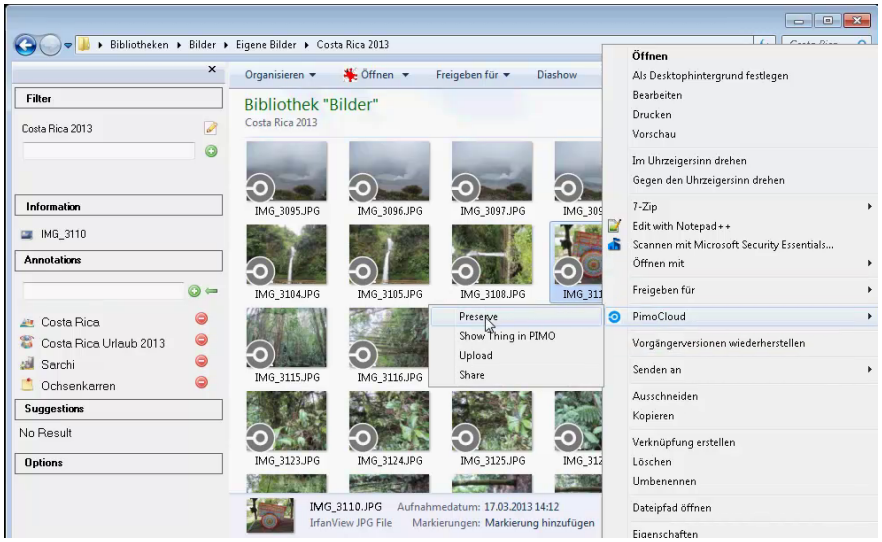

Figure 4: Preserving embedded in the MS Windows Explorer context menu of a file: issuing manual preservation (see PD.10.1.1).

visible in the end-user interface of the pilot (however, as part of the pilot this is visible in the middleware dashboard, see [ForgetIT, 2014d]).

As a use case, the pilot also enables a user to manually preserve resources (see **PD.10.1.1**). To do this, the user could choose a file through the file explorer, as shown in Figure 4, and select 'Preserve' in the file's context menu. Preservation of the file is issued without requiring any further interaction by the user with the PoF Middleware.

Technically, the manual triggering of 'Preserve' on one file results in that the preservation value of that file is set to maximum. Depending on the preservation policy, the PoF will then decide on the schedule to preserve. However, to visualize this, currently, the PoF Scheduler then immediately triggers a preservation workflow.

2.4 Forgetting in the Pilot

The Semantic Desktop, with its knowledge base, faces problems of information overflow if each and every event, tasks, note, web page, etc. would be kept and treated with the same importance as currently required information or important ones from the past.

In ForgetIT, we tackle this problem by following the example of the human forgetting (see also deliverable D2.2 [ForgetIT, 2014a] Section 2) which prevents humans from problems caused by too much information: a substantial amount of information is forgotten after a mere short time; the forgetting process continues over time, reducing more and more details. If not relearned, information will be hard to recall or eventually not be available anymore. Trying to remember will help in retrieving some details, however, other information might be lost forever. This view on forgetting is also applied in the concept and calculation of the memory buoyancy which is motivated and explained in detail in deliver-

able D3.3 [ForgetIT, 2015a].

The pilot provides two types of evidences for the calculation: First, by user events in case of interaction with resources such as 'create', 'view', or 'modify'[11]. Second, by the semantic representation of a resource in the PIMO and its connections to other resources which is used, e.g., for the spreading algorithm in the memory buoyancy calculation enabling it to decide where to spread with which weight to other relevant resources which are connected (see deliverable D3.3 [ForgetIT, 2015a] Section 2 *Memory Buoyancy*). This enables for memory buoyancy to be available in the pilot for every resource in the PIMO and for the values to be updated if the user interacts with the resources or as time elapses (such as decay, see again D3.3).

The pilot applies two kinds of forgetting strategies in case of low memory buoyancy: removal of resources and information hiding. In the following, we focus on how forgetting functionality is brought to the Semantic Desktop infrastructure in the pilot and how to remember resp. to restore forgotten resources.

2.4.1 Decluttering the Desktop by Applying Forgetting on the File System

With the help of the Semantic Desktop infrastructure being embedded on a computer's file system, the forgetting functionality is available on the user's desktop file system. Now it is possible for files to be removed from the computer if the PoF Framework decides to do so based on low memory buoyancy. Depending on the user's chosen policy, either the user will be informed of files to be removed or this will happen automatically in the background.

The Semantic Desktop checks at certain time intervals[12] if there exist local files whose memory buoyancy dropped under a certain threshold which would indicate that these should be forgotten. Identified files are proposed to the user to be forgotten, i.e., removed from the computer.

The current implementation of the pilot (see **PD.9.1.1**) employs a preservation policy which – after an OK from the user as shown in Figure 5, removes the file from the computer, but leaves the file in the PIMOCloud untouched. The preservation decision is left to the PoF Middleware. That means that details pertaining to what happens in the case of removal from the computer w.r.t. preservation will be clarified in a preservation policy (which is one topic of D3.3 [ForgetIT, 2015a]). For instance, the file in the PIMOCloud may be preserved and then deleted in the PIMOCloud, or the file will be preserved and kept in the PIMOCloud until the memory buoyancy drops below an even lower threshold. As long as the cloud file is available, it can be restored on any computer or accessed on a mobile device. If the cloud file is also removed, only restoring it from the archive will be

[11] i.e., the Semantic Desktop delivers an user action event which contains the type of user action, a timestamp, the user, and the URI of the Thing. This information is an evidence that the user did something with the resource which is used in the calculation.

[12] The pilot's interval can be set by a parameter. Reasonable time frames range from a daily basis – although no quick changes are to be expected – to a week.

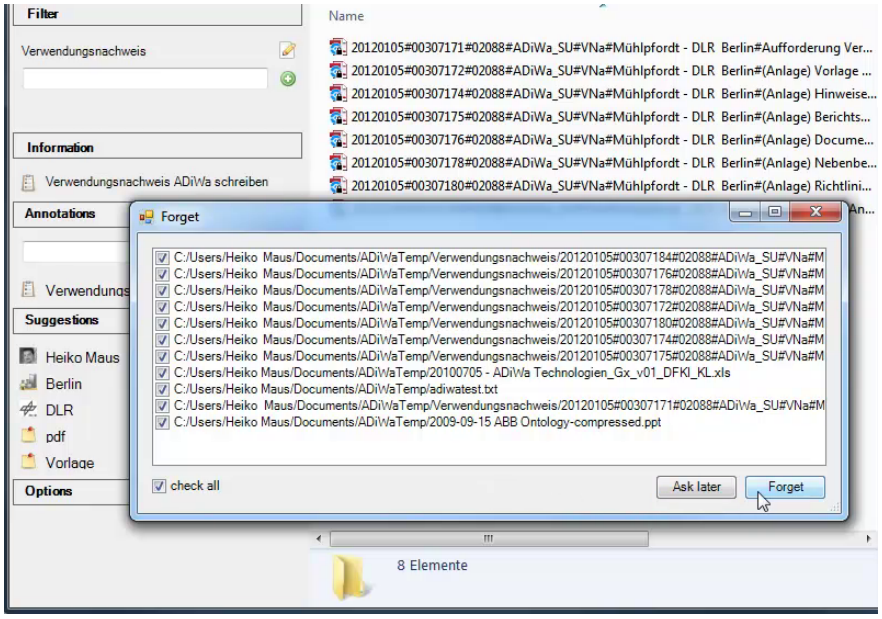

Figure 5: The PoF Framework has decided to forget several files on the user's computer. A dialog shows the user which ones and allows modification of the list before the forgetting is executed (see PD.9.1.3).

possible (if actually preserved there).

This strategy keeps high- and medium-buoyant resources on the desktop and gradually removes resources with low memory buoyancy from the desktop.

Although the files are forgotten, their semantic representation (i.e., the thing) in the PIMO is still there (see **PD.9.1.2**). However, due to the low memory buoyancy it will not show up in normal browsing (i.e., the user will not stumble upon it easily; see also Section 2.4.2). At what time such things shall be removed from the PIMO is subject of the user's policy of trashing things as well as condensing things into a landmark[13].

2.4.2 Hiding non-relevant Information by Forgetting in the PIMO User Interface

A further benefit of having memory buoyancy calculations is the possibility to actively hide things from the user when browsing the PIMO if the memory buoyancy is low. Not showing each and every resource in the PIMO, but only those of current and medium-term relevancy (as expressed by a high memory buoyancy) prevents the users from an information overload.

In the pilot, the PIMO5 user interface applies information hiding if the memory buoyancy (MB) of a thing is below a certain threshold (see **PD.9.2.1** for hiding during browsing and **PD.9.2.1** for hiding during search). This threshold is set differently for browsing on the

[13] As introduced in deliverable D9.2 in UC.3, the PIMO uses landmark concepts for condensing a set of connected concepts.

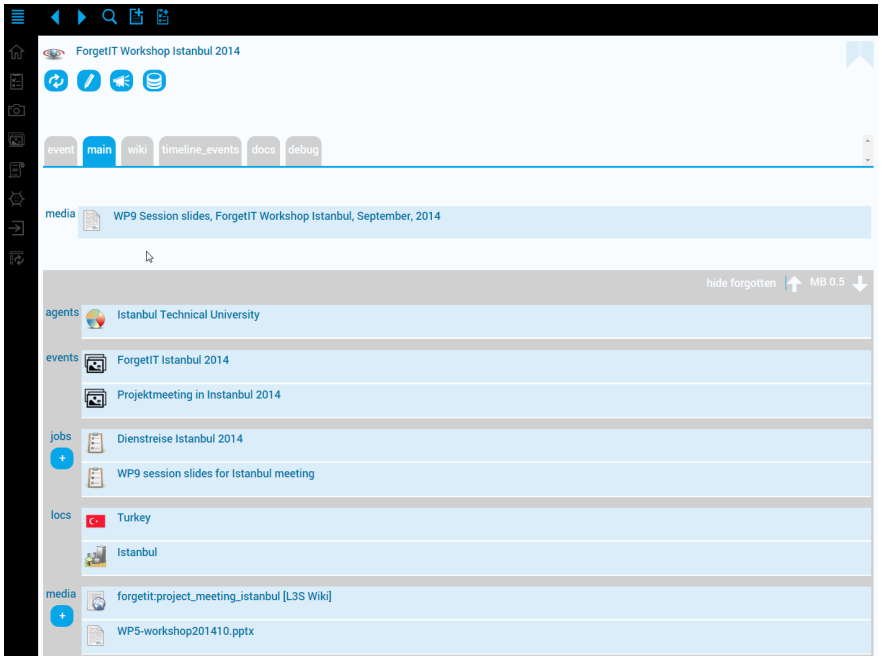

Figure 6: The PIMO5 GUI (a HTML5 client in the Semantic Desktop infrastructure) shows a view on the event ForgetIT Workshop in Istanbul with expanded 'show forgotten' (the grey part only visible on explicit request): only the slides are still shown (which are still in use); see also PD.9.2.1.

desktop or on a smartphone and can be adjusted by the user. Figure 6 shows an example of a view of a thing with a standard memory buoyancy threshold applied: the upper part shows only annotated concepts with higher MB than the threshold, in the grey part, all things are shown with lower MB. Things with low MB are usually hidden, and forgotten resources are shown only if the user issues an explicit request ('show forgotten').

These were the generic features of the Semantic Desktop in the Personal Preservation Pilot. In the following sections, we will address dedicated applications contributing to the personal preservation application scenario which are part of the pilot.

2.5 Dedicated applications & services

These are applications and services which support needs or desires of users. In the ForgetIT Framework, they are part of the Active System to provide benefits for the users and rely on ForgetIT components and features. In turn, again contribute to the goals of ForgetIT, e.g., by allowing to collect and organize photos which enable preservation by ForgetIT.

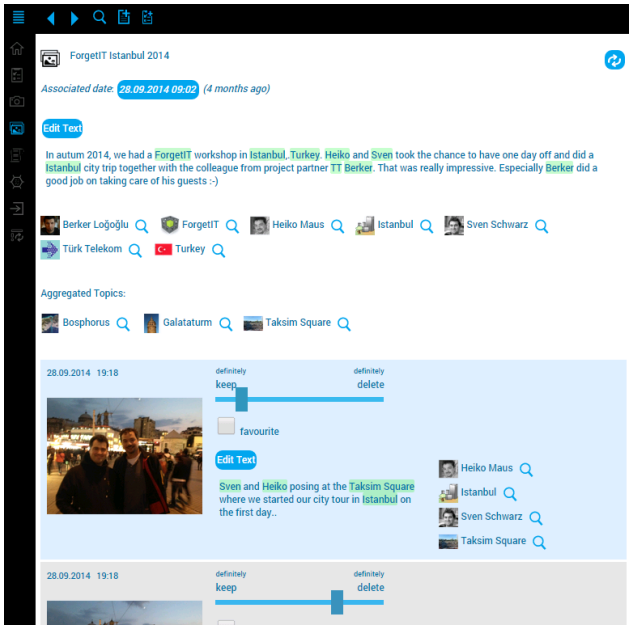

Figure 7: Photo collection in the PIMO: collection from Istanbul trip and ForgetIT workshop.

2.5.1 PIMO Photo Organization

The first user survey in D9.1 [ForgetIT, 2013b] showed that users would appreciate support in organizing their photos. Based on experiences with the first prototype for photo collections PIMORE (see deliverable D9.2 [ForgetIT, 2014e]), the Personal Preservation Pilot includes an application allowing users to organize their photo collections as part of the Semantic Desktop infrastructure and using the PIMO.

The idea for the photo organization in the Semantic Desktop, is to provide the user with an application that could be part of his usual Personal Information Management for the family. Because the application is part of the PoF Framework, evidences can be derived which contribute to ForgetIT's goals without much effort required by the user. Such evidences can be favourite photos, the event a photo belongs to, further annotations and descriptions of a photo, as well as further material connected to the collection which could provide even more contextual information on the event or on the photo itself.

The challenge such an application faces is how to motivate people to use and contribute information when organizing photos. The approach taken is first to provide a better experience than organizing photos simply by naming folders on the computer (or even keeping them on the camera).

The application which is part of the pilot (see Figure 7, **PD.4.1.2**, and **PD.4.1.3**) allows users to organize their photos by providing the user's mental model (as expressed in the PIMO) as well as details to activities and events (such as a trip) from the other sources in the Semantic Desktop (events, documents, notes, etc.). Furthermore, by using the mental model represented in the PIMO, we reduce effort for the user both for organizing photos as well as reusing the mental model constructed while organizing photos.

Photo organization software for home users such as Apple's iPhoto[14] already provide easy organization by events (in iPhoto an event is simply a set of photos with a label, with no connection to or interpretation as a calendar event), geo-location, and persons (via face detection). We extend this by allowing to annotate with PIMO concepts and resources, (semantically interpreted) texts (using Seed, see **PD.4.1.3**), condensation of events, etc. The PIMO introduces the concept of a life situation which can contain photo collections and may differ in its view depending on the specific type of life situation such as birth, wedding, or trip (though, this view is currently only available in the PIMORE prototype).

Furthermore, annotations added to one photo are aggregated for the photo collection. Combined together they comprise an aggregated description of the collection (as shown in Figure 7). And vice versa, annotations at the photo collection are propagated to the contained photos, which also helps finding photos by using annotations from the collection which are not separately added by the user to a photo, again, reducing effort for contextualizing each and every photo.

The resulting life situation with its photos is embedded in the PIMO, thus, services will be possible such as preserving, forgetting, reminiscence, cloud storage in the PIMO and mobile access, contextualizing and condensation, or associative retrieval of photos via PIMO's facetted search (see **PD.4.1.4**).

Please note that by using only this application for photo organization purposes and not using all other Semantic Desktop applications, we are already able to get hold of photo collections, can enrich these with contextual information, and enable selection for preservation by the PoF Middleware. This in contrast to the Photo Preservation Application in Section 3 which focuses on the selection for preservation.

2.5.2 Supporting Contextual Remembering with PIMO Diary

Using the Semantic Desktop regularly leads to a PIMO enriched with plenty of semantically annotated information, e.g., documents, web pages, emails, calendar events, etc. Sorting, mentally connecting and abstracting from parts of these things in order to remember what actually happened in a given period of time is typically a difficult and time-consuming task.

The PIMO Diary realizes contextual remembering by enabling a user to generate a personal (or group) diary based on these information items from the PIMO (see **PD.8.1.1**).

To support contextual remembering and at the same time to prevent the diary from being a confusing, large, sequential collection of material, we need to identify semantic relationships among possibly several thousands of individual information items and create suitable abstractions from them. If a user, for example, looks back on the last decade, he/she should not be overwhelmed with a view showing plenty of individual events, but compact statements like project names, stages of life, life situations, etc. Examples for those are terms like school years, studies, wedding or the name of a place where a vaca-

[14] http://www.apple.com/mac/iphoto/

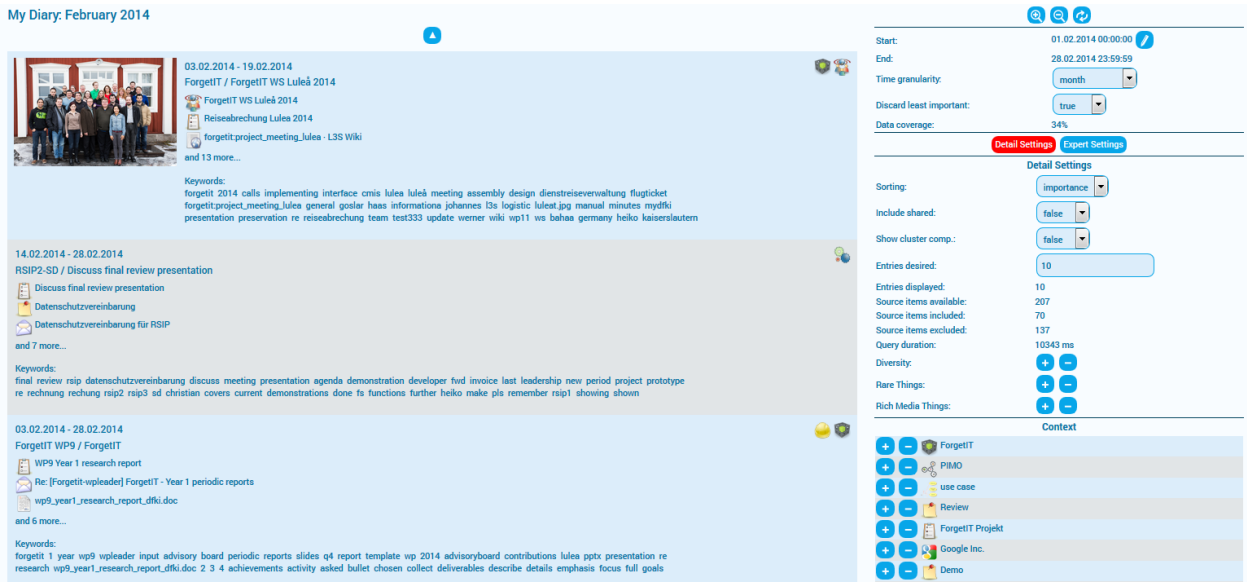

Figure 8: PIMO Diary showing February 2014 of a ForgetIT member using DFKI's PIMO instance in daily work (see PD.8.1.1).

tion or longer stay abroad has been spent. The user literally *zooms out* of an overwhelming mass of details. If desired, these abstractions can easily be resolved by selecting a sub-period of time for concretization (*zooming in*), e.g., a year of a decade or a month of a year. Concretizations (half-years, quarters, months, weeks, days) can be performed until the user reaches the actual (non-rehashed) basic material, which are the concrete information items like notes, photos, documents, etc. Clicking any of these items directly opens them in the appropriate app, e.g., notes are opened in the note app.

The system applies a combination of merging and filtering by clustering related or very similar things to diary entries and evaluating their importance for the user. The former aspect fosters a high diversity within the diary, making it interesting and fun to read, whereas the latter aspect is a necessity induced by the fact that the number of diary entries to be generated is usually limited. It is also compliant to ForgetIT's basic idea of forgetting less important, situative short-term information in favor of more important items selected for long-term preservation. In the example screenshot (Figure 8), we see that there are 207 information items available for the selected month. By using 70 of these items, ten diary entries were generated and displayed. The other 137 items were disregarded in the current diary view, which leads to a data coverage of 34%. Nevertheless, diaries covering 100% of the available data can be generated if desired (in these cases, the feature of sorting out items by their importance is turned off).

As depicted in Figure 8, a typical diary entry consists of a date (or time interval), a headline, and the most prominent things and keywords gathered from all information items that were clustered to form this entry. On an entry's right-hand side there are its most prominent annotations revealing more of its contextual background. Additionally, if a photo is associated with the entry (or more precisely: with one or more of its items), it is displayed on its left-hand side.

In the lower right of Figure 8, another feature called the 'concept context' (or 'context' for short) is illustrated. It should provide a quick overview of those things of a user's life (reflected by his/her PIMO) that concerned him/her the most in a given time period. For example, the ForgetIT project, the PIMO, use cases and reviews are among the most prominent things in the depicted sample month.

Users also have the possibility to incorporate shared data of their family, friends or colleagues - represented by a *group information model (GIMO)* - into their own personal diary turning it into a *group diary* (see **PD.8.1.3**). As a consequence, a friend's shared photo collection may appear as a separate entry in a user's own diary, or some of his/her entries may be complemented by additional information items coming from other people's PIMOs.

The web-based pilot documentation shows an example of a diary from a real PIMO and possible interactions with it (see **PD.8.1.1-8**).

2.5.3 Semantic Text Composition with Seed

As previously explained, at the heart of the Semantic Desktop is PIMO, which models the personal knowledge of the user. Interaction with the knowledge modeled in the user's PIMO takes place through various interfaces. One of the important interfaces is natural language text. This is where Seed – short for the *Semantic editor* – plays an important role. Seed is result of WP4 (see deliverable D4.3 [ForgetIT, 2015b]) and work in WP6 for early contextualization (see deliverable D6.3 [ForgetIT, 2015c]).

Seed is an extensible knowledge-supported web-first[15] natural language text composition tool. It builds upon state-of-the-art technology in the fields of NLP (Natural Language Processing), LOD (Linked Open Data) and Semantic Web technologies to provide a user-friendly way of interacting with complex knowledge systems.

Seed is integrated in multiple GUIs of PIMO (see **PD.7.1.1** and **PD.4.1.3**). It helps in exploring, modifying and creating semantically annotated textual content. It greatly reduces prerequisite domain knowledge and the accompanying cognitive load on users. Figure 9 shows a sample text being composed in Seed. Seed's main features are:

Annotate as you type: Users naturally compose text in a WYSIWYG[16] fashion, while Seed analyzes the content in real-time and annotates mentions of PIMO entities in a non-obtrusive way.

Immediate contextualization: By annotating the text with entities from the user's PIMO, Seed saves the context with the text allowing it to be better retrieved, recalled, preserved, and ultimately understood.

Proactive information delivery: Seed not only annotates text, it also presents infor-

[15] Although it can be integrated in other platforms, it first and foremost targets the Web.

[16] What you see is what you get: The final appearance of the text is directly shown to the user, e.g., headlines are shown in bold.

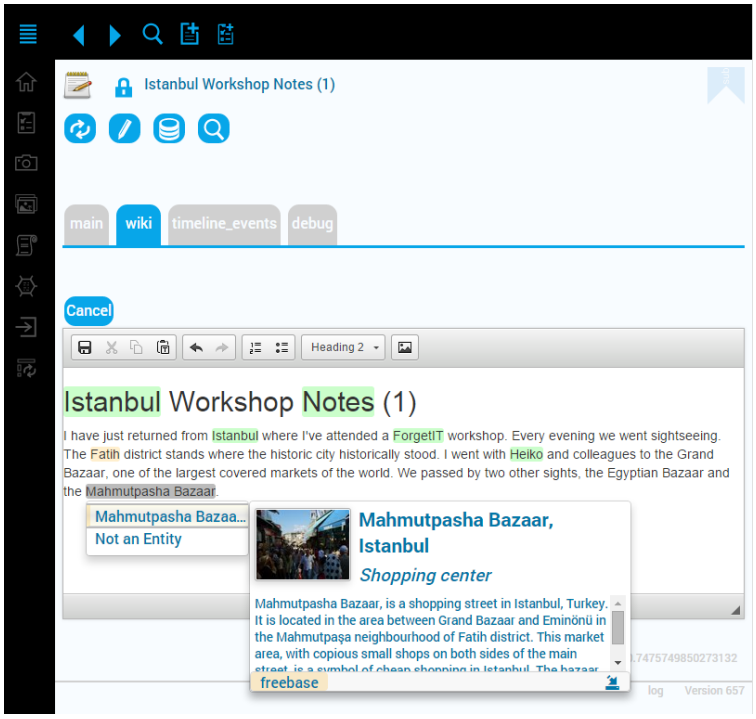

Figure 9: Text being composed using Seed (highlights in green: annotated entities from the PIMO; yellow: entities from external knowledge-bases; grey: unresolved entity detected); see PD.7.1.1.

mation about the entities annotated in an interactive way allowing the user to discover knowledge about things mentioned in the text.

Access to public knowledge: Entities mentioned in texts for which no entities in the PIMO exists, are also discovered by Seed by linking to entities from public LOD sources such as DBPedia and Freebase. Seed proposes such entities to the user (grey highlights in Figure 9). Users are able to import the entities found in those sources along with their metadata into PIMO, facilitating the growth of the personal knowledge model of the user.

Allow for correction: The ability to proactively annotate text as it is written does not subsume the user's ability to control annotations. All annotations made by Seed can be modified or rejected allowing the user full control over the semantic content.

Collaboration: Texts created and annotated in Seed can immediately be reused by others. This allows for many collaborative editing scenarios that make use of the semantic annotations. For example one user can create and annotate a text, which is later accessed by another user who reviews and modifies the annotations as well as the content of the text. (Please note that the annotations of private, i.e., non-shared, entities in texts stay private and are not visible to other users.) Furthermore, as the PIMO server also contains shared entities from other users as group knowledge, private texts can also be annotated with entities shared by other users.

2.6 Assessment of Use Cases in the Personal Preservation Pilot I

In deliverable D9.2 we identified several use cases which we want to address in ForgetIT for the Personal Preservation application scenario. In the following, we revisit those use cases and assess how they are supported by the Personal Preservation Pilot I. This contributes to the success indicators of WP9 as listed in ForgetIT's Description of Work [ForgetIT, 2012]: the use cases identified for personal preservation are assessed according to their progress and fulfillment which is a success indicator for WP9. The assessment of the indicators is done separately in Section 5.1.

The use cases (UC) and their assessment are listed as follows:

UC.<**id**> <**use case name**> <*short description*>
<use case assessment for Personal Preservation Pilot I>[17]
Priority: {**high, medium, low**}

UC.1 preserving resources *The Semantic Desktop automatically preserves valuable resources.*
By connecting the Semantic Desktop to the PoF Middleware, resources can be preserved if the PoF Middleware decides to do so. Then, the CMIS interface of the PIMO server is accessed by the PoF Middleware and resources can be retrieved for preservation. The decision which resources are valuable relies on the preservation value computed in the PoF Middleware. The Semantic Desktop infrastructure provides evidences for this decision using various applications and plug-ins.

Furthermore, allowing to trigger manual preservation on resources enables the user to impose his/her decision. Choosing preservation manually causes the as the preservation value is to be set to the maximum value, which is an indication for the PoF Middleware to preserve it (see for manual preservation **PD.10.1.1**).

Still to be done is to integrate preservation policies in the PoF Middleware which then decide, based on the policy and the computed preservation values, which resources should be preserved and when.
Priority: **high**

UC.2 forgetting resources *The Semantic Desktop hides resources from the user similar to a human forgetting.*
The pilot uses memory buoyancy to hide PIMO concepts and resources on the PIMO5 interface (see **PD.9.2.1** and **PD.9.2.2**). Likewise, files with low memory buoyancy values are removed from the file system (see **PD.9.1.1**).
Priority: **high**

UC.3 a landmark concept represents a set of connected concepts *The PIMO uses landmark concepts for condensing a set of connected concepts. The set of concepts*

[17] To keep the assessment explanations short, the text explaining the use case is not repeated here. The reader might refer for details of the use case to deliverable D9.2.

can then be hidden, and the landmark is then the representative of the set.
The PIMO Diary (see **PD.8.1.1**) is a step towards this solution: it computes a set of clustered concepts which represent an activity in the respective time frame. It generates an entry for this timeframe with a title, the most prominent resources, an image if one is available, and condensed text and context. The next step will be to store such entries permanently (i.e., rebirth in the PIMO) and also identify a hierarchy of such entries, ultimately resulting in a landmark.

Furthermore, to cover several candidates for landmarks in a personal setting, a new class of for PIMO concepts has been introduced starting with the PIMORE prototype in [ForgetIT, 2014e]: the so-called 'LifeSituation': it is a sub-class of an event and shall represent various life situations of users such as birth, wedding, or holidays. Thus, they are close to landmarks for users in personal settings, whereas the PIMO already contains classes (for instance, projects) as landmark candidates for organisational settings. See **PD.4.1.2** for the use of the LifeSituation in the PIMO Photo organization app.
Priority: **high**

UC.4 collect information *The user shall be able to collect & organize information for an event, i.e., during an extensive search. The resources range from web pages, emails, notes, to files.*
The Semantic Desktop infrastructure provides support for this PIM activity with the various plug-ins and applications (see, e.g., **PD.2.1.1**). Users are able to link any kind of information with events (which could also originate from a calendar).
Priority: **medium**

UC.5 tasks/checklist *The user can keep a list of tasks or a checklist with relation to the event.*
The Semantic Desktop allows to manage tasks (see **PD.5.1.1**). This ranges from simple task lists – which can be put on a stack (e.g., checklist) and synchronized with different third party applications – up to more sophisticated task management with start/end dates, reminder, notes, annotations, sharing etc.
Priority: **medium**

UC.6 schedule *The user keeps a calendar with events.*
The Semantic Desktop allows to organize events in the PIMO, but also enables user to connect to third party applications where users keep and manage their own calendar (see **PD.6.1.1**). The current pilot supports the CalDAV standard protocol for this, other protocols would be possible.
Priority: **medium**

UC.7 access information *During the event, the user accesses relevant information on mobile.*
With the PIMO5 interface, the pilot shows that accessing the PIMO on mobile devices is already possible by using a web browser on the device (see, e.g., **PD.5.2.2**). This prepares for the next pilot which will show the benefits of using memory buoyancy to keep high buoyant information on the mobile resp. pushing it there in the

background.
Priority: **medium**

UC.8 take photos *The user takes photos.*
The PIMO5 interface includes a dedicated app which allows to take photos and upload them directly into the PIMO. Photos are represented in the PIMO without further interaction of the user.

Photos taken outside of the Semantic Desktop, e.g., on the iPhone, can be easily uploaded PIMO5 allows to select a photo from the picture roll and upload it.
Priority: **medium**

UC.9 take notes *The user takes notes.*
By embedding Seed from WP4, this use case is fully covered (see **PD.7.1.1**). Especially as the pilot shows different places where (semantically-enriched) notes are available now: simple note-taking, notes on tasks, pictures, events, actually it is possible on every thing in the PIMO.
Priority: **medium**

UC.10 share photos *The user shares photos with others.*
The pilot allows to share photos with other users registered on the same PIMO server. The possibility to upload a photo to a social platform is not (yet) implemented. The sharing itself provides a further evidence for preservation value assessment. That means, by the ability to provide sharing on a restricted social platform (i.e., the group of users on a PIMO server), the user action of sharing a resource is available as an evidence for assessment in ForgetIT, e.g., for the preservation value.
Priority: **low**

UC.11 organize photos *The user will inspect and organize their photo collection.*
This use case is supported twice: in the pilot the PIMO photo organization app in Section 2.5.1 allows users to organize their photos with the help of the PIMO (see **PD.4.1.1-3**). Connecting them to PIMO concepts such as persons, events, topics, documents, etc. is possible. Furthermore, the Photo Preservation Application in Section 3 supports users in selecting photos for preservation.
Priority: **high**

UC.12 add contextual information *After the event, the user adds additional information with relation to the event.*
The Semantic Desktop infrastructure allows to add information to an event from various sources such as address books (contact details of persons), calendars (e.g., location, duration, attendees), the internet (e.g., web pages on the event), or events (e.g., arrange a photo collection and connect it to the event).
Priority: **medium**

UC.13 reminisce event *After several years, the user reminisces an event.*
This use case is realized in the PIMO Diary for contextual remembering in Section 2.5.2 (see **PD.8.1.1-8**). A condensed view of events or activities are shown.

User's are able to zoom into time intervals from decades[18] down to days. Contained resources can be accessed if they are still in the PIMO. What is still missing is to access archived material after it is removed from the PIMO. In the next versions, the links into the archive will also be stored in the condensed entry.
Priority: **high**

UC.14 reuse details from last event *The user wants to recall booking details to reuse for this year's event.*
This use case is supported by the PIMO as such material stays accessible in the PIMO an can be found and reused.

Parts of the material might be forgotten and hidden from the user's view. Still, the UI allows for showing such forgotten details (see **PD.9.2.1** and **PD.9.2.2**).
Priority: **medium**

UC.15 annotate photos *The user adds tags and descriptions to photos, organizes them together as events.*
The photo organization application in Section 2.5.1 realizes exactly this use case (see **PD.4.1.1-3**).
Priority: **high**

UC.16 evaluate photos *The user evaluates the quality and content of the photos.*
The Photo Preservation Application in Section 3 shows how image analysis components from PoF Middleware are used for image quality assessment in order to support users in the selection of photos for preservation.

The PIMO photo organization app allows to mark photos as 'favourite' and have a slider for keep and delete (see **PD.4.1.2**).
Priority: **medium**

UC.17 delete photos *The user deletes unwanted photos.*
Deletion is possible in the pilot. There are several supported steps: deletion of a local file which was synced with the PIMOCloud; in such cases the file in the cloud will be kept. It will be deleted finally, when the user deletes the thing representing the file to which the cloud file is connected.

However, we still have to address deletion if the file resides in the archive: If the preservation value was high, it might already be in the archive. In that case the user should be informed if the archive file should also be deleted. This should cover privacy issues for the user.
Priority: **high**

UC.18 view a photo collection in the archive *The user wants to see a photo collection in the archive to get an overview before accessing single photos.*
The PoF Middleware allows to access single resources and collections by an authenticated user via browser interface. This also allows to view photos in the browser.
Priority: **medium**

[18] Although the diary application supports this time frame, what can be shown with the pilot on the DFKI's PIMO is only from years to days as that PIMO covers approx. 4 years.

UC.19 retrieve a specific photo *The user wants to retrieve a specific photo.*
This use case is supported in the pilot by browse and search facilities. There are different ways of retrieving a photo, e.g., by accessing the life situation it is contained in (see **PD.4.2.1**), by drilling down in the PIMO Diary (**PD.8.1.3**), or by the facetted search (**PD.4.1.4**), which allows an associative retrieval of the photo. This is supported by the PIMO photo organization app which adds annotations to the photo with entities found in the descriptive text of the photo (using Seed), and also adds concepts[19] annotated to the life situation the photo is contained in.

If a photo is no longer available in the PIMO, it might be accessed in the archive. Search & retrieval in the archive is supported by the PoF Middleware components and can be found in deliverables such as D3.3 [ForgetIT, 2015a] and D7.2 [ForgetIT, 2014c].
Priority: **high**

UC.20 reminisce based on a photo *The user starts to reminisce based on a photo.*
This use case is supported by two different applications: PIMO Diary (see **PD.8.1.3**) and PIMO photo organization app (see, e.g., **PD.4.1.2** and **PD.4.1.5**) which allows to browse and inspect the relations of a photo in the PIMO such as the life situation a photo is contained in, the annotated and inferred concepts, etc.
Priority: **high**

UC.21 import photos *The user introduces new photos to their collection.*
This use case is supported by the pilot (via SemanticFileExplorer and PIMO5 which allows to add a photo collection to the PIMO; see **PD.4.1.1**) and the Photo Preservation Application in Section 3.
Priority: **high**

UC.22 preserve situation *The user preserves the situation for later generations.*
In the pilot, the user has several possibilities to add contextual information for a situation such as photos, textual descriptions, concepts from the PIMO as well as various external resources such as web pages or documents. As this is connected to the situation, the PoF Framework is able to identify connected information objects in case of preservation to add this to the contextualization step as well as archive this additional material.
Priority: **high**

UC.23 decide for preservation *The user explicitly decides to preserve a set of photos.*
The Photo Preservation Application presented in Section 3 realizes this use case.

In the pilot, manually triggering preservation is realized (see **PD.10.1.1**). However, PIMO5 will be extended with more possibilities to trigger such decisions, especially in the photo organization app.
Priority: **high**

UC.24 browse a person *The user browses a person's connections in the PIMO.*
Persons are important concepts in the PIMO and are supported in the pilot. They

[19] Technically, the property used for such inferred annotations is *pimo:hasInferredTopic* to differentiate btw. those annotations the user explicitly added to the photo.

have dedicated properties to represent and synchronize information from and to address books as well as further relations such as attendee, mentioned in texts, member of organization, etc. (see **PD.6.2.1**).
Priority: **high**

UC.25 view a condensed resource *The user inspects a resource which was already condensed.*
This is realized in the PIMO Diary as explained above by allowing to zoom into a diary entry (see **PD.8.1.5**).

The development of a condensed view outside of the diary app is required as soon as the condensation is stored (i.e., rebirthed) in the PIMO and can be accessed independent from the PIMO Diary.
Priority: **high**

UC.26 timeline view *The user wants to see a timeline with things (i.e., events or life situations) with relation to the person.*
This use case is supported by the PIMO Diary which allows to set time intervals and browse the condensed activities there (see **PD.8.1.1** and **PD.8.1.3**). Focusing on a specific person (or any other thing), this person can be selected from the context sidebar (see **PD.8.1.4**). A re-computation of the time interval giving the selected person a higher weight, will put more focus on those occasions where that person (or thing) is connected to.

A more basic timeline view is available for each and every thing in the PIMO. This timeline shows what happened with the thing in the PIMO, e.g., modifications of the thing or relations to other things.

Also, the RememberMe App in deliverable D3.3 [ForgetIT, 2015a] presents users a timeline view on their resources from a social media platform.
Priority: **high**

UC.27 investigate details *The user will investigate, e.g., an event in more detail, therefore will retrieve details.*
This too is supported in the PIMO Diary where more information on an (condensed) activity can be requested, drilled down, and resources inspected if still available (see **PD.8.1.5**).

For things in the PIMO, PIMO5 shows a view of a thing (person, event, life situation, document, etc.) which hides other connected things if their memory buoyancy is below a threshold. If wished, the user can request to also show forgotten details (see **PD.9.1.2**).
Priority: **medium**

UC.28 access locally unavailable material *The user accesses forgotten resources which were removed from the local computer.*
This is realized in the pilot: files which were forgotten can be retrieved by either finding them directly (see **PD.9.1.2**) or through other connected material (for instance, by accessing the task context as shown in **PD.9.1.3**). Files can be restored on the

any computer using desktop clients or accessed via the PIMO5 interface by any browser.
Priority: **high**

UC.29 restore material from archive *The user accesses material already moved to the archive.*
This is also shown in the pilot, a preserved file is retrieved from the archive. This scenario is supported by the pilot's desktop clients (SemanticFileExplorer resp. PIMOCloud, see **PD.10.1.1**). We will extend the PIMO5 interface with the ability to view and access preserved material, which is currently not yet in the UI.
Priority: **high**

UC.30 discard restored material again *The user decides that the restored material is not required locally anymore and discards all or parts of it.*
This is supported by the PIMOCloud versioning mechanism in the pilot (see also **PD.10.1.1**): a restored resource is retrieved from the archive and copied into the PIMOCloud as new version of the preserved file.[20] This general approach makes sense as the restored file might not be the same as the preserved one due to format transformations.

The restored file is then downloaded[21] and can be inspected safely without interfering with other versions. The user may decide that the restored version is the 'correct' version. Alternatively, the restored version could be 'discarded', in which case the locally restored file as well as the version in the cloud will be deleted.
Priority: **medium**

UC.31 contribute new information to archive *The user adds new information to an already preserved information object.*
On the pilot's side, it is possible to continue to use a preserved resource. Changes, additions, especially new contextual information from the PIMO are possible. It is then up to the PoF Middleware to decide how to treat this additional information of a once preserved resource. This is topic of a separate preservation workflow but not yet realized in the pilot.
Priority: **medium**

UC.32 search & access archive *After the disaster, the user wants to access preserved material in the archive.*
This is possible with the PoF Middleware. However, a dedicated UI for this is not yet available.
Priority: **medium**

UC.33 restore environment *The user wants to regain access to the PIMO on a new computer.*
The pilot realizes this use case as the Semantic Desktop installation page allows

[20] Technically, a branch branching from the preserved version is created.

[21] The restored file gets a ForgetIT-green icon with red exclamation mark, to indicate it is an unchecked restored version.

the user to quickly set up the Semantic Desktop infrastructure again.
Priority: **medium**

UC.34 restore Semantic Desktop *The user wants to restore all material on their desktop computer with connection to the semantic Desktop.*
After setting up the environment, the user decides to restore all material connected to the PIMO in the setting as it was before the disaster. The Semantic Desktop infrastructure would restore the resources on the user's computer which were not forgotten.

This is possible in the current setting. PIMO files in the cloud, as well as material available in the archive, can be restored on the computer. However, there is no convenient uniform UI available for that so far.
Priority: **medium**

UC.35 restore from archive *The user wants to get back material from the archive without requiring the Semantic Desktop.*
This is possible to some extent with the web-based access to the archive the PoF Middleware provides. A dedicated end-user interface for this is not available.
Priority: **medium**

UC.36 view resource in archive *The user wants to access a resource where no possibility to view it on the local machine exists anymore. The archive will provide this access.*
This is already possible with the PoF Middleware as part of the pilot.
Priority: **medium**

UC.37 archive migration *The user wants to migrate their archive content to a new provider.*
This is not addressed in the pilot.
Priority: **low**

UC.38 reminisce using the PIMO *A user wants to reminisce using the PIMO.*
This is realized with the PIMO Diary as part of the pilot (see **PD.8.1.1-8**).
Priority: **medium**

UC.39 death of a user *The PIMO owner has died. Apply a policy to handle the PIMO & the archive after the user's death.*
This use case is not addressed so far in the pilot.
Priority: **medium**

UC.40 purchase decision *The user looks in the internet for making a purchase decision.*
This use case is supported by the pilot by various possibilities to search, collect, and organize information for a decision such as annotating web pages, writing notes, storing images, etc.
Priority: **low**

UC.41 cleanup material

This use case is supported in the pilot by the forgetting and removing of files with low memory buoyancy. The video about the 'task context' (see **PD.9.1.3**) shows how short-term material stored on the desktop to fulfill a task gets deleted after the task has been finished a while ago and the buoyancy drops.
Priority: **low**

This completes the assessment of the use cases for the Personal Preservation Pilot I. As we have shown, most of the use cases are addressed in the pilot. Several use cases are supported by the PoF Middleware although no user interface supporting these specialized use cases are available yet.

The use cases also show that some work still needs to be done.

3 Supporting Users in Selecting Photos for Preservation

Although immediate photo sharing has become very popular, photos are also taken for capturing memorable moments and preserving them in the long run. The diffusion of high-resolution cameras allows people to take even thousands of images during relatively short events, and cheap storage indeed allows them to store all these images in some devices. Although stored, over decades storage devices break down and formats and storage media become obsolete, making parts of the photo collections inaccessible. This can be alleviated by preservation techniques, which require either continuous effort of the content owners or paying for preservation services. Due to the growing volume of content, it is expected that preservation effort and cost (which goes far beyond mere storage cost) will not allow preserving every bit of content that is created, but just the most valuable content. Thus, the problem arises, how to support the user in selecting the subset of photos to preserve.

To support users in the selection of photos for preservation, we have developed a separate application where we implement and experiment different methods that users can exploit to automatically select valuable photos from their collections for subjecting them to special preservation activities without the requirement of an existing Semantic Desktop. We also acquire feedback and insights about user expectations by giving users the possibility to revise the automatic selections done by the system. This can be used to further improve the automatic selection methods, in order to make their results closer to user expectations.

The following sections explain the application from the viewpoint of the WP9 application scenario of personal preservation. The technical details of this application are explained in more detail in WP3 deliverable D3.3 [ForgetIT, 2015a].

3.1 General Architecture

The photo preservation application is written in Java and designed as a client/server architecture. The architecture and the workflow are given in Figure 10.

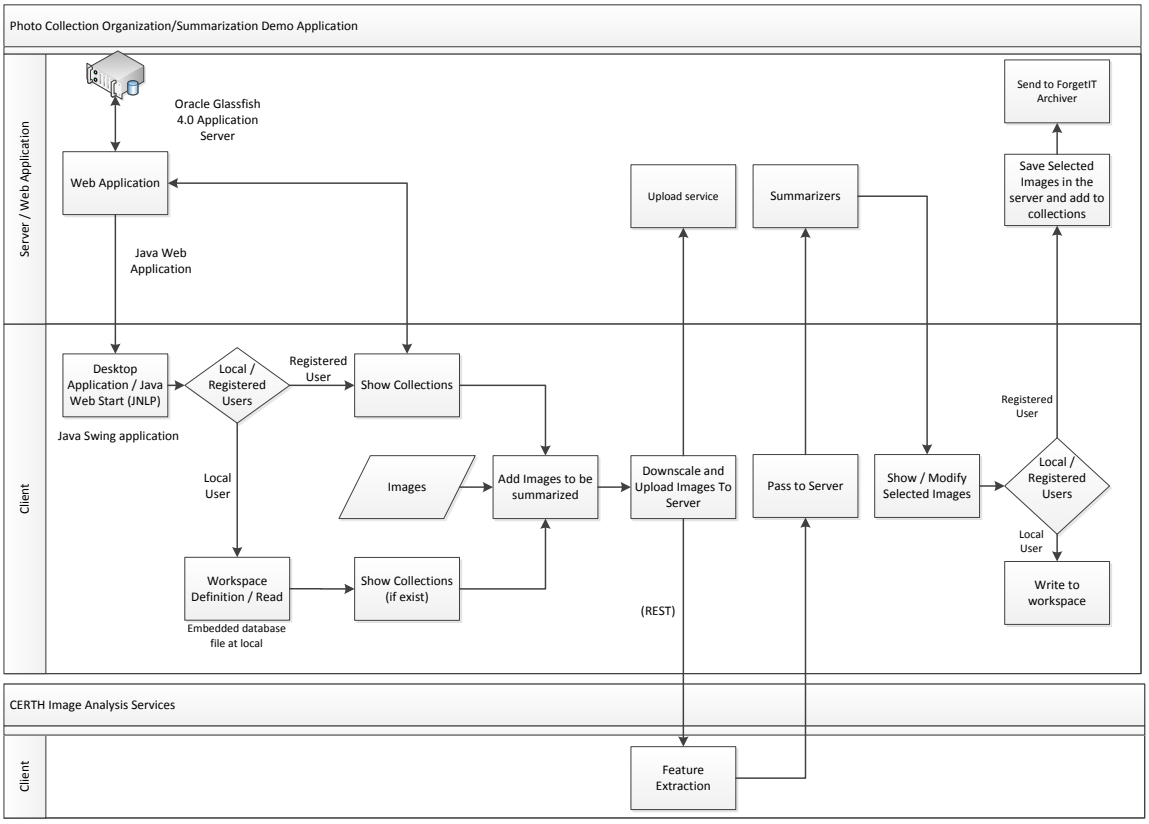

Figure 10: Photo collection summarization/organization demo application workflow

The application is composed of 3 main modules:

- Server Side
- Client
- Selection Algorithms

3.2 Client

Client is a desktop application which is written with *Java Swing*[22]. It can easily be run by clicking a simple *html* link. The main functionality of the client application is organizing all

[22] A platform independent GUI framework for Java `http://docs.oracle.com/javase/8/docs/technotes/guides/swing/`

images at the local system without waiting for the images to upload. The features are as follows:

- Login management
- Collecting images - from user folders and displaying them in folder groups including thumbnails
- Feature extraction - using CERTH services/algorithms
- Summarization/selection - of images using the extracted features
- Browsing/exploration/navigation of collection/selection, highlighting near-duplicates/ clusters
- Possibility of revising the automatic selection and feeding the modification back in the system
- Archiving to ForgetIT Archive System

Some of these features are further explained below:

3.2.1 Login Management

Each user has to register to the application in order to use it. This allows to distinguish between different user profiles and to use the corresponding selection model. The system supports 2 different login types:

Local:

- All user and image details are stored at the local system which is an h2db[23] embedded relational database.
- A user can only access his/her own collections list from the computer that he/she uses.
- No need for internet connection for organizing collections/images.

Registered/Server:

- Each user has to register and login to the server with his/her user information.
- Users can easily access their data from any computer. All data, including images are stored on the server.

[23] H2 Database Engine: a Java SQL database; `http://www.h2database.com/`

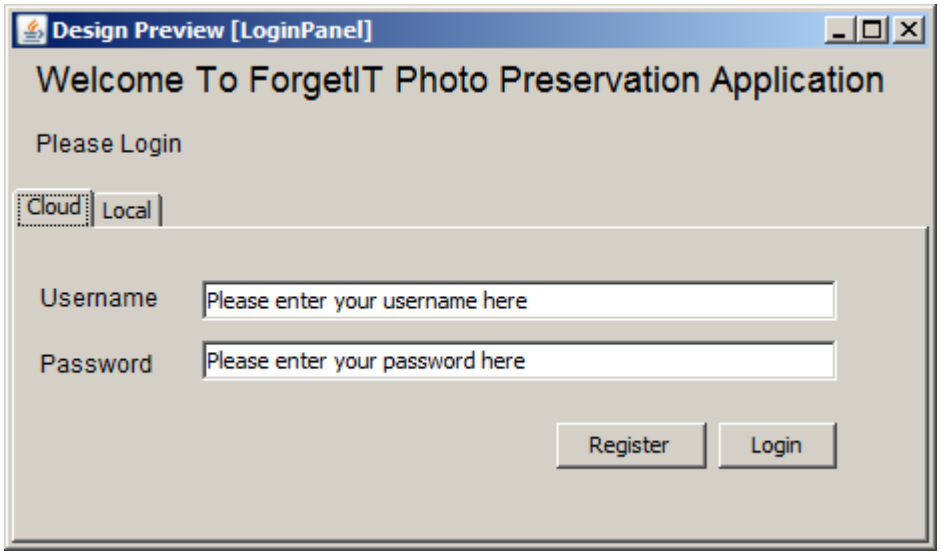

Figure 11: Photo preservation application login panel

3.2.2 Collecting Images

- Multiple files/folders can be selected and added to a collection (see Figure 12 and Figure 13).
- Files are filtered by resolution (filtering icons) and type (jpg, png, bmp, and gif are supported for now).
- Collections are viewed in a tree structure. A user can add or remove images from these collections easily.
- Thumbnails are created and saved to the local system.
- Pictures can be viewed as thumbnails.
- Pictures can be viewed in full size with double click (see Figure 14).
- Additional image selection can be performed whenever the user wants, before sending the selected images for image analysis to CERTH services.
- All the actions that are conducted are saved so that, when the app is reopened, the process doesn't start from scratch, but rather resumes where was left off.

3.2.3 Feature Extraction

- All images are uploaded to a server by the client application so that the CERTH service can access images over the internet. The images will be downscaled according to the specifications provided by CERTH.

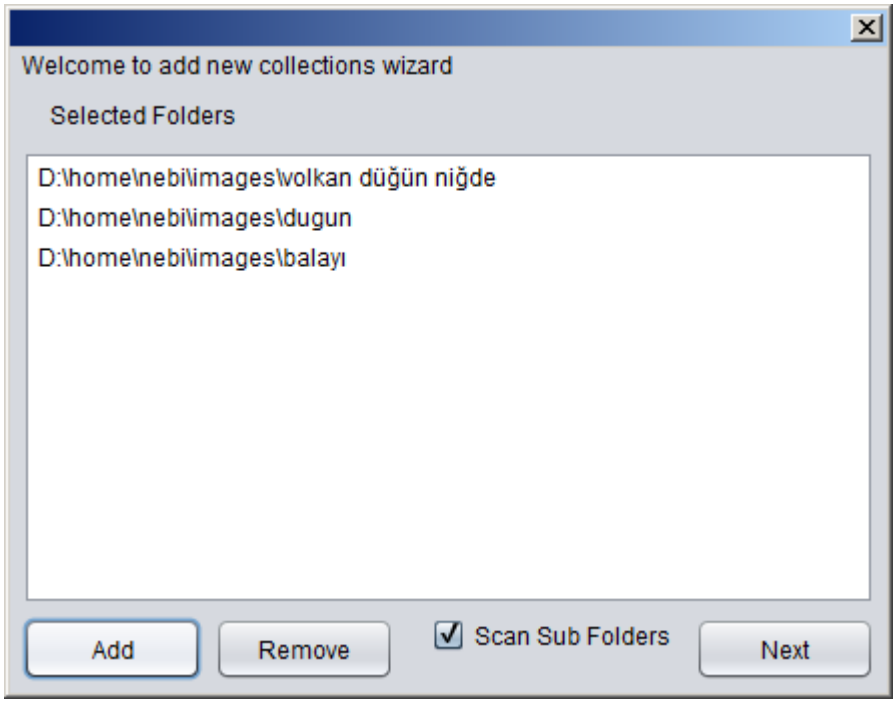

Figure 12: Photo preservation application: selection of files and folders

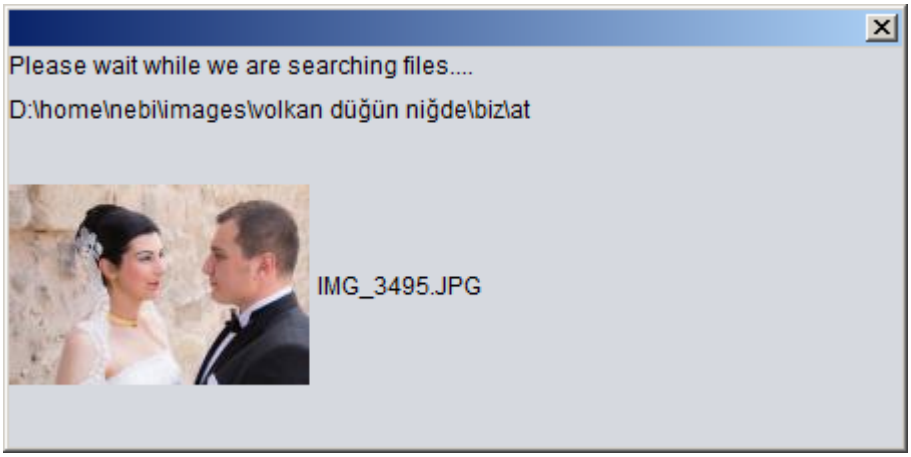

Figure 13: Photo preservation application: searching and filtering of supported image types inside selected folders

- The application will be connected to CERTH analysis services from the local system. The images URLs will be sent to CERTH and the results will be retrieved. In local mode this information is written to the local database and in remote mode all this information is stored at the server via Java Web Services[24].

[24] http://docs.oracle.com/javaee/6/tutorial/doc/gijvh.html

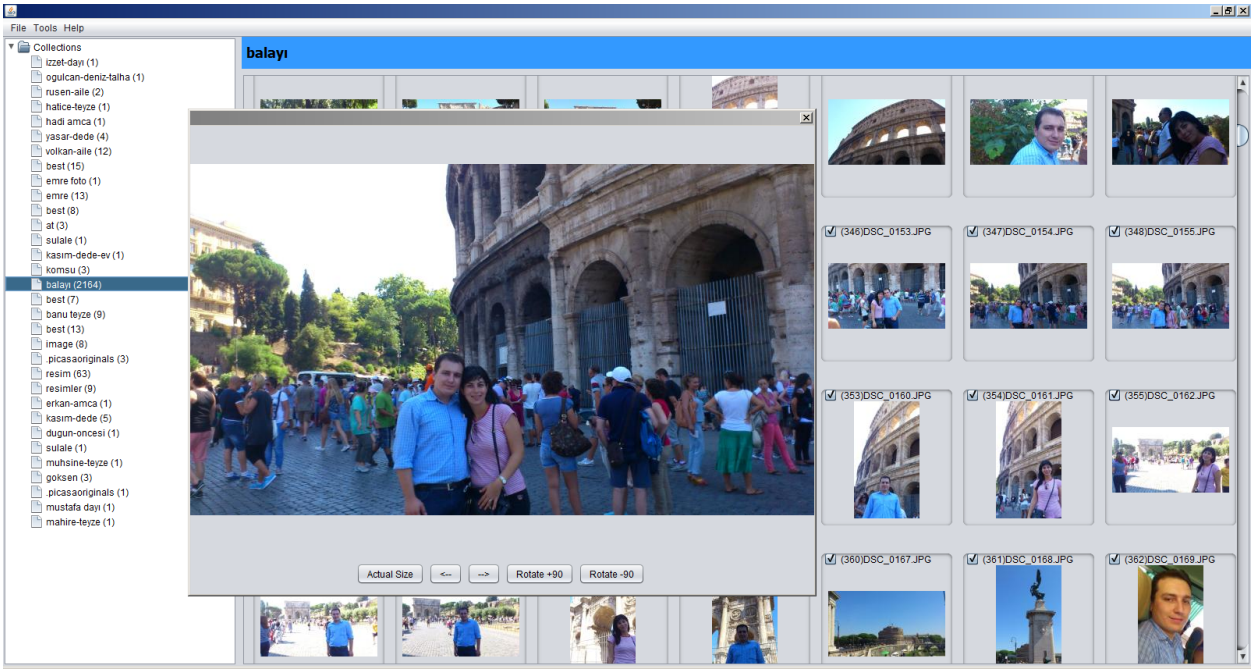

Figure 14: Photo preservation application: displaying/browsing the collection in thumbnails and full size view

3.2.4 Summarization/Selection

The user will be able to choose among the number of provided different selection/summarization algorithms. The algorithms are detailed in deliverable D4.3 [ForgetIT, 2015b] and deliverable D3.3 [ForgetIT, 2015a]. For all of these algorithms:

- The client application sends feature data of images to the server for processing using web services.
- The system returns selected and unselected image details to the client application which then displays these results.
- Additional information, e.g., clusters/near duplicates sets in the collection, will be exploited to enhance the browsing and revision of the selection. For instance, photos belonging to the same cluster might have a border of the same color.
- The user may change these selections any time and all this information will be saved to databases according to the login type. If it is a remote usage, these images are also sent to the ForgetIT Archive System.
- The user feedback, i.e., the revision of the initial automatic selection, will be used to adapt the selection model to the user preferences via the algorithms developed [ForgetIT, 2015a].

3.3 Server Side

The server side is a Java web application that works over a *Glassfish*[25] *Open Source Edition 4* application server and uses *PostgreSQL 9.4*[26] database at the back end. Main features are as follows:

- Hosts client application via *Java* web start (*JNLP*[27]) technology.
- Web services for user, collections/images, and selection algorithm management.
- Upload images service over *Java Http Servlet*.

3.3.1 Image Upload Service

The client application needs to send images to the CERTH system for feature extraction but the API only accepts images from direct web URLs. Thus, we created the *Image Upload Service* which helps to submit images for direct accessing from CERTH without opening any TCP port at the local system. It is a simple 'multipart/form-data' upload servlet that accept files sent by the client application. It writes all images in a specific folder on the server.

At the following URL, an example of an image in such a folder can be accessed:
`http://forgetit.argela.com.tr/forgetIt_media/20150114/184.JPG`

3.3.2 Summarizer API

The user will be able to choose among a number of different selection/summarization algorithms.

The *Summarizer API* is a Java-based web service for applying summarization/selection algorithms (explained in deliverables D4.3 [ForgetIT, 2015b] and D3.3 [ForgetIT, 2015a]) on the images using the extracted features. The image URLs, feature data, and algorithm type information will be sent to the selected algorithm and the results will be retrieved.

3.3.3 Archive API

The *Archive API* is a Java-based web service for archiving images by the ForgetIT Archive system for the remote login type.

[25] An application server for Java; `https://glassfish.java.net/`

[26] An SQL database; `http://www.postgresql.org/`

[27] JNLP: Java Network Launch Protocol; `http://docs.oracle.com/javase/tutorial/deployment/deploymentInDepth/jnlp.html`

4 Formative Evaluation of Personal Preservation

Formative evaluations with end users are an important part of the software development cycle. Such evaluations provide feedback on intermediate stages of a system and highlight potential problems early so that they can be integrated into the task list and appropriately prioritised during development.

For the Personal Preservation system, this data was gathered in a series of long-term case studies where four students used the PIMO system for six months (May-November 2014) in exchange for course credit. The acronym for the group, PANIC, is derived from the students' names. The project is described in Section 4.1), and intermediate results are reported in Section 4.2. Further evaluation plans are outlined in Section 4.3.

4.1 The PANIC Student Group

For the usage period, the PANIC group had access to a dedicated PIMO server where only the students were registered as users. Each of the four students had their private PIMO instance. They used the Semantic Desktop infrastructure and early prototypes of components such as the PIMO5 interface, Seed, the photo organisation app, and PIMO diary to organise their personal, student, and professional lives.

The four students were also able to share PIMO resources with each other. Although the four PANIC PIMO instances are private and only accessible to the students themselves, we also collected anonymous usage data.

The students met with the DFKI team each week and documented their experience, including success and failure stories using Powerpoint slides. These slides, together with notes from each session, are documented in a Wiki. The data also includes information about bugs, new use cases, and feature requests. The DFKI team was available to students seven days a week via instant messaging in case of problems.

As part of the web-based documentation, a video produced by the PANIC group showing a demonstration of their usage of the Semantic Desktop for a private event is included.[28].

4.2 PANIC Interviews

The intermediate interviews were conducted in late June 2014, when the students had been working with the system for eight weeks. They had mainly been using the desktop version of the Semantic Desktop and built up a substantial amount of semantically annotated data. All four had switched to Firefox and Thunderbird, the supported web browser and email client, for their day to day work. They had also installed the Semantic File Ex-

[28]See Section 11 of the web-based documentation; `https://pimo.opendfki.de/wp9-pilot/pimopanic.html`

plorer for tagging files and relational search, they had recently begun to work with Seed, the Semantic Editor introduced in D4.2, and they had been experimenting with the mobile version of the Semantic Desktop system.

4.2.1 Method

Each student participant was interviewed individually by a researcher from UEDIN (Wolters). In order to establish their background, they were asked questions about their technology use and exposure to technology. They were then asked to fill in the ForgetIT survey (c.f. D2.3 [ForgetIT, 2014b]) which explored the ways in which they obtained, managed, and stored information using the example of digital photographs.

Building on the structure of the weekly meetings between the students and the project team, the students were asked to recall a recent success story and a recent failure that involved the Semantic Desktop. For both episodes, they were asked to describe what happened, explain why they rated it a success or a failure, and fill in the NASA TLX workload questionnaire about their experience [Hart and Staveland, 1988]. This questionnaire rates the workload of six tasks, where workload measures the amount of time and effort required to complete a task. Workload is rated on a scale from 1–21. Lower values indicate lower demand, lower effort, and better performance. Here, we use workload in the human factors sense of people's experience of trying to complete tasks under pressure. Typically, this is time pressure, but the concept can be extended to cover the facets listed in Table 3.

Participants were also asked to summarise their experience with the Semantic Editor, Seed, and the Semantic Desktop itself using the System Usability Questionnaire (SUS, [Brooke, 1996]). Each item on the SUS scale is rated on a scale from 0—4, with higher values indicating better usability. The final SUS scores range from 0—100.

4.2.2 Results

The four students vary greatly in terms of their background, their attitude to preservation, and their photo taking and sharing habits. What they have in common, though, is that all of them use the file manager on their computer to file and manage their photo collections; none uses specialised software. For all four, storing their photos safely and privately is a key preservation strategy. Most of their photos come from their smartphone, and they store photos on a hard drive and on the cloud.

Student A uses the Semantic Desktop for her studies, and for her job as a sports reporter for a news paper. She also has an extensive music collection that she likes to delve into for reminiscing. She cares about preserving her photos for future generations, and worries about losing photos that are important to her. She mostly files her photos carefully, and therefore it is very frustrating when they are not filed correctly, and it can take a long time to find a specific photo.

Student C is highly IT literate. He has several gadgets, including a tablet and an eBook reader. He noted that he only began to see how useful the system could be two or three weeks ago, once it had become sufficiently stable and contained enough data. He liked the automatic tagging functions and used the system both for course work and for organising personal documents such as guitar tabs for songs.

C deletes most of the photos he takes and does not care that much about preserving them. Despite this, his preservation strategies are relatively sophisticated. He makes regular manual backups, and moves photos to new storage devices. What frustrates him most about photo management is having too many similar photos, too many unfiled photos, and the inability to search them efficiently.

Student N's computer desktop tends to be full of files, photos and documents. The ability to tag resources has made a real difference to the way in which she deals with her documents and her email, even though the web browser and email program are somewhat slow due to the demands of the Semantic Desktop.

N takes many photos, mostly with her smart phone. Old photos are archived in a folder every couple of months. This lack of a good filing system makes it hard to find photos, and to determine when a photo was taken. She worries about losing important photos, and it is very important to her to pass on photos of events that were important to her, such as holidays, to future generations. She preserves her photos carefully using manual backup, and regularly moves photos to new storage devices.

Student P mostly uses Office and Google Apps related functionality. He takes photos with his smartphone almost daily, but finds searching and managing them very frustrating, because they are not filed properly and specific photos take a long time to find. While he worries about losing important photos, which he stores on an external hard disk drive, he does not really care about passing them on to future generations.

The students' overall opinion of the Semantic Desktop is summarised in Table 1. All of them like the system and would use it frequently. The system was difficult to learn and relatively complex, but once the students had seeded it with sufficient data, and initial stability and functionality issues had been resolved, they came to appreciate the extensive contextual links that the underlying Semantic Desktop functionality of the Semantic Desktop provides.

The successful use cases all involved complex planning and search: searching annotated guitar tabs, planning a weekend trip, planning a trip to a festival, and organising course material. The problematic use cases involved problems with the system infrastructure (P: difficulties with downloading the PIMOCloud; C: uploading large files) or the editor component Seed (both A and N), which was relatively new at that time and did not reliably recognise and tag concepts in the notes.

Participants' views on Seed are summarised in Table 2. P, who used Seed as a simple editor for taking notes, was happy with the functionality, but would not use it frequently, as the key function that would have set Seed apart, automatic semantic annotation, was still being refined. This lack of reliable automatic tagging was the main reason why N gave

Table 1: System Usability Scale - PIMO, June 2014. Scores have been converted from the original scale to 0–4.

Criterion	Ratings				
	A	C	N	P	Median
Would use frequently	3	4	3	3	3.0
Simplicity	1	2	1	2	1.5
Easy to use	2	1	1	1	1.5
Self-explanatory	1	1	4	3	2.0
Well Integrated	2	3	2	2	2.5
Consistency	1	4	4	3	3.5
Quick to Learn	2	1	2	0	1.5
Straightforward to Use	2	2	3	1	2.0
Confident Using System	2	3	2	2	2.0
Learning Effort	1	2	1	4	1.5
Score (Total * 2.5)	37.5	57.5	57.5	52.5	

Seed very low usability scores.

The workload data for the success and failure episodes is summarised in Table 3. Overall, success stories tend to involve a higher mental workload, and take longer, as participants successfully use the system for a complex planning, organisation, or search task. Despite these demands, participants reached their goals with relatively little effort. The common denominator of the four failures, on the other hand, was that high perceived effort did not result in successful task completion, and this led to substantial frustration.

4.3 Further Evaluations

Due to changes in the development schedule, the evaluation plan has changed from the plans originally outlined in the revised version of D2.1 [ForgetIT, 2013a]. In addition to evaluations of individual components such as Seed, there will be two further studies.

4.3.1 Study 1: Final Debriefing of PANIC Team.

The UEDIN team will conduct semi-structured interviews with the PANIC team via Skype Instant Messenger to gain further insight into their experience with the contextualised remembering component of the Semantic Desktop as embedded in the Semantic Desktop and the PIMO5 user interface.

These interviews will be based on content analyses of the PANIC Wiki, the students' regular Powerpoint presentations where they summarised their experience of the system, and the observations of the Edinburgh ForgetIT team during the September 2014 Festival study, when participants annotated their photos using a custom version of PIMO5.

Table 2: System Usability Scale - Seed, June 2014. Scores have been converted from the original scale to 0–4.

Criterion	Ratings				
	A	C	N	P	Median
Would use frequently	3	0	1	1	2.0
Simplicity	3	4	1	4	3.5
Easy to use	0	3	1	4	2.0
Self-explanatory	4	1	1	3	2.0
Well Integrated	2	3	2	4	2.5
Consistency	1	4	0	1	1.0
Quick to Learn	1	3	0	4	2.0
Straightforward to Use	3	1	1	3	2.0
Confident Using System	3	1	0	3	2.0
Learning Effort	3	3	1	2	2.5
Score (Total * 2.5)	57.5	55.0	20.0	72.5	

Table 3: Workload Associated with Using the Semantic Desktop. Scores range from 1 to 21. Lower values indicate lower demand, lower effort, and better performance.

	Success					Failure				
	A	C	N	P	Median	A	C	N	P	Median
Mental	14	4	10	11	10.5	10	3	18	3	6.5
Physical	5	3	3	7	4.0	4	1	2	4	3.0
Temporal	15	5	15	12	13.5	5	2	11	15	8.0
Performance	3	3	2	5	3.0	18	2	20	16	17.0
Effort	15	8	5	3	6.5	12	4	15	16	14.0
Frustration	1	4	4	2	3.0	18	17	18	15	17.5

The interviews will take place in February and analysed using a combination of content analysis [Krippendorff, 2013] and Framework Analysis [Ritchie and Lewis, 2003]; results will be documented in D9.4.

4.3.2 Study 2: Final User Evaluation of Personal Preservation Pilot.

The final evaluation of the Personal Preservation Pilot will focus on a subset of the complete system, namely the photo management and preservation functionality of PIMO5. Since the current full Semantic Desktop requires users to limit themselves to specific operating systems, web browsers, and email clients (MS Windows, Firefox/Chrome, Thunderbird), recruiting more than 5 participants willing to install and use the system is not feasible within the time frame.

The PIMO5 user interface, on the other hand, provides a platform-independent window into the full ForgetIT functionality that can easily be used by participants outside of a formal usability study. This allows us to combine data on real-life use with data collected under controlled conditions where parameters that are relevant to usability can be measured.

This study, which will take place from June - October 2015, will replace the cross-sectional study outlined in the revised version of D2.1 [ForgetIT, 2013a] and focuses especially to the success indicators for evaluation as addressed in Section 5.1.

5 Summary

This deliverable described the first Personal Preservation Pilot as result of WP9 and how it contributes to ForgetIT goals. The pilot is deployed and running in the ForgetIT testbed environment and consists of components from technical work packages which are part of the PoF middleware but also part of the Semantic Desktop infrastructure.

To demonstrate the progress and success of this first pilot in WP9, the performance indicators for WP9 are assessed in the following.

5.1 Assessment of Performance Indicators

In the ForgetIT's Description of Work [ForgetIT, 2012], the following expected outcomes for WP9 were listed together with success/progress indicators:

Expected Outcomes

1. *A preservation strategy suitable for personal information management using the Semantic Desktop approach*
2. *Prototypes showing the realization of the personal preservation*
3. *Evaluated best practice of preservation-oriented knowledge modelling and personal preservation*

In the following, the progress of these outcomes are discussed along the success indicators.

A preservation strategy suitable for personal information management using the Semantic Desktop approach

- *Availability of several documented use cases for personal preservation, with appropriate presentation materials*

  Deliverable D9.2 [ForgetIT, 2014e] listed in Section 2 relevant use cases which are addressed by the pilot. Their fulfillment is assessed in Section 2.6. We have shown that most of the use cases are addressed in the pilot. Furthermore, the pilot documentation demonstrates the realization of several use cases. It will be a task until the end of the project to enhance the documentation in this respect.

- *Documented preservation strategy suitable for users applying PIM*

  The initial preservation strategy realized and documented by the Personal Preservation Pilot I is as follows:

The pilot shows how PIM is supported by the Semantic Desktop infrastructure. In doing so, various evidences are collected for the PoF to make preservation decisions. Furthermore, all resources known to the Semantic Desktop are candidates to be preserved automatically without user intervention. In addition, manually triggering preservation on resources is also possible even if the resources are not initially known to the PIMO.

Making resources known to the Semantic Desktop is simply done by either annotating a resource, defining file folders where files are automatically introduced into the PIMO, or using a specialized app from the Semantic Desktop infrastructure such as photo organization, task management, or note-taking, thus, being part of user's PIM.

The preservation is then realized by connecting the Semantic Desktop with the PoF Middleware. This is documented in the web-based documentation of the Personal Preservation Pilot I.

Thus, users using at least some of the components offered by the Semantic Desktop infrastructure for their PIM are supported by synergetic preservation of PoF. As pointed out in this deliverable, even if a user just uses one app from the infrastructure, e.g., photo organization for their photo collections, their resources are connected to the PoF.

Further adjusted preservation strategies will be introduced by the preservation policy from WP3. Such specific preservation strategies then demand an extension of the documentation for preservation strategies which will then be done in the web-based documentation.

- *Level of contribution of Semantic Desktop for selection and preservation activities*

  The pilot shows how user's are able to interact with resources in the Semantic Desktop. This allows to provide evidences for the PoF to decide for preservation. In deliverable D3.3 [ForgetIT, 2015a] the contribution for calculating the preservation value is explained. For the memory buoyancy, the calculation is based solely on interaction evidences and semantic model information from the PIMO. The pilot shows several benefits for which make use of the memory buoyancy to realize managed forgetting.

Prototypes showing the realization of the personal preservation

- *Existing prototype for a concise preserving personal desktop*

  The pilot shows the implementation of the preservation workflow by connecting to the PoF Middleware. This enables the Semantic Desktop to preserve resources in the preservation system. Restoring preserved resources is also realized within the Semantic Desktop infrastructure.

  Several applications within the pilot contribute to the ForgetIT goals of managed forgetting, synergetic preservation, and contextualized remembering as pointed out in Section 2.

- *Existing prototype for a concise preserving mobile information assistant*

  In ForgetIT, the Semantic Desktop infrastructure was extended by an HTML5 interface (the so-called 'PIMO5') which is capable of running on mobile devices. Apart from the desktop clients for preserving on the desktop, PIMO5 is the main focus of development to realize the use cases for personal preservation. With this, the second pilot which focuses on mobile devices will embrace the functionality of PIMO5 in pilot I. However, the specific requirements for pilot II are to be addressed in the upcoming months.

Evaluated best practice of preservation-oriented knowledge modelling and personal preservation

- *Ease of use of the personal preservation strategy, as shown in demonstration use cases*
- *Low additional effort for personal preservation, as shown by experimental use of the prototypes*
- *User satisfaction with the identified personal preservation strategy, as assessed by user feedback during evaluation experiments*

With the Personal Preservation Pilot I, these evaluations are now possible and are on the agenda as explained in Section 4.3.

5.2 Next steps

The next steps for WP9 are to continue work on the Personal Preservation Pilot II which also focuses on mobile devices. Furthermore, work in cooperation with WP3 will continue on preservation policies for personal preservation.

The Photo Preservation Application (Section 3) as well as the PIMO Diary for contextual remembering (Section 2.5.2) will be extended to be demo scenarios for the upcoming Year 2 review. The pilot including the PIMO Diary demo scenario will be shown at the CeBIT 2015 at the DFKI booth.

As pointed out in Section 4.3, the evaluation of the PANIC-group will be finalized and then the focus lies on the evaluation setting for Outcome 3.

References

[Brooke, 1996] Brooke, J. (1996). SUS: A Quick and Dirty Usability Scale. In Jordan, P., Thomas, B., Weerdmeester, B., and McClelland, I., editors, *Usability Evaluation in Industry*. Taylor and Francis, London, UK.

[Dengel, 2012] Dengel, A., editor (2012). *Semantische Technologien*. Spektrum, Heidelberg. in German.

[ForgetIT, 2012] ForgetIT (2012). Annex I: Description of Work. Grant Agreement 600826, ForgetIT consortium.

[ForgetIT, 2013a] ForgetIT (2013a). D2.1: Evaluation plan. Deliverable D2.1, ForgetIT consortium.

[ForgetIT, 2013b] ForgetIT (2013b). D9.1: Application use cases & requirements document. Deliverable D9.1, ForgetIT consortium.

[ForgetIT, 2014a] ForgetIT (2014a). D2.2: Foundations of forgetting and remembering - preliminary report. Deliverable D2.2, ForgetIT consortium.

[ForgetIT, 2014b] ForgetIT (2014b). D2.3: Foundations of forgetting and remembering intermediary report. Deliverable D2.3, ForgetIT consortium.

[ForgetIT, 2014c] ForgetIT (2014c). D7.2: Computational storage services first release. Deliverable D7.2, ForgetIT consortium.

[ForgetIT, 2014d] ForgetIT (2014d). D8.3: The preserve-or-forget framework - first release. Deliverable D8.3, ForgetIT consortium.

[ForgetIT, 2014e] ForgetIT (2014e). D9.2: Use cases & mock-up development. Deliverable D9.2, ForgetIT consortium.

[ForgetIT, 2015a] ForgetIT (2015a). D3.3: Strategies and components for managed forgetting second release. Deliverable D3.3, ForgetIT consortium.

[ForgetIT, 2015b] ForgetIT (2015b). D4.3: Information analysis, consolidation and concentration techniques, and evaluation - second release. Deliverable D4.3, ForgetIT consortium.

[ForgetIT, 2015c] ForgetIT (2015c). D6.3:: Contextualisation tools - second release: Updates to the context modelling framework and modules. Deliverable D6.3, ForgetIT consortium.

[ForgetIT, 2015d] ForgetIT (2015d). D8.2: The preserve-or-forget reference model initial model. Deliverable D8.2, ForgetIT consortium.

[Hart and Staveland, 1988] Hart, S. G. and Staveland, L. E. (1988). Development of nasa-tlx (task load index): Results of empirical and theoretical research. In Hancock, P. A. and Meshkati, N., editors, *Human Mental Workload*. North Holland Press.

[Krippendorff, 2013] Krippendorff, K. (2013). *Content Analysis*. SAGE, 3rd edition.

[Maus et al., 2013] Maus, H., Schwarz, S., and Dengel, A. (2013). Weaving personal knowledge spaces into office applications. In Fathi, M., editor, *Integration of Practice-Oriented Knowledge Technology: Trends and Prospectives*, pages 71–82. Springer.

[Ritchie and Lewis, 2003] Ritchie, J. and Lewis, J., editors (2003). *Qualitative Research Practice: A Guide for Social Science Students and Researchers*, volume 134. SAGE.

[Sauermann et al., 2005] Sauermann, L., Bernardi, A., and Dengel, A. (2005). Overview and Outlook on the Semantic Desktop. In Decker, S., Park, J., Quan, D., and Sauermann, L., editors, *Proceedings of the First Semantic Desktop Workshop at the ISWC Conference 2005*.

[Sauermann et al., 2007] Sauermann, L., van Elst, L., and Dengel, A. (2007). PIMO – A Framework for Representing Personal Information Models. In Pellegrini, T. and Schaffert, S., editors, *I-SEMANTICS Conference 5-7 September 2007, Graz, Austria*, J.UCS, pages 270–277. Know-Center, Austria.