Project no. 269977

APARSEN

Alliance for Permanent Access to the Records of Science Network

Instrument: Network of Excellence

Thematic Priority: ICT 6-4.1 – Digital Libraries and Digital Preservation

D22.2 SET OF ADDED VALUE SERVICES AND EVALUATION OF USER SATISFACTION (INTERIM: MONTH 16)

Document identifier:	**APARSEN**-REP-D22_2-01-1_7
Due Date:	2012-04-30
Submission Date:	2012-12-31
Work package:	WP22
Partners:	FRD, UNITN, UESSEX, CERN, STM, DANS, DNB, BL, GLOBIT, FORTH, BL, MRI, APA
WP Lead Partner:	FRD
Document status	Released
URN	urn:nbn:de:101-20140516119

Abstract: This report aims at describing the interoperability services, which are at the core of the Interoperability Framework described in D22.1 and a set of dependant services enabled by it.

Delivery Type	Report
Author(s)	Emanuele Bellini (FRD), Chiara Cirinnà (FRD), Maurizi Lunghi (FRD), Barbara Bazzanella (UNITN), Paolo Bouquet (UNITN), Salvatore Mele (CERN), Hervé L'Hours (UESSEX), Yannis Tzitzikas (FORTH), Marjan Grootveld (DANS), Rene van Horik (DANS), Gerald Jaeschke (GLOBIT), Eefke Smit (STM), Sabine Schrimpf (DNB), Kirnn Kaur (BL), Heikki Helin (CSC), David Giaretta (APA)
Approval	Simon Lambert
Summary	This report aims at describing the interoperability services, which are at the core of the Interoperability Framework described in D22.1 and a set of dependant services enabled by it.
Keyword List	Persistent identifiers
Availability	☒ Public

Document Status Sheet

Issue	Date	Comment	Author
1.4	26 April 2012		
1.5	14 June 2012		
1.6	1 Oct 2012	After internal review by Orit Edelstein, Stefan Pröll, John Lindström	
1.7	20 Oct 2012	Minor edits prior to submission to EC	Simon Lambert

Project information

Project acronym:	**APARSEN**
Project full title:	**Alliance for Permanent Access to the Records of Science Network**
Proposal/Contract no.:	**269977**

Project coordinator: Simon Lambert/David Giaretta

Address:	STFC, Rutherford Appleton Laboratory Chilton, Didcot, Oxon OX11 0QX, UK
Phone:	+44 1235 446235
Fax:	+44 1235 446362
Mobile:	+44 (0) 7770326304
E-mail:	simon.lambert@stfc.ac.uk / david.giaretta@stfc.ac.uk

CONTENT

EXECUTIVE SUMMARY

This intermediate version of deliverable D22.2 defines a set of services built on the Persistent Identifier (PI) Interoperability Framework that was defined in D22.1. This step will be followed by a prototype implementation of the interoperability framework and some of its services, which is the final stage of this work package and will be presented in the final version of this deliverable.

The services designed are the translation of Use Cases presented in D22.1. According to them we distinguished three kinds of services following the formalization of the Service-Oriented Modeling Framework [2]: atomic, composite and cluster services. While atomic services are defined as services that are an integral part of the framework, composite and cluster services can be considered as dependant services enabled by the core infrastructure of the IF. Since a multitude of different dependant services can be envisioned based on specific interoperability needs, we provide here just few relevant examples of these services, which have been extracted mainly from the scenarios and uses cases described in the D22.1.

The second step of the design activity of the framework will be dedicated to the selection of the services (basic, composite and atomic), which will be implemented in the final prototype.

1 INTRODUCTION

The work of WP22 on identifiers and citability proceeds in three stages:

- definition of an Interoperability Framework for Persistent Identifiers
- definition of a set of services built on the interoperability framework
- prototype implementation of some services and their evaluation with users

The first stage was presented in deliverable D22.1 'Persistent identifiers interoperability framework'. The present deliverable, an intermediate version of D22.2, deals with the second stage. A prototype implementation will be carried out and reported in the final version of this deliverable, in accordance with its title 'Set of added value services and evaluation of user satisfaction'.

The prototype will implement the technical infrastructure to manage the standardized relationships between the identified entities (i.e. digital objects, authors, institutions), their PIs, corresponding resolution services and related information (metadata), creating a common interoperability layer where meaningful information from independent systems can be integrated, re-used and exploited to enable value added services.

The second stage towards the development of the technical infrastructure for the interoperability framework is the definition of the interoperability services enabled by the framework. These services can be distinguished as **atomic**, **composite** and **cluster services** according to the SOMF asset definitions. Atomic services are defined as services that are integral part of the framework and provide the basic atomic tokens for building more advanced interoperability services on top of it. In other words, these services offer the elementary functionalities, which can be adopted and integrated in different higher-level services. PI-resource resolution, PI-metadata resolution and PI-alternative resolution services are examples of atomic level services. From the combination of two or more of these basic level services, a sort of intermediate service, called composite can be provided. An intermediate service is designed to perform a unique interoperability task. For example, combining the PI-alternative resolution service and the PI-metadata resolution service, it is possible to implement a service, which returns all the metadata about a given resource distributed across different systems. Finally, cluster services are defined as services that exploit basic and intermediate services to respond to more specific interoperability needs providing a unique point of access to execute multiple related tasks. These advanced services are built on top of the framework. An example of such a service could be an entity life cycle system for digital preservation, which monitors the level of accessibility of a resource based on parameters like the number of alternative IDs for the resource and orients digital preservation planning.

Figure 1 represents the relations between the IF and these services. It is worth noticing that the user (including both human users and machines) can interact with the services at any level. In fact, they can use atomic, composite or cluster services according to their needs. This approach allows recombination and mashup of the services and promotes the development of other new services.

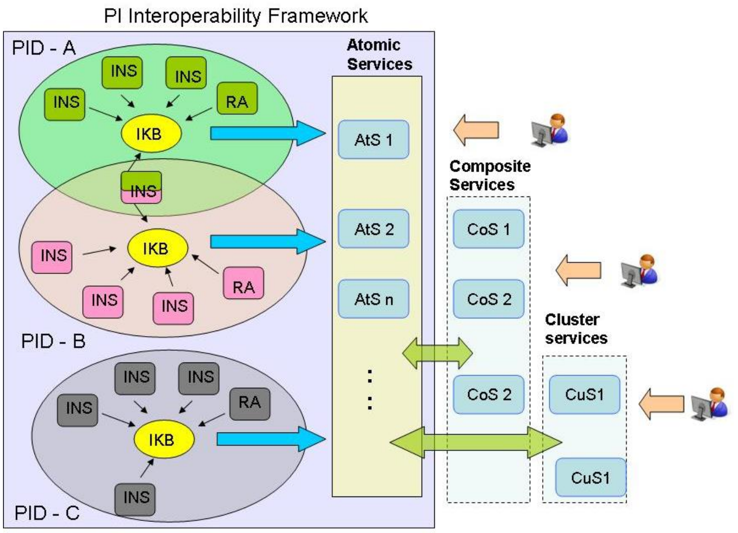

Figure 1: Relationships among IF and Dependant Services

The aim of this report is to describe a set of relevant interoperability services which can be implemented as part of the technical infrastructure of the IF prototype. The definition of the services is based on an accurate analysis of the scenarios and the use cases described in the deliverable D22.1 (Section 3.5 and 3.6). A mapping between services and the uses cases of the D22.1 is provided. While the description of the atomic services, which represent the core of the framework tries to be exhaustive, that of intermediate and especially advanced services is thought to provide some significant examples among the multitude of customized and tailored services, which can be implemented on top of the framework.

1.1 WORK PACKAGE 22 IDENTIFIERS AND CITABILITY: OBJECTIVES AND TASK RELATIONS

To address the issues underlying the interoperability challenge described above, Work Package 22 has three main objectives:

a) to provide an overview of the current PI systems and criteria for evaluation;
b) to design a reference model to describe an interoperability framework;
c) to define community-driven added value services.

WP22 consists of three tasks. The relations between these tasks are shown in Figure 2.

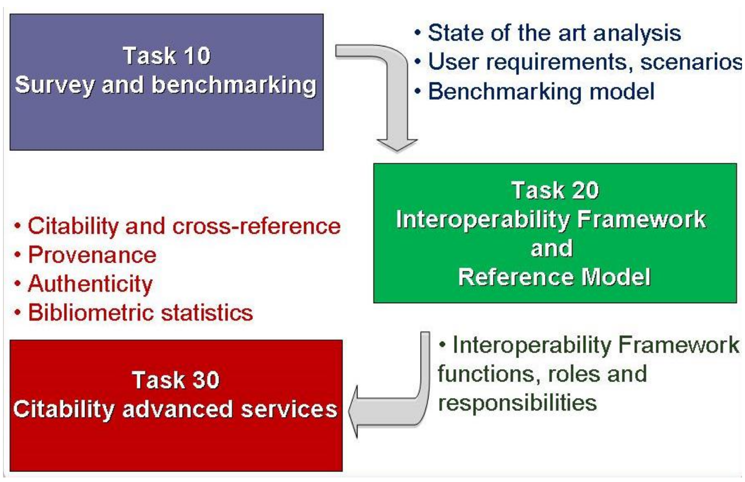

Figure 2: WP22 Tasks relationship

Task 10 has 3 main objectives: 1) to define the complete state of art of the available PI systems; 2) to provide an analysis of user requirements, use cases and scenarios, to outline a set of criteria in terms of organisational framework, object management, infrastructure and security; 3) to provide a benchmarking model, with criteria for evaluation, to support user communities and institutions in the identification of the appropriate PI system.

Task 20 is focused on modelling an IF for PI systems which addresses functions, roles and responsibilities to allow interoperability among these systems.

Task 30 aims at designing some advanced services for resources identified by different PI systems, such as services for citability, cross-referencing, quality assessment, citation metrics and evaluating the user satisfaction about these services.

1.2 SCOPE OF THE DELIVERABLE

Essentially, the present deliverable addresses the first part of the Task 30 and covers the description of services which are integral part of the framework and an exemplificative set of dependant services built up on the IF. This work translates the results reported in the D22.1 (where a number of scenarios and use cases were provided) into related web services descriptions.

In particular a number of scenarios were selected as candidate for the proof of concept implementation, according to the user requirements survey results [3].

1.3 RELATIONS WITH STREAM 2 WORK PACKAGES

Interoperability services and their relationships with other work packages of the project

WP 24 Authenticity and Provenance: a digital resource lifecycle service as described in D24.1 is a good example of a service which can be built on top of the IF. The process of tracking a digital resource throughout its lifecycle can be managed through its PIs, which can be embedded into the logging files. Through the IF and the alternative PIs service, the system could trace all the phases and events which the resource may undergo (also in different systems, including changes in technical system and custody) without losing its authenticity and provenance evidence. The availability of mapping among PIs may also allow the implementation of services for exchanging and integrating provenance information. For example, associating the ontology/schema mapping of provenance models with the PIs mapping provided by the IF could improve provenance interoperability.

WP25 Interoperability and Intelligibility: the definition of the IF and its dependant services is strongly related to the activities of the WP 25 that focuses on investigating and developing techniques to support syntactic and semantic interoperability of data between organizations and disciplines. The IF guaranteeing the persistent access to digital resources across systems, functions, metadata schema, semantic and linguistic barriers, should provide a first layer of interoperability on which more sophisticated models regarding the interoperability between metadata, standard protocols and ontologies can be implemented. For example, semantic interoperability of metadata - which express the relationships that someone claims to exists between entities – depends on the unique identification of all these entities, since otherwise expressing relationships between them and agree on the meaning of these relationships is of scarce utility. Moreover, the identification of the authority, which makes the claim is crucial as well. On this perspective, the IF (and its dependant services) can be at the core of the WP 25 activities related to semantic interoperability since it provides the key tokens to the management of identification for implementing effective solutions to support semantic interoperability.

1.4 METHODOLOGY

This work translates the results reported in DE22.1 where a number of scenarios and use cases were provided, into related web services descriptions. According to the DOW, we selected a subset of services to be implemented as a proof of concept of the IF, based on survey results. In fact the users have clearly indicated which services they consider more crucial. As reported in D22.1 the results of the Persistent Identifiers survey indicate that citability (76%), global resolution service (61%) and a PI resolution service to the resource and digital object certification (both 55% ca.) are considered the most interesting services among the proposed options. These options were presented in order to understand the user expectations of a class of underlying services. For instance, the citability option includes services directly related to the citation capability, but also metrics services (selected by 30% of users). Taking in to account the survey results, we clustered the scenarios and the use cases containing the suggested services into three macro categories: 1) **Citability and Metrics** 2) **Global Resolution services** and 3) **Digital object certification**. Only a subset of these services has been included in this report. A mapping between the selected services and the corresponding use cases is provided when available.

2 IF-BASED SERVICES DESIGN

2.1 SERVICE-ORIENTEDMODELLING FRAMEWORK APPROACH

Service Oriented Architecture model (SOA) is a relatively new approach for defining and implementing services over the network. SOA is a design for linking business and computational resources (principally organizations, applications and data) on demand to achieve the desired results for service consumers (which can be end users or other services). The following definition, given in [1], is now common for Web services and the related architectures.

“A service-oriented architecture is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity. Some means of connecting services to each other is needed”.

There are different approaches for service modelling, e.g. SOMA (Service-Oriented Modeling and Architecture) by IBM and SOMF (Service-Oriented Modeling Framework) [2].

In this document we adopt the SOMF approach. The SOMF offers a simple modeling language that enables traceability of business, architecture, and design decisions.

The SOMF specifies a Logical Design Model Language to support in particular the conceptualization and design phases. Within the framework are defined six software development disciplines that offer corresponding models whose language notion help practitioners to design a service ecosystem. These disciplines are: Conceptualization Model, Discovery and Analysis Model, Business Integration Model, Logical Design Model, Software Architecture Model and Cloud Computing Toolbox.

In this document we adopt the Logical Design Model Language as an approach for representing service relationships. This approach is based on message exchange routes between service consumers and service providers. Namely, the message paths established to carry information and execute transactions drive the association between services and their corresponding consumers. Therefore, we think about service relationship as a model for delivering and routing data by using messages between a service and a related consumer. Conceptual associations or any other business affiliations between a service and a consumer are important; however here the focus is merely on the technical requirements that drive the design of message routing and delivery.

SOMF identifies a taxonomy of software components that provide services. In practice a software component can be any organizational software asset, such as an object, module, component library, application, business process, database, store procedure or trigger, Enterprise Service Bus (ESB), Web service, and more. **Fehler! Verweisquelle konnte nicht gefunden werden.** illustrates these service models, which are defined as follows:

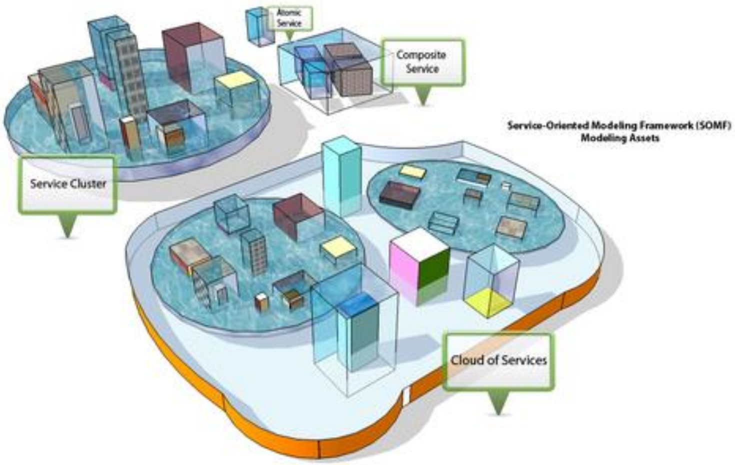

Figure 3: Atomic, composit, cluster and cloud of services (source:Wikipedia)

Atomic service: a software component that is indivisible because of its high granularity and performs a smaller number of technical or commercial functionality. An atomic service is identified when the analysis of the activities or the technological functionalities does not justify development of smaller components.

Composite service: a composite service structure aggregates small and fine-grained services[1]. The term "composite" implies that an entity is comprised of one or more internal software components. In other words, the aggregated child services are contained within a parent service structure. A composite service can aggregate other atomic or composite services. A Composite services is designed to address a unique task.

Cluster service: This is a collection of related services that are distributed and collected because of their mutual business or technological characteristics. A cluster service's affiliate services combine their offerings to solve a business problem. A Cluster service represents an access point to address multiple tasks. A cluster service can aggregate atomic services as well as composite.

Example: An ATM service is an example of cluster service because it provides a unique access point to multiple services (tasks) like money check out, tax payments, mobile recharge, and so forth.

On the contrary, the tax payment service enabled by the cluster service is an example of composite service corresponding to a unique task. In turn this composite service is composed by several atomic services like security code check, bank account authorization, and so forth.

Cloud of Services: a collection of services that are delivered by a cloud computing implementation. These services can be classified as Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and more [1].

This last type of services will be not considered in this document because their implementation complexity cannot be addressed in the context of the present development.

Fehler! Verweisquelle konnte nicht gefunden werden. illustrates a consumer and a generalized notion of any software entity that may not only provide services, but also calls other services for data and information.

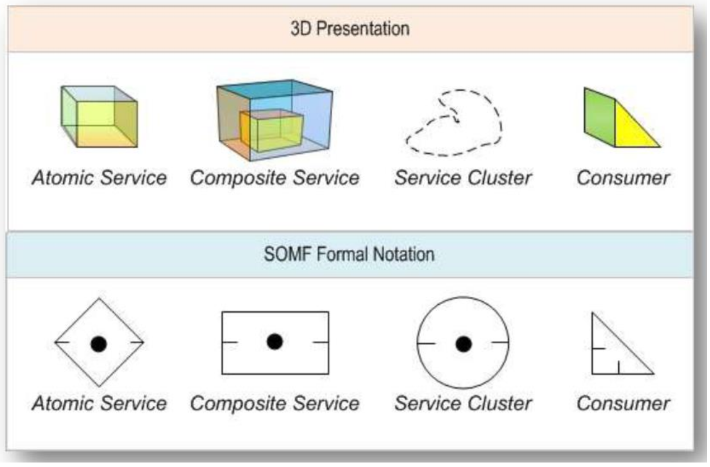

Figure 4: SOMF Logical Design Relationship Modeling Assets

2.1.1 SOMF logical design relationship notation

Fehler! Verweisquelle konnte nicht gefunden werden. illustrates the notation used to denote logical relationships between services and corresponding consumers. These symbols are typically used during the service-oriented logical design phase of a project to identify the message routes that must be established between services and their related consumers.

[1] http://en.wikipedia.org/wiki/Service-oriented_modeling

- *Apparent Bidirectional*: this connector depicts a two-way message routing akin to the request/response message pattern. Typically, the consumer invokes a request and the service responds. The term "apparent" signifies the direct link between a service and a consumer, without any interception of a third party software entity.
- *Apparent Unidirectional*: a one-way solicitation or an acknowledgment message routing, during which either the consumer or a service originate a message. A response is not required by the receiving entity. Again, the term "apparent" pertains to a message route that is not intercepted by any other software entity.
- *Implied Bidirectional*: a request/response two-way message routing between a consumer and a service. The term "implied" identifies a message route scenario that is intercepted by a third party broker to deliver the message to its destination.
- *Implied Unidirectional*: a one-way solicitation or acknowledgment message routing. Again, the term "implied" signifies interception by a third party broker to deliver messages.

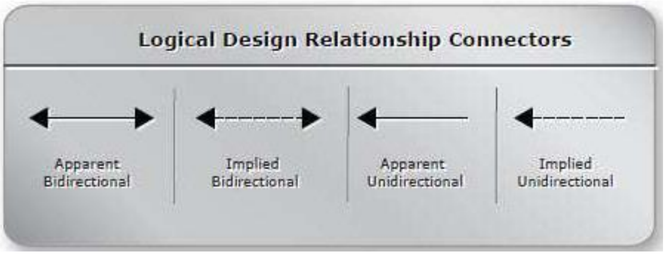

Figure 5: SOMF Logical Design Relationship Notation

2.2 ATOMIC SERVICES

An atomic service is a software component that should be indivisible. In other words, it would be impractical to decompose an atomic service because of its typical fine-grained structure. The term "fine-grained" pertains to a service that offers limited functionality and possesses very few capabilities. As stated above, these "primitive" services are at the heart of the interoperability framework because they provide the pillars on which more complex services can be developed. These services are fundamental to exploit the explicit relationships between entities (e.g. digital objects, authors, institutions, PIs, resolution services and other related contents) which are managed within the framework. The services should be robust and unchanging across time.

Fehler! Verweisquelle konnte nicht gefunden werden. below is applicable to all atomic services in this section

a) Service/consumer: Atomic, Consumer

b) Connector: Apparent bidirectional

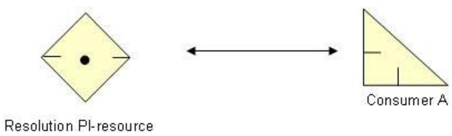

Figure 6: Logical design relationship diagram of PI-R resolution service

2.2.1 Resolution services

The first basic functional requirement for the IF is resolution. There are at least three basic resolution services that can be envisioned based on this functionality:

1. PI-RESOURCE RESOLUTION SERVICE

Input: a PI

Output: a Resource

Description: This service associates a PI to the corresponding resource. Any kind of PI among those provided by trusted Persistent Identifier Domains (PIDs) can be submitted to the service. In the IF the resolution process can be implemented as a multi-phased process. The first step of the resolution process consists of finding the right local resolution service for a given PI (e.g. locating the DOI resolver given that a DOI has been submitted to the service). This means that the framework should be implemented with a register of resolvers mapping the PI type to the corresponding resolver. The resource resolution service finds the resource through the local resolver and returns it to the user.

Use case of reference:

Use case 1: identifier resolution.

2. PI-METADATA RESOLUTION SERVICE

Input: PI

Output: Metadata/Representation information

Description: When the resolution process does not return the resource corresponding to a given PI, but returns the metadata associated with it, a second resolution service is involved which is called the PI-metadata resolution service. This service associates a PI to the metadata of the identified resource. It is important to noticing that the metadata are not necessarily stored in the IF, but can be accessed through the resolution service via a PI in the same way described for the PI-resource resolution service.

Use case of reference:

Use case 1: identifier resolution.

Use case 9: metadata management

3. PI-ALTERNATIVE PIs

Input: PI

Output: PIs list

Description: While the resolution services described above associate a PI to a resource or metadata about the resource, the PI-alternative PIs service associates a PI to alternative PIs for the same resource. In the description of the IF in D22.1, we stressed that the same resource can be identified by more than one PI (e.g. a document can be identified by a DOI and by a URN). The functionality of registering alternative identifiers for the same resource is a fundamental requisite for the IF because it guarantees multiple ways to access the resource and related information, making the resolution process really persistent. Therefore this service can help to fix the broken link problem. Moreover, having an access point to alternative PIs is a prerequisite for building intermediate services which can exploit the alternative identifiers to extract new (i.e. implied) relationships between relevant entities and related information and consequently integrate information across systems boundaries.

Use case of reference

Use case 2: identifier resolution, broken link resolution service.

2.2.2 Resolution counter

Input: PI
Output: Number of times the PI has been resolved
Description: This service returns the number of times that a resource has been accessed through the IF by resolving a given PI. Digital preservation services can benefit from the resolution counter service, implementing tracking metrics which monitor the level of usage of the resource across time.
Use case of reference:

2.2.3 GET Entity PI LIST

Input: Type of entity (author, digital object, institution)
Output: List of PI of all instances of the entity
Description: This service returns the list of PIs that refer to an entity type based on the ontology categorization. For instance the 'Get Digital Object PI List' request will return all the PIs of the digital objects present in the knowledge base.
Use case of reference:

2.3 COMPOSITE SERVICES

Composite services can be seen as intermediate services that are built from the combination of basic services and can be envisioned as constituting an intermediate layer on top of the core of the IF which enables the advanced cluster services. The figure 7 represents such a configuration with two atomic services that are exploited in a composite service. In this report we describe some relevant examples of composite services that have been suggested by the analysis of the scenarios and use cases collected from different stakeholders within the project.

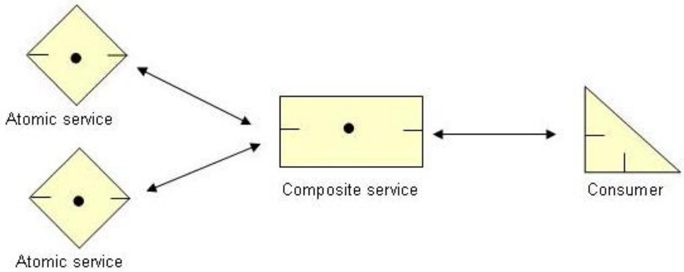

Figure 7: Logical design relationship diagram for Composite services

2.3.1 Entity relationships service

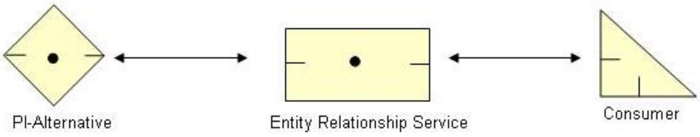

Figure 8: Logical design relationship diagram for Entity-Relationship service

Input: PI, depth of search parameter.

Output: List of related entities

Description: Since the knowledge that can be stored in the IF contains the relevant relationships between the registered entities (i.e. digital objects, authors and institutions), a composite service of the framework could use all the PIs of a given entity (i.e. target entity) to navigate the IF knowledge base and find all the entities related to the target entity. We consider this service a composite service because it exploits the alternative PI service, which is a basic service, and uses the alternative PIs to extract the relevant relationships addressing a unique interoperability task. For example, this service can extract the following relationship:

- from author ID (i.e. ORCID) to research output (data, paper etc.): you enter an author identifier and receive his/her research output, such as papers, presentations, reports, data etc;

- from data to paper discovery: you enter a DOI of a dataset and find associated papers that have either described or reused the particular dataset

- paper to data resolution: you enter an identifier for an eprint/publication or example and find associated data or other linked materials.

Use case of reference:

Use case 4: finding related content

Use case 10: extracting relations between digital objects, authors and institutions

2.3.2 Metadata integration service

a) Service/consumer: PI-Alternative resolution, PI-Metadata resolution, Metadata Integration, Consumer

b) Connector: Apparent bidirectional

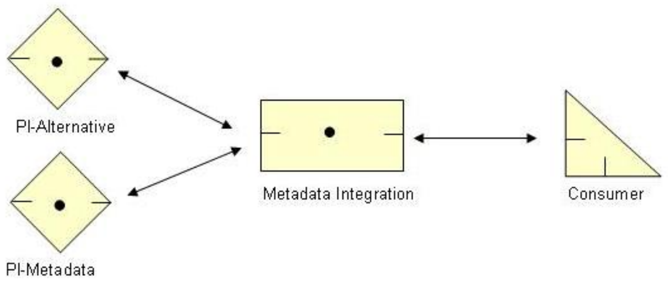

Figure 9: Logical design relationship diagram for Metadata Integration Service

Input: PI

Output: Metadata mashup

Description: This service is similar to the previous one with the only difference that the alternative PIs are used to extract from the knowledge base of the framework all the metadata referring to the same entity. Moreover the metadata resolution service can be used to access to metadata stored by content providers. Once the metadata have been extracted, they can be integrated and organized into a unique and standardized object following a certain agreed metadata schema.

Use case of reference:

2.3.3 Selection and appraisal preservation service

a) Service/consumer: PI-Alternative resolution, resolution, Resolution counter, Selection and appraisal preservation service, Consumer

b) Connector: Apparent bidirectional

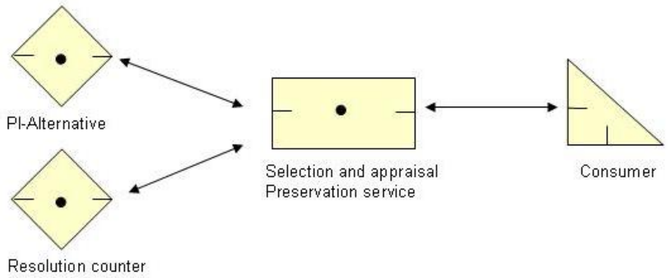

Figure 10: Logical design relationship diagram for Selection and Appraisal preservation service

Input: PI (digital object)

Output: Risk Index

Description: An advanced service for supporting the selection and preservation of Web resources in institutional repositories (e.g. national libraries). This service should implement automatic procedures to help humans (e.g. archivists) in the complex problem of selective harvesting and preservation decisions about digital resources. The service could exploit data provided by basic and intermediate services to suggest selection and appraisal decisions. For example the number of alternative PIs can be used as an index of the relevance of the resource, but also of its level of accessibility, which could influence strategies for preservation planning.

The number of accesses to the resource through the IF is another variable that can be considered by the selection and appraisal service to implement tracking metrics which can be used in turn to estimate the value of the resource.

Use case of reference:

Use case 5: discovery of versions of the "same" digital object

Use case 8: deduplication

2.3.4 Metadata search engine

a) Service/consumer: Get Entity PI List, PI-Metadata resolution, Metadata Search Engine, Consumer

b) Connector: Apparent bidirectional

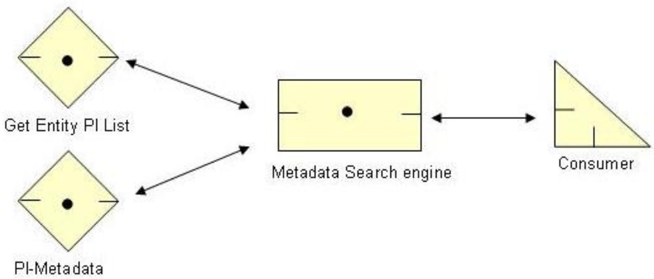

Figure 11: Logical design relationship diagram for Metadata Search Engine

Input: Metadata values (Entity qualified that means that the entity type parameter has been specified)

Output: List of metadata records matched

Description: This service takes as input metadata values and an entity type parameter and performs a search on its own database populated by harvesting the entire list of PIs related to a certain entity type (e.g. digital object) and the atomic PI-Metadata resolution service is called on each PI. The results (metadata) are stored and indexed in the service database. Then, a full text search on the metadata stored is performed to extract all the records containing the metadata provided in input.

Use case of reference:

2.4 CLUSTER SERVICES

A cluster of services is a collection of software entities, such as atomic and/or composite services that collaboratively provide a solution to perform a series of related tasks. Grouping services to offer a joint solution typically provides synergies. Additionally, each participating service in a cluster may belong to another application or composite formation, so software reuse is promoted with services providing business or technological value to implementations that may be unrelated to their original intended use.

2.4.1 Scientific career assessment

a) Service/consumer: PI-Alternative resolution, PI-Metadata resolution, PI-LinkMap resolution Metadata Integration, Scientific Career Assessment, Consumer

b) Connector: Apparent bidirectional

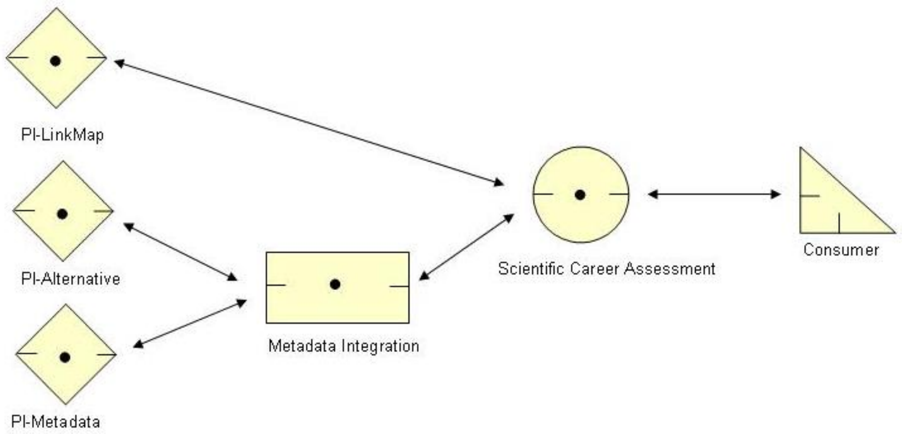

Figure 12: Logical design relationship diagram for Scientific Career Assessment

Input: Author PI + filters

Output: list of publications (metadata + PI) filtered

Description: The scientific career assessment service is built on top of the entity relationship composite service. This service can interrogate the intermediate entity relationship service via the PI of a given author, and retrieve all the related entities of the author. At this point the service allows filtering the returned results by entity type (e.g. only the digital objects) and other features such as provenance (e.g. only official publications), date, co-authors and so on. The final output is the list of publications of the author, which responds to the specified criteria.

Use case of reference:

Use case 7: career assessment

Use case 4: finding related content

2.4.2 PI retrieval service

a) Service/consumer: PI-Alternative resolution, PI-Metadata resolution, Get Entity PI List, Metadata Search Engine, PI Retrieval Service, Consumer

b) Connector: Apparent bidirectional

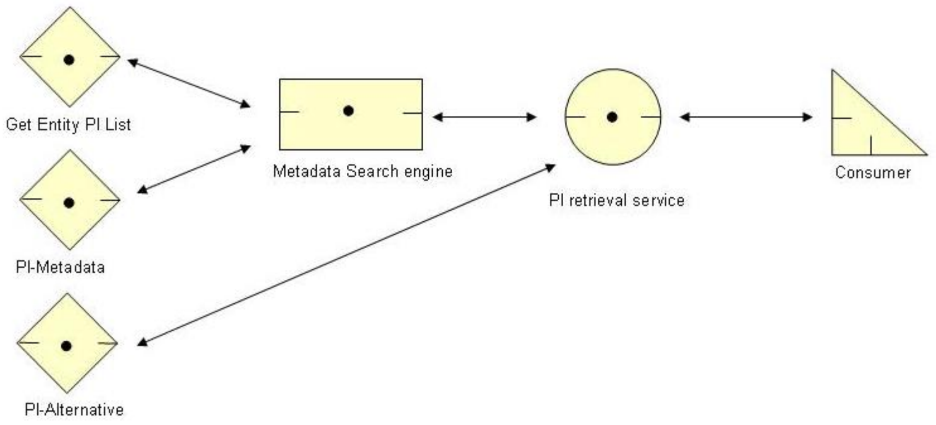

Figure 13: Logical design relationship diagram for PI Retrieval service

Input: metadata value

Output: PI/ PIs

Description: This service allows the use of metadata to retrieve the PI (or the alternative PIs) of a given entity. For example, the title of a paper and its author can be used to obtain its DOI (or alternative registered PIs).

Use case of reference:

Use case 9: metadata management

2.4.3 Archical curation service

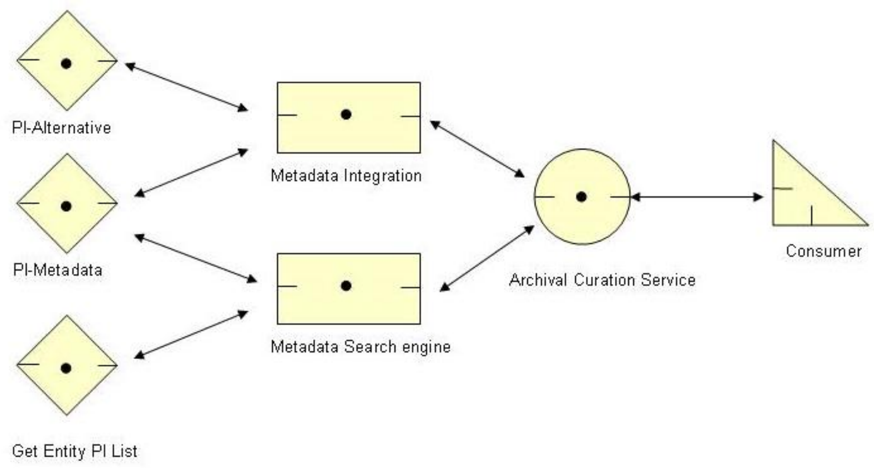

Figure 14: Logical design relationship diagram for Archival Curation Service

Input: Author PI and/or metadata based resource discovery
Output: alternate digital objects assigned to same author or closely related via metadata (Link Map, Alternative PI)

Description: Upon ingest to a digital repository the author PI and relevant metadata are used for resource discovery via the Framework. A number of key ingest tasks can be supported including:

- Ensuring the submitted material is not a duplicate/partial duplicate of already ingested materials (the Archive then rejects the submission or integrates the material into an existing digital object).
- Identify when the submitted material is a result of secondary analysis from materials already stored in the Archive (the Archive then builds a relationship between the digital objects)
- Identify when the submitted materials have already been assigned a PI (Optionally the Archive chooses not to assign a second PI).
- Identify when the submitted material (or related important contextual material) is already richly described elsewhere (the Archive incorporates the metadata found via the Framework into local holdings).
- Identify when submitted materials constitute an update to existing holdings (the Archive integrates the new submission and versions metadata accordingly.

Use case of reference
Use case 8: deduplication
Use case 9: metadata management

3 REFERENCES

[1] SOA definition Barry &Associates, Inc.
http://www.service-architecture.com/webservices/articles/serviceoriented_architecture_soa_definition.html

[2] Bell, Michael (2008). "Introduction to Service-Oriented Modeling". Service-Oriented Modeling: Service Analysis, Design, and Architecture. Wiley & Sons. ISBN 978-0-470-14111-3.

[3] APARSEN D22.1 Persistent Identifier Interoperability Framework