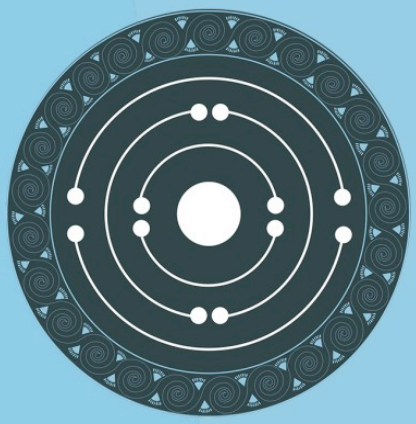

PARTHENOS

Pooling Activities, Resources and Tools for Heritage E-research Networking, Optimization and Synergies

Design of the Joint Resource Registry

AUTHORS: Nicola Aloia,
Leonardo Candela,
Franca Debole,
Luca Frosini
Matteo Lorenzini
Pasquale Pagano

DATE 12 April 2017

PARTHENOS is a Horizon 2020 project funded by the European Commission. The views and opinions expressed in this publication are the sole responsibility of the author and do not necessarily reflect the views of the European Commission.

HORIZON 2020 - INFRADEV-4-2014/2015:

Grant Agreement No. 654119

PARTHENOS

Pooling Activities, Resources and Tools for Heritage E-research Networking, Optimization and Synergies

NAME OF THE DELIVERABLE

Report on the Design of the Joint Resource Registry

Deliverable Number	D5.2
Dissemination Level	Public
Delivery date	12 April 2017
Status	Final
Authors	Nicola Aloia Leonardo Candela Franca Debole Luca Frosini Matteo Lorenzini Pasquale Pagano
Contributors	Francesca Frontini Athanasios N. Karasimos Kostas Stefanidis Jennifer Edmond Sara Di Giorgio Jan Kocon Reto Speck Adeline Joffres

Project Acronym	PARTHENOS
Project Full title	Pooling Activities, Resources and Tools for Heritage E-research Networking, Optimization and Synergies
Grant Agreement nr.	654119

Deliverable/Document Information

Deliverable nr./title	D5.2 Report on the Design of the Joint Resource Registry
Document title	Report on the Design of the Joint Resource Registry
Author(s)	Nicola Aloia, Leonardo Candela, Franca Debole, Luca Frosini, Matteo Lorenzini, Pasquale Pagano
Dissemination level/distribution	Public

Document History

Version/date	Changes/approval	Author/Approved by
V 0.1 11.03.16	Initial version	Matteo, Lorenzini Nicola Aloia
V 0.2 22.03.16	Franca & Francesca paragraph added	Nicola Aloia
V 0.3 24.03.16	Integrated with Matteo contributions	Nicola Aloia
V 0.3 06.04.16	Athanasios and Kostas contribution added and whole Document revision.	Nicola Aloia
V 1.2 12.04.16	New 4.3.4 paragraph contributed by Athanasios and whole Document revision.	Nicola Aloia
V 1.3 28.04.16	New 4.1.2 paragraph contributed by Matteo. New 4.8 contributed by Sara. New 4.10 contribution by Reto Speck. Whole Document revision and enhancement.	Nicola Aloia
V 1.4 23.05.16	Improvement of paragraph 5 and 6. Huma-Num contribution added.	Nicola Aloia Franca Debole

	General review.	
V 1.4.1 23.06.16	Integrated CLARIN new contribution from Matteo, Minor changes from Francesca and from Athanasios	Nicola Aloia
V 1.5 30.07.16	New Chapter 6	Luca Frosini Leonardo Candela Pasquale Pagano
V 1.5.1 21.09.16	Chapter 6 revision to consider modification performed in Task 5.1	Luca Frosini Pasquale Pagano
V 1.6.0 12.10.17	New Chapter 7	Luca Frosini Pasquale Pagano
V 1.6.1 22.03.17	Chapter 6 revision to consider modification performed in Task 5.1	Luca Frosini Pasquale Pagano
V 1.6.2 30.03.17	Modification to Chapter 7 to update port-type operations	Luca Frosini
Final 10.04.17	Final Candidate Release	Pasquale Pagano
Final 11.04.17	Final Quality Review	Sheena Bassett

Table of content

Index of Figures

Index of Tables

1. Executive Summary

The activity of design of the Joint Resource Registry is propaedeutic to the building phase of a comprehensive inventory of resources and is the result of two different activities:

- A survey of resources (datasets, collections, infrastructures, services) available in the archaeological context;
- The definition of the main entities for the PARTHENOS data model taking into account the ontology defined in T5.1.

In this document, we present a description of the existing registries in the humanities area, derived from the analysis conducted in the first activity. The goal of the activity was to identify the main descriptive conceptual entities and features of existing resources that would be useful for the definition of the PARTHENOS Registry Data Model.

This document is organized as follows: Section 2 describes the best-known standards used in the definition of registry models; Section 3 describes the survey carried out of existing registries in the thematic area of the PARTHENOS project; Section 4 contains a detailed description of the surveyed registries; Section 5 summarizes the main entities and functionalities of the analysed registries; Section 6 presents the PARTHENOS Joint Resource Registry Data Model; and finally, Section 7 presents the architecture of the Joint Resource Registry service that implements the model.

2. Common registry standards

In this section, we will describe two of the best-known standards used for the description of some of the existing registries on the Web.

2.1. The W3C DCAT standard

DCAT is an RDF vocabulary, published by the Government Linked Data Working Group at W3C as a recommendation[1] to describe datasets and catalogues on the Web in order to enable their discoverability and consumption by services. The DCAT model "*is well-suited to representing government data catalogs such as Data.gov and data.gov.uk*" and has been proposed as a tool for publishing datasets as Open Data. Currently, various datasets have been published in according to the DCAT specifications[2] and various European projects officially recommend its adoption[3]. The main classes that constitute the DCAT model are:

- dcat:Catalog represents the catalogue (a curated collection of metadata about datasets).
- dcat:Dataset represents a dataset in a catalogue (a collection of data, published or curated by a single agent, and available for access or download in one or more formats).
- dcat:Distribution represents an accessible form of a dataset (each dataset might be available in different forms, these forms might represent different formats of the dataset or different endpoints).

DCAT re-uses classes and properties from several other vocabularies, such as *foaf:Agent*, skos:*Concept*. Table 1 contains the complete list of namespaces used by DCAT.

Prefix	Namespace
dcat	http://www.w3.org/ns/dcat#
dct	http://purl.org/dc/terms/

[1] http://www.w3.org/TR/vocab-dcat/
[2] http://www.w3.org/2011/gld/wiki/DCAT_Implementations
[3] https://joinup.ec.europa.eu/asset/dcat_application_profile/

dctype	http://purl.org/dc/dcmitype/
foaf	http://xmlns.com/foaf/0.1/
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
skos	http://www.w3.org/2004/02/skos/core#
xsd	http://www.w3.org/2001/XMLSchema#

Table 1. DCAT namespaces

The following figure gives an overview of the main interactions among the basic classes of DCAT.

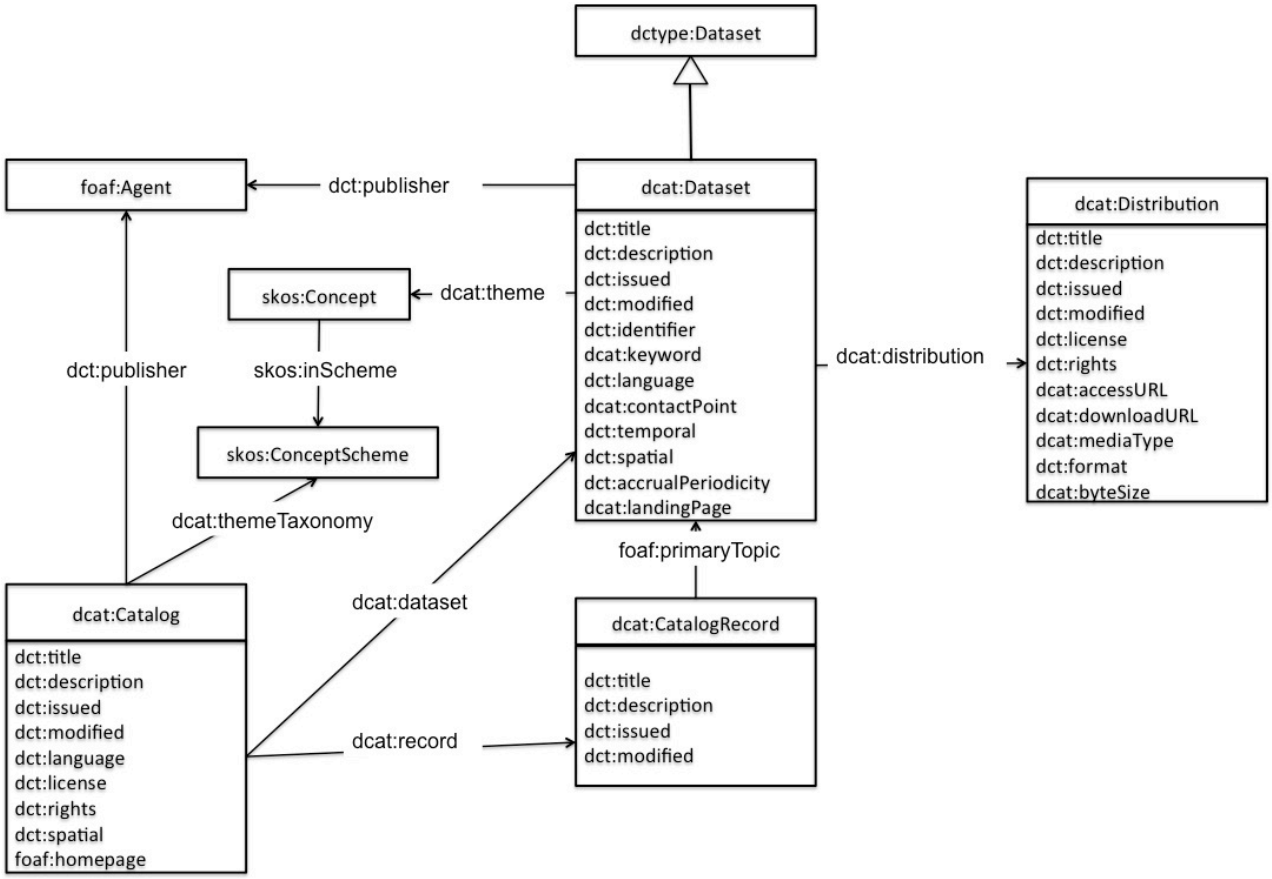

Figure 1. The DCAT data model

2.2. ISOcat – a Data Category Registry

The ISOcat Data Category Registry (DCR[4]) has been a joint project of both ISO TC 37 and the European CLARIN infrastructure. ISO 12620 is a standard from ISO/TC 37, which defines a Data Category Registry, a registry for registering linguistic terms used in various

[4] http://www.isocat.org.

fields of translation, computational linguistics and natural language processing and defining mappings between both different terms and the same terms used in different systems. The goal of the registry is that new systems can reuse existing terminology, or at least be easily mapped to existing terminology.

Data categories as used by TC 37 are based on data elements as defined by the ISO 11179 standards (ISO 11179-1. Information technology -- Metadata registries (MDR) -- Part 1: Framework, 2004). In the framework of this family of standards, a data category is a concept with additional specification of its representation, i.e., does the data category have a value domain (complex data category) or not (simple data category), and if so what kind of domain (open, closed or constrained) and of which data type.

The ISOcat DCR is the result of an on-going effort by TC 37 to standardize data categories (Kemps-Snijders, 2009). As a successor of the paper list of data categories in ISO 12620:1999 (ISO 12620. Data Categories, 1999) and all the shortcomings of that, i.e., hard to extend with new data categories needed by the community, it was decided to create a registry. The data model of, and procedures around, the registry are described in ISO 12620:2009 (ISO 12620. Specification of data categories and management of a Data Category Registry for language resources, 2009).

Although ISOcat is a completely new implementation of this standard, it is also the successor of SYNTAX, a pilot DCR implementation. The data categories stored in SYNTAX and the registered user base were transformed and imported into ISOcat.

On the one hand, this underlines the intention and obligation of TC 37 to keep the past data category specification work available. On the other hand, the quality of a lot of these inherited specifications is also considered problematic. Uptake of ISOcat by new users is hampered when they inspect such sub-optimal entries, either inherited or recent additions.

The ISO 12620:2009 standard and its implementation in ISOcat have been largely driven by the requirements of ISO TC 37 and far less so by CLARIN, as design and development of the CLARIN infrastructure was just starting up. However, the CLARIN infrastructure has been growing the last few years and its own requirements have now become clearer.

The standard was first released as ISO 12620:1999 (ISO 12620. Data Categories, 1999) which was rendered obsolete by ISO 12620:2009 (ISO 12620. Specification of data categories and management of a Data Category Registry for language resources, 2009). The first edition was an English-only, the second one was bilingual (English-French).

The standard is relatively low-level but it is used by other standards such as Lexical Markup Framework (ISO 24613. Language resource management — Lexical markup framework, 2008).

The maintenance of ISOcat was handed to the Max Planck Institute (MPI) which also serves as its ISO Registration Authority since the end of 2008. Although the MPI was successful in improving considerably on the old SYNTAX implementation and integrated ISOcat in the CLARIN CMD framework, complaints about the current user interface persist. In line with these insights and MPI's renewed focus on institute research, the MPI stopped being the RA and hosting provider in December 2014. After reviewing potential replacement systems, ISO TC37 selected TermWeb, from Interverbum Technology, due to its support of the required data model.

For users from the European CLARIN research infrastructure, the Meertens Institute develops and hosts a new registry for CLARIN relevant concepts based on the corresponding ISOcat data categories, such as those used for the Component MetaData Infrastructure (CMDI). One of the principles behind ISOcat is that it is an open registry, so it is very easy to get a login and to get the rights to enter new data categories (private or public).

3. Survey of existing registries

The survey of existing registries has been carried out in two phases. The first phase (*general census*) aims to collect basic information from the existing registries developed throughout other projects[5] in the humanities and cultural heritage domain. The second phase (*detailed census*) consists of a deeper analysis of the registries, examined in the first phase, chosen for the definition and extraction of the main entities and concepts for PARTHENOS's registry.

3.1. General census

This phase is functional to the selection of the registries concerning the humanities domain that we can take as model for the further development of the PARTHENOS registry. In agreement with the other partners, we defined the basic information necessary for the census of existing registries. Table 2 shows the list of basic information.

Registry	Name of the registry or name of the project
Registry URL	URL of the registry
Description	Short description of the registry reported
API	Specify if the registry provide an API service
Metadata Schema	Specify the metadata schema of the registry
Protocol	Specify the protocol used for metadata access and harvesting
Licence	Specify the licence of the registry
Contributed by	Indicate who provided the information for given registry (name, mail address, Institution)

Table 2. Required information for general census

All the partners involved in T5.4 have been invited to provide information via a Google Sheet according to the table above.

3.2. Detailed census

The detailed census starts from the survey carried out during the first phase and consist of a deeper analysis of the registries chosen for extracting the main entities and concepts

[5] ARIADNE, CLARIN, DARIAH, CENDARI, etc.

suitable for PARTHENOS's registry. The analysis of the existing registries considers both functional and non-functional features.

The partners have been invited to supply detailed information using the following template.

Registry Name	The registry name	Note
Web site	A Web page that gives access to the registry or provides information on it	
Contact person	Name, email	
Thematic Area	One or more areas related to the PARTHENOS Project	
Description	Textual description	
License	A licensing scheme, e.g. CC-BY	
Dynamicity	Average frequency of change	
Metadata Schema	The schema used, possibly with a link to the definition or the specification.	
Access Policy	One of: open Access, restricted, copyrighted, embargo	
Contact in PARTHENOS	Person in the PARTHENOS project to be contacted	
Download URL	A URL to a downloadable distribution of the registry, if any	
Issued	Date of formal issuance (e.g., publication) of the resource.	
Modified	Most recent date, on which the resource was modified. If not provided the issued date will be used.	

Table 3. Detailed registries census

Entity	Y/N	Number of entries	Description
Dataset			
Institution			
Ontology			
Person			
Service			
Software			

Table 4. Categories of objects described in the registry

Operation	Y/N	Note/Description	Instructions
Online GUI for editing			“Y” if the registry allows to enter, modify or delete information via an on-line GUI, “N” otherwise
Import			“Y” if registry offers a functionality to get data into the registry, “N” otherwise. You may additionally specify the type of import, such as through OAI-PMH providers.
Browsing			“Y” if registry offers a functionality to browse the registry, “N” otherwise.
Search			“Y” if registry offers a functionality to search the registry via a query, “N” otherwise.
Export			“Y” if registry offers a functionality to get data out from the registry, “N” otherwise. You may additionally specify the format(s) of the exported data
API			“Y” if the registry has an API, “N” otherwise. You may additionally specify the type of the API, e.g. Rest/SOAP/… + output formats and give a pointer to the definition or the specification

Table 5. Functionalities offered by the registry.

Many of the T5.4 partners provided files containing detailed descriptions of known registries. The collected files were subsequently processed to produce a uniform description and format. All the descriptions are accessible at the following link: https://goo.gl/Xqmnri.

This activity of collecting registry descriptions is a continuous activity that will continue to be performed for the duration of the project. An updated version of the collected information can always be accessed at the *Original* and *Aligned* folders accessible through the aforementioned link to the D4Science infrastructure workspace.

4. Detailed description of the surveyed registries

This analysis aims to describe the main component and the architecture of the surveyed registries. The description will be useful to understand the different solutions adopted by other projects about the development of the registries and how the different entities interact among them. The description of the surveyed registries is based on the information acquired by the census tables[6] integrated with the available documentation provided by the various projects.

4.1. CLARIN

CLARIN is a research infrastructure initiative that aims at providing a single domain of Language Resources (LR) and Language Technology (LT) to researchers from the Social Sciences and Humanities (SSH) disciplines. CLARIN is on the ESFRI roadmap and was awarded an ERIC in 2012. The basis of the CLARIN infrastructure was already created in the so-called preparatory phase, with funding from the EC in the CLARIN EU project, that ran from 2008 to 2011. In this project phase, the CLARIN infrastructure organisational and functional architectures were designed and largely built on a foundation of CLARIN centres that commit to the responsibility of delivering their usual infrastructure services in a CLARIN compatible manner (CLARIN B-type centres and services) and on a subset of centres, who commit to delivering general infrastructure services beneficial for the whole CLARIN community (CLARIN A-type centres and services). CLARIN centres are usually National (Research) Institutes or Departments of Universities involved with linguistic research that also have a role in providing language type data for the research community. The CLARIN infrastructure relies on a set of integrated dedicated registries (Table 6):

[6] https://docs.google.com/spreadsheets/d/1l5FRgifNIf1sP2_8IC4BB28H9v8i3eNTC-j8FwcbuS8/edit#gid=0

Registry / Catalogue	Stores and exposes information about:	
CLARIN Centres registry	CLARIN Centres, their capabilities (especially available endpoints), as well as responsible persons	http://clarin.eu/clarin-eric-datatables/centres https://centres.clarin.eu/
Component Registry	CMD profiles (/metadata schemas)	https://www.clarin.eu/componentregistry
CLARIN Concept registry	Concepts used for semantic annotation of metadata schemas and their definitions	https://www.clarin.eu/conceptregistry/
Virtual Language Observatory	Metadata catalogue. Faceted browser over the metadata collected from CLARIN partners	https://vlo.clarin.eu/

Table 6. CLARIN registries

Figure 2 shows the relationships among the various registries.

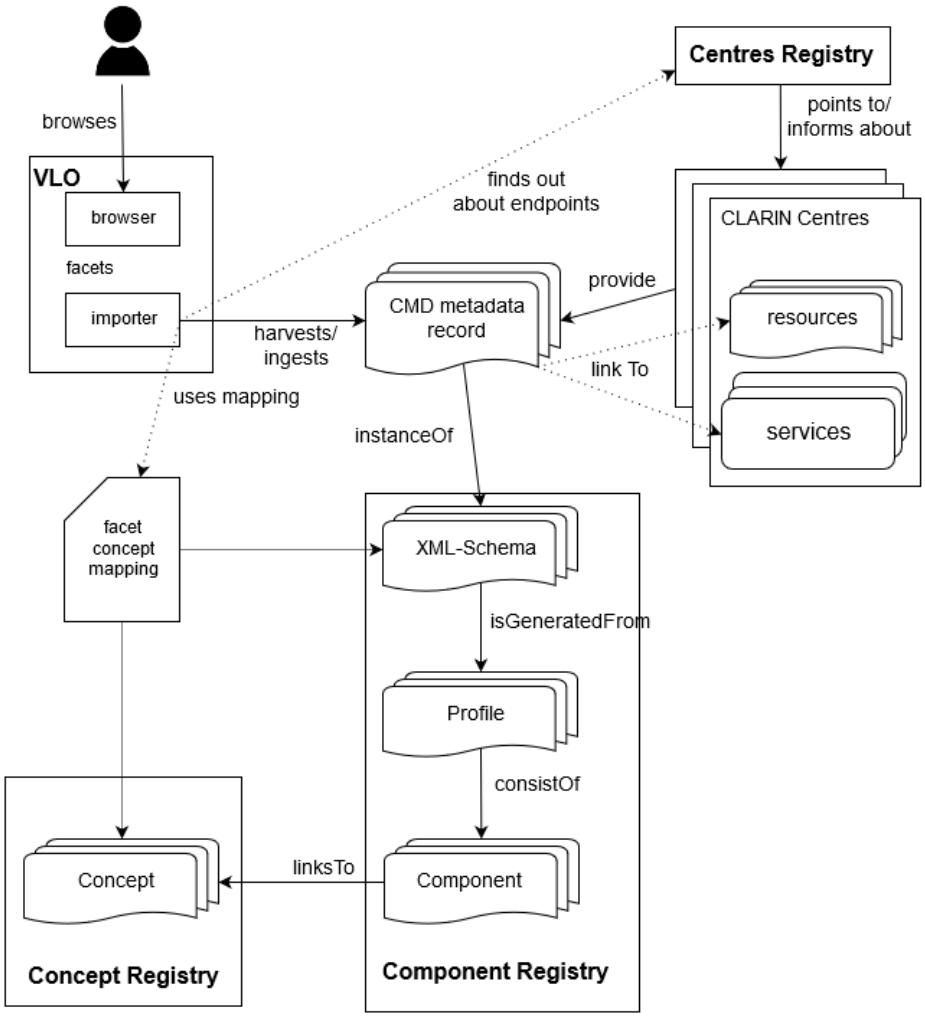

Figure 2. CLARIN registries relationships

4.1.1. CLARIN Centres Registry

The CLARIN Centres Registry is the primary starting/entry point into the CLARIN universe. It is the authoritative source of approved CLARIN Centres including information about the contact and the different available endpoints. Other parts of the infrastructure (especially the harvester) consult this registry to find out about (discover) existing centres and the endpoints they feature. A list of CLARIN centres is provided in Appendix A.

4.1.2. CLARIN Concept Registry

The CLARIN Concept Registry (CCR) is an OpenSKOS instance, which implements the W3C SKOS recommendation and data model. The CLARIN Concept Registry (CCR) forms the basis of the semantic interoperability layer of CLARIN, especially in the context of metadata, i.e., the Component MetaData Infrastructure (CMDI[7]). It does so by offering a collection of concepts, identifiable by their persistent identifiers, relevant for the domain of language resources. This registry contains the relevant concepts (based on the corresponding ISOcat data categories), such as those used by CMDI. The Concept Registry can be accessed by anyone using a read only facetted browser or via the search facilities of the CMDI Component Registry (see 4.1.3). Adding new concepts or changing existing ones can only be done by the national CCR coordinators, whose mission is to improve the quality of the concepts used within CLARIN.

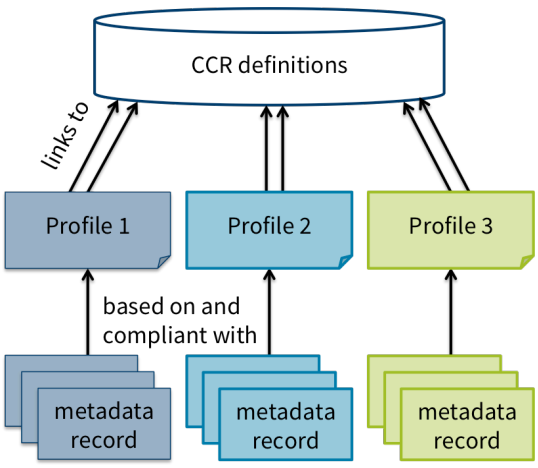

Figure 3. CLARIN Concept Registry

[7] http://www.clarin.eu/content/component-metadata.

The OpenSKOS software provides an OAI-PMH server to export structured metadata, an API (**Error! Reference source not found.**) to search, create, update, delete, and share thesauri and/or vocabularies (concepts in our case), and also provides a web-based editor (Figure 4) for most of these tasks.

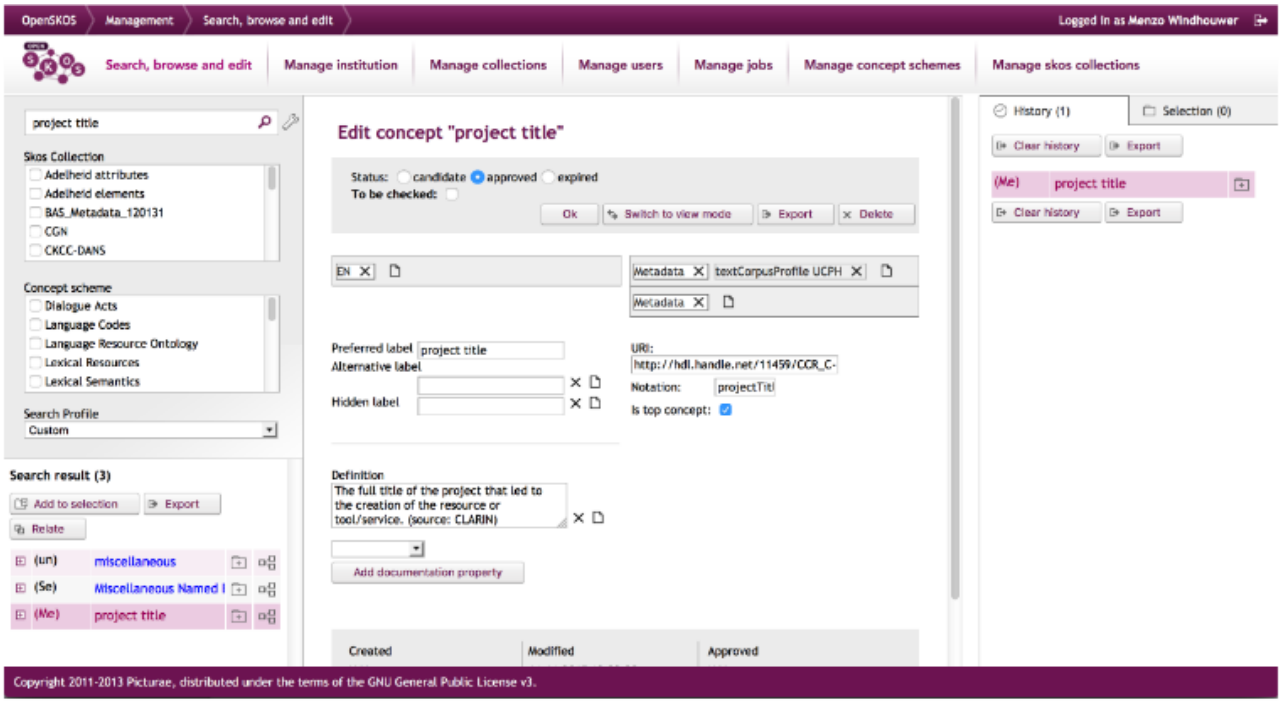

Figure 4. CCR Editor

4.1.3. CLARIN Component Registry

The CLARIN Component Metadata Infrastructure (CMDI) provides a framework to create and use self-defined metadata formats. It relies on a modular model of so-called metadata components, which can be assembled together, to improve reuse, interoperability and cooperation among metadata modellers. The CMDI Component Registry was created to promote the re-use and sharing of metadata components and profiles. The registry contains all CLARIN metadata components and metadata profiles used to describe all metadata. It is expected to contain around 1,000 components and around 200 profiles. Reuse of components and profiles is encouraged as much as possible. Components and profiles are linked to the concepts from CCR for interoperability reason.

The component Registry has the following features:

- Register and store CMDI components/profile
- Enable a user to edit and create Components/Profile

- Enable a user to browse and register Components/Profile

Browsing, searching and using CMDI metadata is also available through the Virtual Language Observatory (VLO).

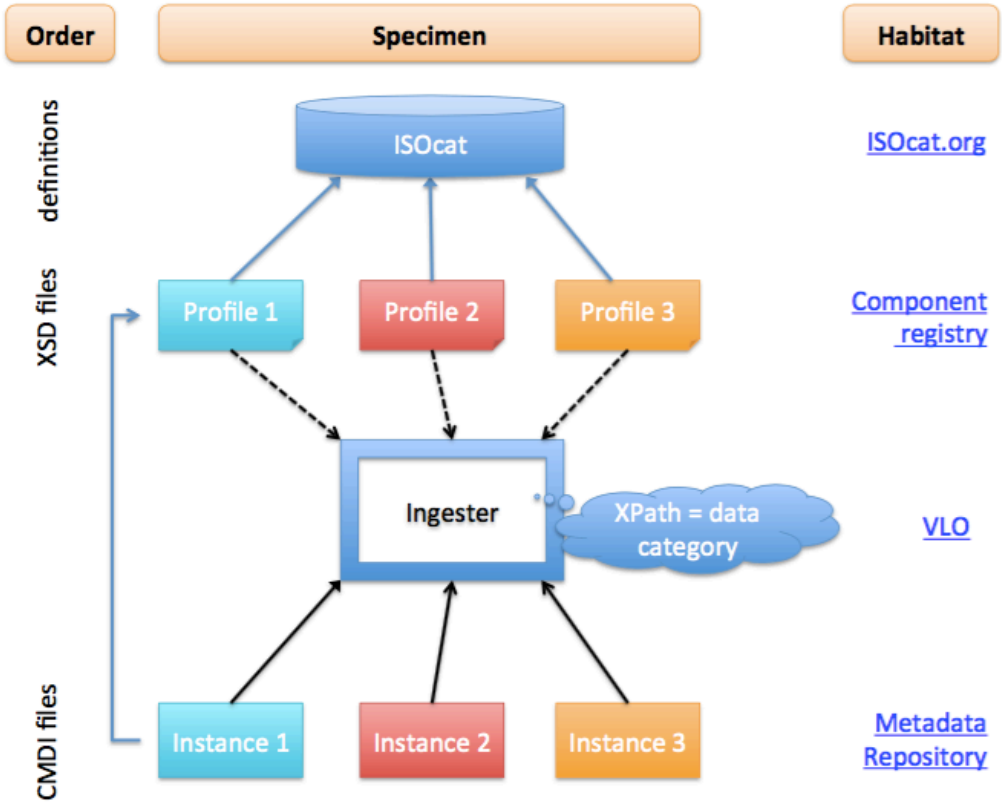

Figure 5. CLARIN CMDI

4.1.4. CLARIN Virtual Language Observatory

The Virtual Language Observatory (VLO)[8] faceted browser was developed within CLARIN as a means to explore linguistic resources, services and tools available within CLARIN and related communities. All information in the VLO faceted browser is based on the metadata descriptions of resources as provided by the parties (CLARIN centres[9]) that host the original data. It gets refreshed regularly but may not be completely up-to-date with the current state of the original data and metadata depending on the date, time and state of the metadata providing/retrieval process (OAI-PMH is employed).

The VLO is designed to help researchers to easily find suitable language resources and tools to carry out their research work. They can do this by searching, browsing and navigating geographically. Once they have found a useful resource they can then easily

[8] https://vlo.clarin.eu
[9] For a list of all CLARIN centres see https://centres.clarin.eu/

find tools with which to work on it. The purpose is that users may directly access the resources or services they have found, given they have the necessary permissions.

The faceted browser allows users to narrow down results using the following facets:

- Language
- Collection
- Resource Type
- Country
- Modality
- Genre
- Format
- Organisation
- Availability
- National Project
- Keyword
- Data Provider

In addition to navigating the resources by means of the selection of facet values, the VLO faceted browser also allows for searching by means of textual queries, by either simply typing terms or by writing advanced queries using Lucene syntax.

4.2. OTHER LANGUAGE RESOURCES REGISTRIES

Besides CLARIN, several other European initiatives have been carried out in the last decade, with the goal of improving the documentation of Language Resources (LRs). Some of these initiatives are now collaborating with the CLARIN efforts, but the unification of language resources descriptions is not yet completed; as a consequence of this, resources that aren't currently documented in the CLARIN VLO may be found in some of the following registries.

4.2.1. META-SHARE Registry

The META-SHARE registry federation was implemented in the framework of the META-NET[10] Network of Excellence. It is designed as a network of distributed repositories of LRs, including language data and basic language processing tools (e.g., morphological

[10] http://www.meta-net.eu/

analysers, PoS taggers, speech recognizers, etc.). Currently six main nodes and several secondary nodes are active[11]. A faceted search allows users to filter results based on:

- Language
- Resource Type
- Media Type
- Availability
- Licence
- Restrictions of Use
- Validated
- Foreseen Use
- Use Is NLP Specific
- Linguality Type
- Multilinguality Type
- Modality Type
- MIME Type
- Conformance to Standards/Best Practices
- Domain
- Geographic Coverage
- Time Coverage
- Language Variety

The META-SHARE data model [4] is described in Figure 6.

[11] http://www.meta-share.eu/

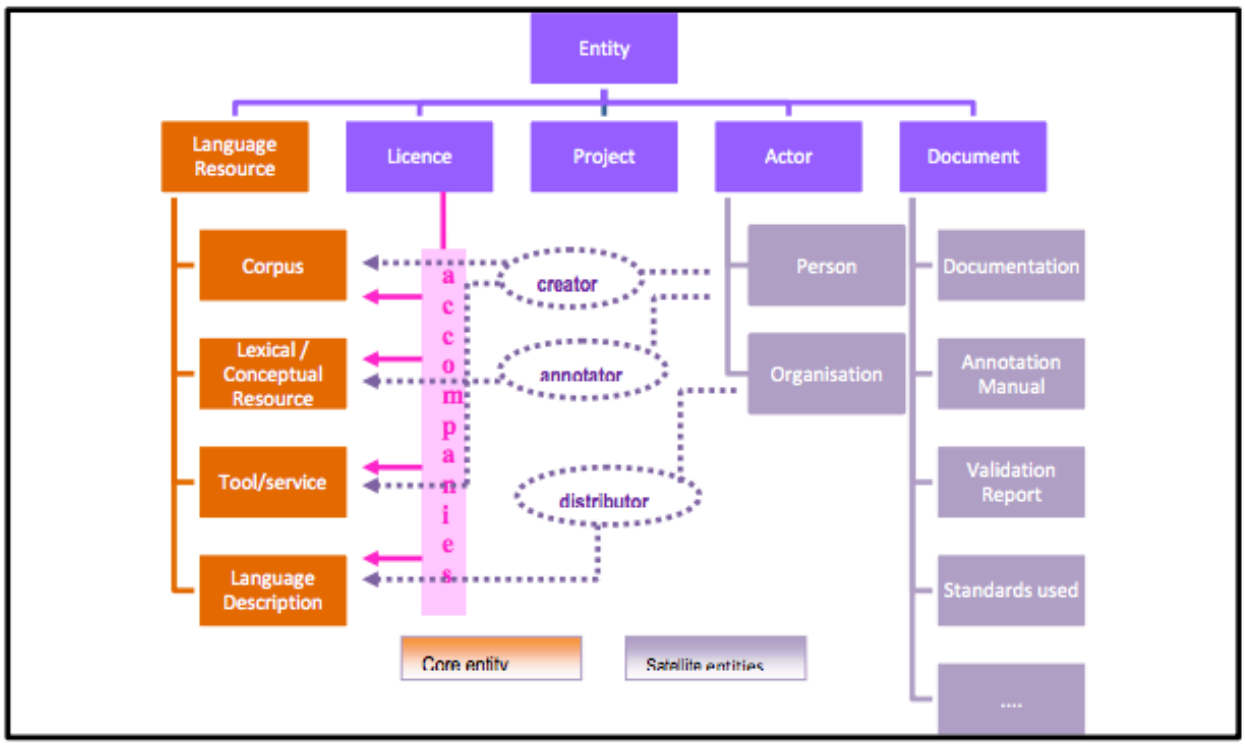

Figure 6. The META-SHARE Catalogue Data Model

The main entities are Language Resource, License, Project, Actor and Document.

As to Language Resources, five different profiles are available, for *Corpora*, *Lexical and Conceptual Resources* (Lexicons, Ontologies…), *Tools and Services* (such as NLP software and online applications) and *Language descriptions* (e.g. language models or grammars).

The META-SHARE nodes are not directly harvestable, but administrators can provide XML dumps of the registry, that can be reused and converted to other standards.

The five main LRs profiles of META-SHARE have been implemented as CMDI profiles as well, so that it is easy to import META-SHARE documented resources into the CLARIN VLO. Two of the original META-SHARE nodes are now CLARIN nodes too, and will soon offer META-SHARE resources for harvesting to the VLO.

4.2.2. LRE MAP

The Language Resource and Evaluation Map initiative issued out of the FLaReNet project[12], whose mission was to develop a common vision of the area of LRs and to foster a European strategy for consolidating the sector, thus enhancing competitiveness at EU level and worldwide. The FLaReNet project produced a set of recommendations [5] for the

[12] http://www.flarenet.eu/

sector of digital Language Resources, encompassing creation, standardization, curation and long-term preservation. The correct documentation of LRs was indicated as crucial, and an initiative at the Language Resources and Evaluation Conference (LREC2010) was launched in collaboration with ELRA, to crowd-source the usage of LRs in papers submitted to the conference.

The initiative continued within the following LREC conferences and extended to other events; today the LRE MAP[13] is a large repository of data, documenting language resources (well known ones, but also minor ones and resources under development) using a lightweight metadata scheme [6]. Authors are asked to enter a description for each language resource (whether their own or those of others) that they have used to carry out the research described in their paper.

- Resource Type, (lexicon, corpus, tool…)
- Resource Name
- Resource Production Status (new, existing, under construction)
- Use of the Resource (a list of possible natural language processing tasks)
- Language(s)
- Modality (speech/written)
- Resource Availability (with information about licenses)
- Resource URL (if available)
- Resource Description
- Resource Size
- Resource License
- Resource Documentation (with possible links to documents)

The LRE MAP is not actually a catalogue of language resources, but a collection of instances of uses of resources, so, for instance, well known and used LRs (e.g. Princeton WordNet, or the British National Corpus) have several entries in the LRE map.

The web search interface allows users to search results by:

- Resource type
- Languages
- Language type (monolingual, multilingual)

[13] http://www.resourcebook.eu/searchll.php

- Modality
- Resource Use
- Production Status
- Conference(s) at which the resource use was “described“ in a paper
- Name.

The LRE MAP is not harvestable, but the LREC2014 dataset is available as a Linked Open Data RDF dump; this RDF version also contains the information on which papers “cite” which resource[14] [7].

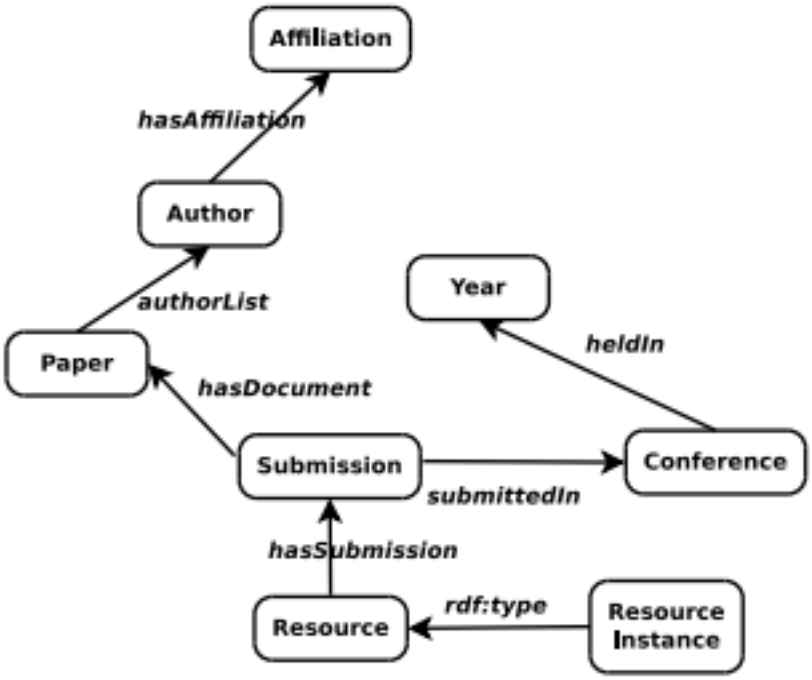

Figure 7. The data schema of LRE-MAP

4.2.3. LINGHUB

Linghub[15] [8] is an initiative launched within the recently concluded LIDER project, and is defined as a comprehensive location for finding information about language resources. It contains information about over 100,000 resources, for over 1,000 languages, drawn from multiple sources, such as the CLARIN VLO, META-SHARE, the LRE MAP and Datahub.

[14] http://linghub.lider-project.eu/datahub/lremap-conf
[15] http://linghub.lider-project.eu/

Metadata are mapped to the Data Catalog Vocabulary (DCAT)[16] that in turn largely re-uses Dublin Core vocabulary.

The web interface allows for the search of entries using the following DCAT metadata properties:

- Title
- Language
- Rights
- Type
- Creator
- Source
- Contributor
- Subject
- Description
- Access URL
- Contact Point

An advanced search facility with a SPARQL endpoint is also available[17].

For the purpose of PARTHENOS, it should be noted that Linghub's main goal is to provide a unified search environment, but it does not contain any new resource descriptions. So, it should not be considered as a first hand source of LR documentation

4.3. DARIAH

DARIAH is a pan-European infrastructure for arts and humanities scholars working with computational methods. It supports digital research as well as the teaching of digital research methods. DARIAH currently connects several hundreds of scholars and dozens of research facilities in 17 European countries. People in DARIAH provide digital tools and share data as well as know-how. The mission of DARIAH (Digital Research Infrastructure for the Arts and Humanities) is to enhance and support digitally enabled research across the humanities and arts. DARIAH aims to develop and maintain an infrastructure in

[16] https://www.w3.org/TR/vocab-dcat/
[17] http://linghub.lider-project.eu/sparql/

support of ICT-based research practices. DARIAH is working with communities of practice to:

a) Explore and apply ICT-based methods and tools to enable new research questions to be asked and old questions to be posed in new ways,

b) Improve research opportunities and outcomes through linking distributed digital source materials of many kinds, and

c) Exchange knowledge, expertise, methodologies and practices across domains and disciplines in Arts and Humanities.

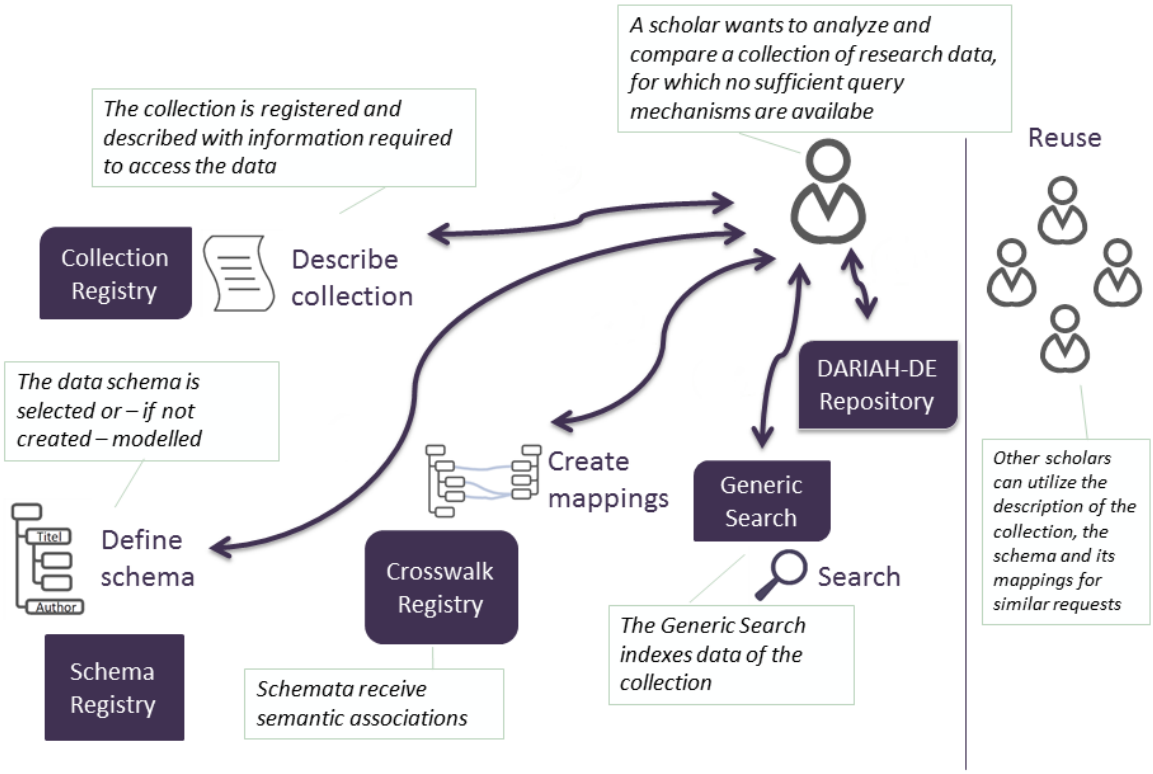

Figure 8. DARIAH

The DARIAH repository contains:

- Publications (journal articles, conference paper, books, posters, patents)
- Documents (preprint, working papers, report…)
- Academic works (theses, accreditation to supervise research, lectures)
- Research data (photos, videos, maps, audios)

The DARIAH website is accessible at http://www.dariah.eu/.

4.3.1. DARIAH collection registry

The Collection Registry is a simple web application that holds information about research collections relevant for arts and humanities research. The term collection herewith refers to a set of entities such as books, images, or statues. A collection description stored in the collection registry contains general information such as the location and access points of the collection. It may also carry collection specific metadata such as e.g. a spatial or temporal coverage of the entities. The main entities of the registry are:

- *Collection* (subject, owner, location, spatialCoverage, contentDataRange, service, user)
- *Agent* (name, language, identifier, address, email, homepage, phone)
- *Location* (title, language, identifier, address, email, homepage, phone, administrator)
- *Service* (title, accessMethod, serviceUrl, function, description, accessControl).

The collection registry provides an OAI-PMH server as well as REST web services to get collection descriptions. The basic access details for the Collection Registry OAI-PMH service are as follows:

- Locator (base URL): http://colreg.de.dariah.eu/colreg/OAIHandler
- OAI-PMH version: 2.0
- Maximum records: 200
- Metadata Formats: OAI_DC (Simple Dublin Core); DCLAP (DARIAH Collection Level Application Profile)
- Not supported: ListSets

The DARIAH Collection registry is available online[18].

4.3.2. DARIAH Schema Registry

The Schema Registry allows the storing of different metadata schemas for use by the Crosswalk Registry and Generic Search. The DARIAH Schema Registry is the central access point to schema descriptions. The Schema registry is available online[19].

[18] http://colreg.de.dariah.eu/colreg/colreg/main;jsessionid=0C1B2FA77A8C210F0350640A7E4F4603?execution=e1s1

[19] https://de.dariah.eu/schema-registry

4.3.3. DARIAH crosswalk registry

The Crosswalk Registry is a graphical tool, enabling researchers in the Arts, the Humanities, and Social Sciences to map different metadata standards stored in the Schema Registry. This mapping allows automated translation from one data schema to another, and that, in turn, allow scholars to use just this one tool in order to search data of different collections.

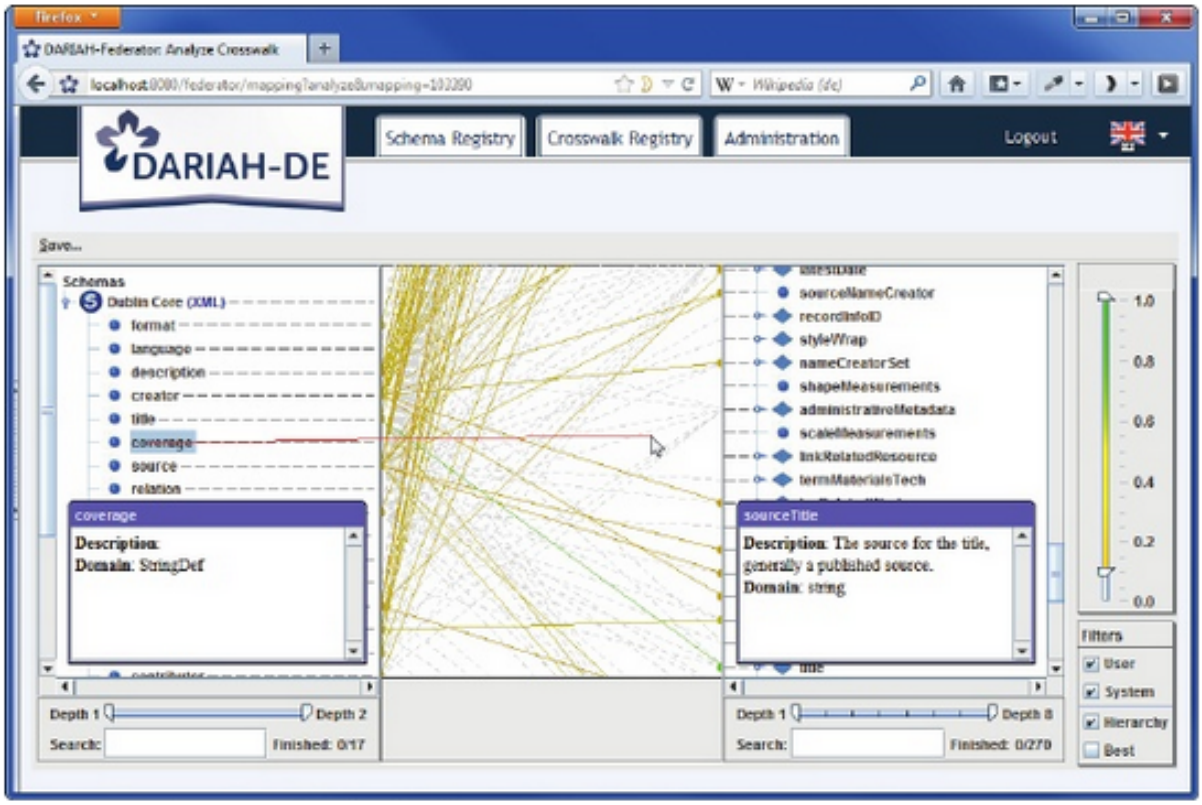

Figure 9. DARIAH GUI for registries

Crosswalk registry is available online[20].

4.3.4. DARIAH-GR/DYAS Organizations and collections registries

The DYAS Organizations and Collections Registries is a web application that provides access information on Greek institutions or individuals and the collections, both analogue and digital, they own or curate. This tool takes advantage of the available expertise and digital resources to improve the quality of users' research and to serve educational purposes. It covers seventeen disciplines of Humanities and Arts and all the fields fall within the Greek History, Culture, Heritage and Language categories. The registry data

[20]https://wayf.aai.dfn.de/DFN-AAI-Test/wayf/WAYF?entityID=https%3A%2F%2Fdev3.dariah.eu%2Fschereg&returnIDParam=idp&return=https%3A%2F%2Fdev3.dariah.eu%2Fschereg%2Fsaml%2Flogin%2Falias%2Fschereg%3Fdisco%3Dtrue

model is based on FOAF for Persons and Organisations and Dublin Core Collections Application Profile (DCCAP) for Collections; it presents, with detailed information the available collections and organisations of Humanities and Arts. The architecture of the system balances between needs of implementation, maintenance cost and interoperability. It is encoded in XML and it can be provided and harvested via OAI-PMH 2.0.

The contained fields of the registries are the following:

- *dc:type & dc:indentifier* about the nature and the URI of the resource;
- *dc:title. dcterms:alternative & [dcterms:abstract* about the title of collection and its possible alternative, and an abstract description of the collection
- *dcterms:extent, dc:language, cld:itemType & cld:itemFormat* about the extent of the collection, the language(s) of the items (especially for texts) and the format of the items.
- *dc:rights, dcterms:accessRights, dcterms:accrualMethod, dcterms:accrualPolicy, dcterms:accrualPeriodicity & dcterms:provenance*, about the rights over the collections, the rights and options of accessibility by the public, and their method of obtaining and expanding the collection.
- *dc:subject*, a controlled vocabulary with hierarchy of the topics that describes the collection
- *dcterms:spatial, dcterms:temporal, dcterms:created, cld:dateItemsCreated, dc:creator & marcrel:own*, about the necessary spatiotemporal information of the collection and its items, its collector and owner
- *cld:isLocatedAt, cld:isAccessedVia, dcterms:hasPart OR dcterms:isPartOf, [cld:associatedCollection] & dcterms:isReferencedBy*, about the location of an analogical collection and the access via web of a digital collection, its sub-collections or the over-collection and the possible connection with other similar collections.

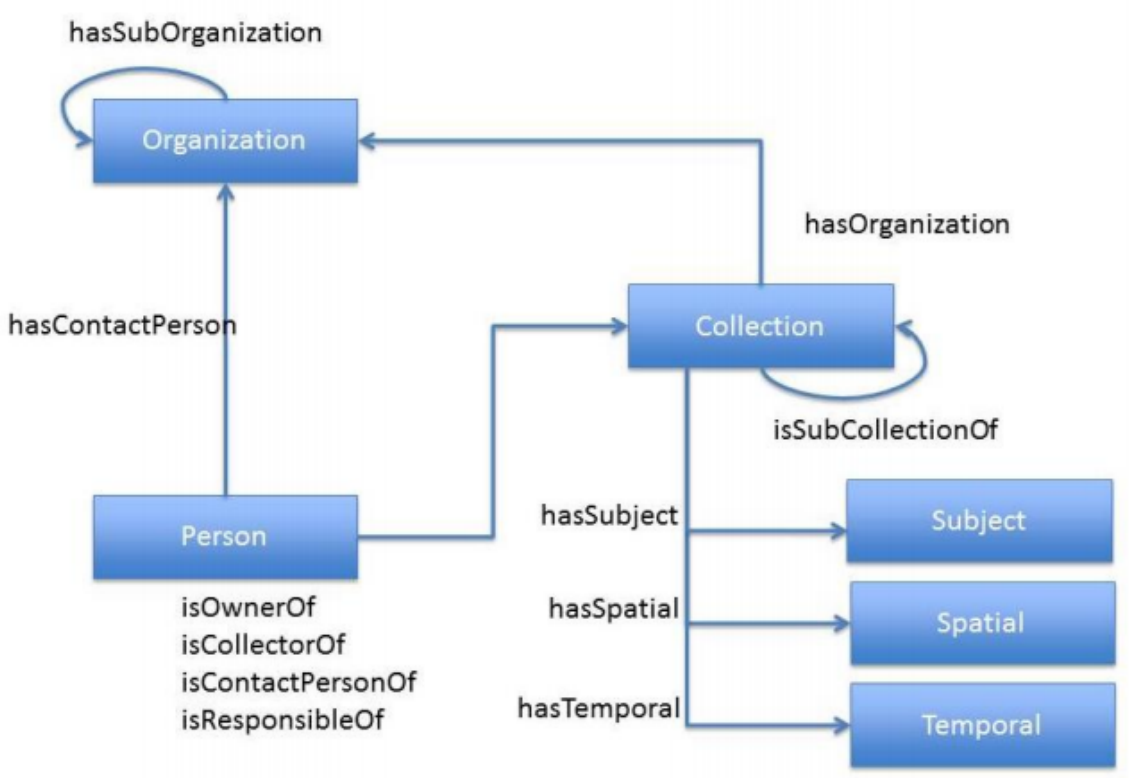

Figure 10. Connections between Persons/Organisations and their Collections

Some of the aforementioned fields are chosen from other classes that are implemented into different tables. These fields are: *Owner* and *Collector* from class *Person*; *Subject* from class *Subject; Spatial Coverage* from class *Spatial; Temporal Coverage* from class *Temporal*.

Additionally, the needs of this registry could not be covered by the DCCAP protocol; therefore, some extra metadata fields have been introduced, such as:

Creative Commons, Metadata schemata, Contact Person Identifier, (full) address, Coordinates/ System of Coordination, Digitization Method, Photography conditions and Scanner characteristics.

DYAS registries are available online[21].

4.4. ARIADNE

The ARIADNE project aims to integrate the archaeological resources made available by the partners of the project for the purposes of discovery, access and integration on a research infrastructure. These resources include data, services and language resources,

[21] http://registries.dyas-net.gr/en

such as metadata formats, vocabularies and mappings. The registry is addressed to cultural institutions, private or public, which wish to describe their assets in order to make them known to e-infrastructures. The registry data model, called ACDM (ARIADNE Catalogue Data Model), extends the DCAT vocabulary [3] and describes the available resources among the various partners of the project. Figure 11 shows a simplified UML diagram of the ACDM, which includes the most relevant classes and associations.

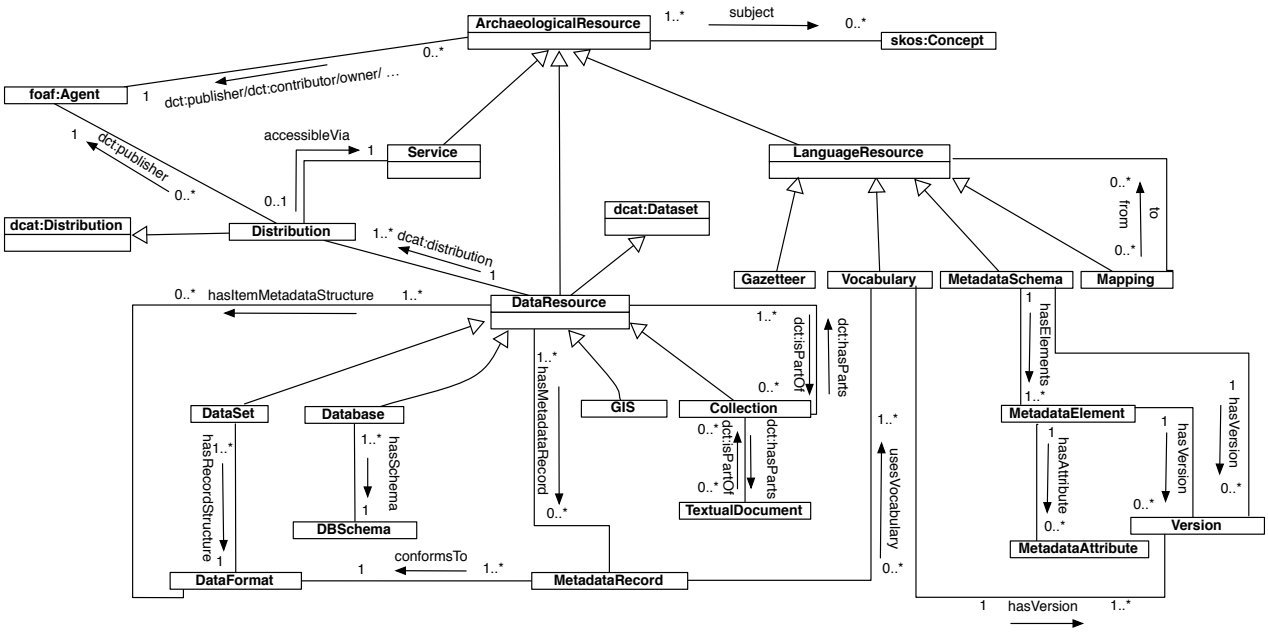

Figure 11. The ARIADNE Catalogue Data Model

As illustrated in Figure 11, the central notion of the model is the class *ArchaeologicalResource,* specialized as:

- *DataResource*, whose instances represent the various types of data containers owned by the ARIADNE partners and lent to the project for integration. This class is created for the unique purpose of defining the domain and the range of a number of associations. It is therefore an abstract class.
- *LanguageResource*, having as instances vocabularies, metadata schemas, gazetteers and mappings (between language resources). To describe language resources, the ISO/IEC 11179 'Specification and Standardization of Data Elements' [1] has been extended.
- *Services*, whose instances represent the services owned by the ARIADNE partners and lent to the project for integration.

The *ArcheologicalResource* class defines the properties common to its subclasses, mostly using the terms of the DCAT vocabulary. The main associations having this class as domain are:

- *dct:publisher:* associates any archaeological resource with an agent responsible for making the resource publicly available.
- *dct:creator:* associates any archaeological resource with an agent primarily responsible for creating the resource.
- *owner:* associates any archaeological resource with an agent that is the legal owner of the resource.
- *legalResponsible:* associates any archaeological resource with a person holding the legal responsibility of the resource.
- *scientificResponsible:* associates any archaeological resource with a person holding the scientific responsibility of the resource.
- *technicalResponsible:* associates any archaeological resource with a person holding the technical responsibility of the resource and contact person.
- *dct:subject:* associates any archaeological resource with a subject drawn from an existing vocabulary.

For a detailed and updated description of ACDM, see the official documentation on the project web site[22].

4.5. CENDARI

The CENDARI project (Collaborative European Digital Archive Infrastructure) is a Research Infrastructure project aimed at integrating digital archives for medieval and modern European history. The project consortium, comprised of 14 institutions across Europe, is working to pilot the development of a research infrastructure, leveraging analogue networks to integrate digital resources for historical researchers. The project aims to build a research infrastructure that is easy to use/access and essential to researchers' goals. The CENDARI Repository holds information on over 1,200 collection holding institutions and over 300,000 records (metadata and some full text) relevant for the study of the medieval period and First World War.

[22] http://www.ariadne-infrastructure.eu

Given the wide range of data sources and paths of entry into the CENDARI system, a flexible technical approach was called for. Instead of applying a classical integration approach and defining a common description format, the CENDARI repository was designed to allow for the coexistence of such heterogeneous content. The classical approach almost always suffers from loss of information, which occurs during translation from the original into a common format accepted by a repository. In addition, it requires substantial intellectual effort and technical work invested upfront in defining and implementing translation rules, such as whether the “common-denominator” or “union-all” principle is used. By contrast, the method adopted by the CENDARI project was based on ensuring a set of common functions over diverse formats and allowing for an evolutionary approach in providing more specific and semantically rich services. The need to perform transformations over collection descriptions, encoded in various formats in order to achieve a certain level of semantic integration, was not avoided. However, the upfront efforts were lower and the system allows for incremental integration over time. Main entities from the model are:

- **Places/spaces**: geographic locations relevant to research topics or other contextual entities.
- **Persons/role**: individuals associated with the research topics or other contextual entities, and their associated roles.
- **Institutions**: organizations associated with the research topics or other contextual entities.
- **Dates**: specific dates or periods of time associated with specific events, people, or organizations.
- **Events**: notable events associated with the particular topics, as well as with other contextual entities.
- **Topics**: subjects associated with the two research areas.

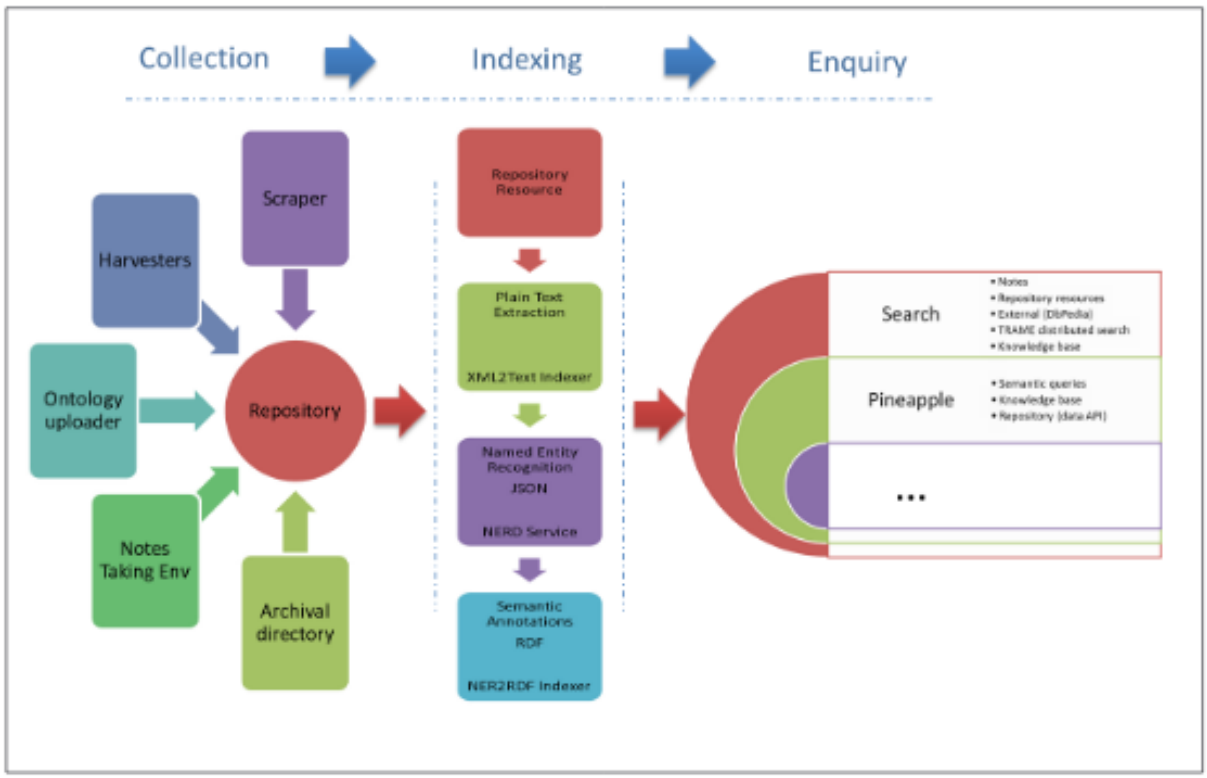

Figure 12. CENDARI

The CENDARI Archival Directory is implemented through AtoM, which stands for Access to Memory. It is a web-based, open source application for standards-based archival description and access in a multilingual, multi-repository environment. AtoM uses the standardized format Encoded Archival Description (EAD) to present information about an archival unit. A mapping of CENDARI Data Model to the Europeana Data Model (EDM) is shown in Figure 13.

The CENDARI Registry offered functionalities are:

- Online GUI for editing
- Search/Browse
- API (REST)
- Import (OAI-PMH)

The CENDARI registry is available at https://archives.cendari.dariah.eu/.

CENDARI contextual classes	EDM contextual classes
Places/spaces: geographic locations relevant to research topics or other contextual entities.	Places
Persons/role: individuals associated with the research topics or other contextual entities, and their associated roles.	Agents
Institutions: organizations associated with the research topics or other contextual entities.	Agents
Dates: specific dates or periods of time associated with specific events, people, or organizations.	Timespans
Events: notable events associated with the particular topics, as well as with other contextual entities.	Events
Topics: subjects associated with the two research area.	Concepts

Figure 13. CENDARI to EDM mapping

4.6. LifeWatch Greece Metadata Catalogue registry

The main goals of LifeWatch Greece[23] data services is to a) ***support cataloguing and publishing of*** all the relevant metadata information of the Greek biodiversity domain, b) ***integrate data from heterogeneous sources*** by supporting the definitions of appropriate models and c) ***efficiently discover biodiversity data*** of interest and enable the answering of complex queries that could not be answered from the individual sources. To achieve these goals, it was mandatory to design and develop a registry of resources, the Directory of LifeWatch Greece.

The role of the LifeWatch Greece Directory (Registry) is to support the discovery of registered resources within the biodiversity information community and return information that allows a user to locate and access the resource and its curator/creator. A number of services have been developed to provide the users with means to create, edit, update, search information about providers, datasets, to access and communicate with them. The directory is independent of the domain.

Special focus was given not only on how to access the datasets or where they are stored but also to whom to communicate with, since the information of how to access the

[23] https://www.lifewatchgreece.eu/

dataset/collection or where it is stored is incomplete without the data that connects it with the curator/creator.

Some indicative queries that can be answered by the Directory Service are the following:

- Which institution hosts the dataset with title “Thunnus Occurences” and how can I have access to it?
- Return all the datasets that have been created from HCMR.

The schema of the directory service is based on CIDOC-CRM[24] and its extensions (CRMdig, CRMsci) and the main entity that is being described is the ***dataset.***

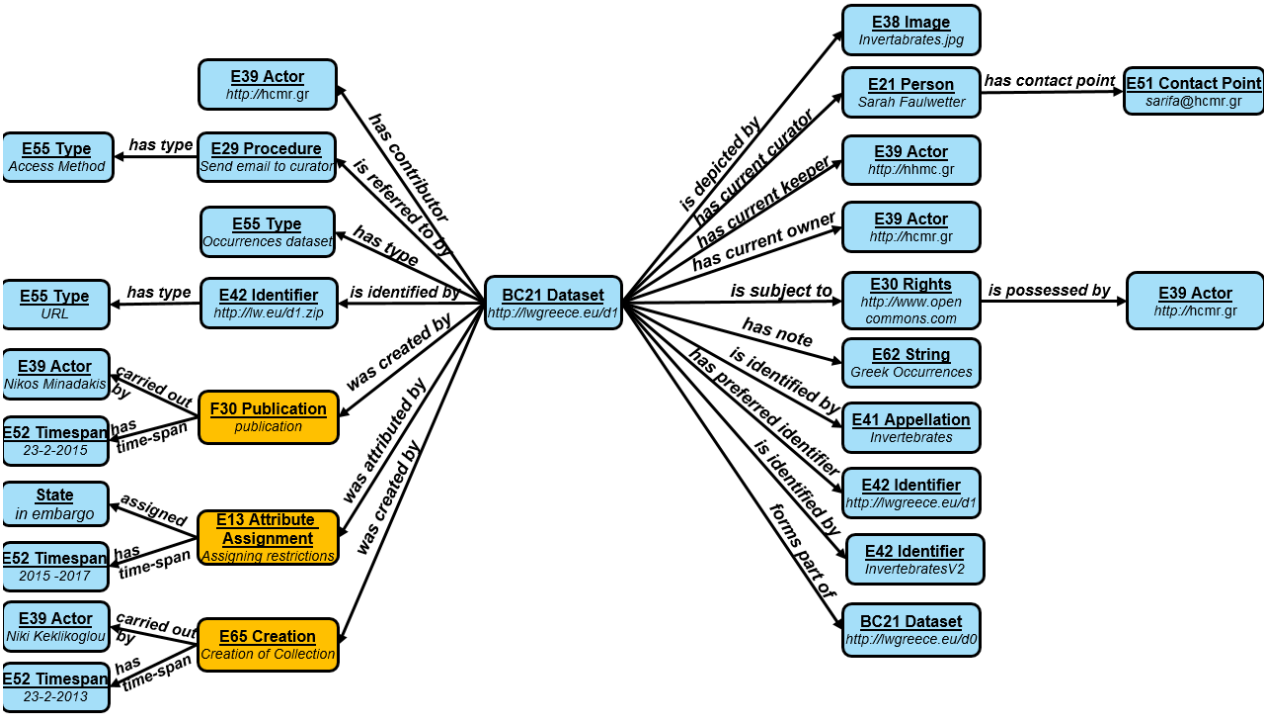

Figure 14. LifeWatch data model

A description of some indicative fields follows:

- ***Dataset*****:** the dataset or the digital collection of datasets that is identified by a unique dataset identifier
- ***Has current owner->Actor:*** the organization or person that owns the dataset, *e.g. HCMR*
- ***Has contributor->Actor:*** the actors that contributed data to the dataset, e.*g. NHMC*

[24] http://www.cidoc-crm.org/

- ***Has current curator->Person:*** the person that manages the dataset, *e.g. Sarah Faulwetter*
- ***Person-has contact point->Contact Point:*** the actual contact point of the curator, *e.g. sarifa@hcmr.gr*
- ***Has type->Type*****:** the type of the dataset, *e.g. Occurrence Records Dataset*
- ***Was created by->Creation:*** the event that created the dataset
- ***Creation-carried out by->Actor:*** The creator of the dataset.

There are currently a JAVA API[25] and a number of SOAP web services that implement the LifeWatch Greece Directory. Furthermore, a web application has been developed and integrated in the LifeWatch Greece portal[26] in order to provide the capabilities that the directory offers to the LifeWatch's users.

4.7. Open Metadata Registry

The Open Metadata Registry is a fundamental piece of technical infrastructure for the Semantic Web. While originally built to support the National Science Digital Library (NSDL), the Registry is openly available to all who wish to use its services. The Registry provides a means to identify, declare and publish through registration, metadata schemas (element/property sets), schemes (controlled vocabularies) and Application Profiles (APs). In addition to support registration of schemes, schemas and APs for consumption and use by human and machine agents, the Open Registry will support the machine mapping of relationships among terms and concepts in those schemes (semantic mappings) and schemas (crosswalks). Thus, the Registry will support the key goals of metadata discovery, reuse, standardization and interoperability both locally and globally. The Registry used as its inspiration the open-source Dublin Core Metadata Initiative (DCMI) Registry. The Registry extended the original DCMI goals to support:

- The automated creation and maintenance of schemas and application profiles;
- The submission of schemas and schemes to a registry workflow for review and publication.

[25] https://github.com/isl/LifeWatch_Greece
[26] http://metacatalogue.portal.lifewatchgreece.eu/

All of the development work leverages the latest knowledge and standards for networked knowledge organization systems, schema and application profile declaration, and registry development. The Open Metadata Registry project was funded by the National Science Foundation for its first three years. It is currently managed by Metadata Management Associates, a consulting partnership committed to maintaining the Registry as an open system.

4.8. Tools E-Registry for E-Social science, Arts and Humanities (TERESAH)

TERESAH (Tools E-Registry for E-Social science, Arts and Humanities) is a cross-community tools knowledge registry aimed at researchers in the Social Sciences and Humanities. It aims to provide an authoritative listing of the software tools currently in use in those domains, and to allow their users to make transparent the methods and applications behind them.

TERESAH has been developed as part of the Data Service Infrastructure for the Social Sciences and Humanities (DASISH), a Seventh Framework Programme funded project. DASISH collaborates with the five ESFRI Infrastructures in the field of Social Science and Humanities (CESSDA, CLARIN, DARIAH, ESS, and SHARE). The tools and knowledge registry is aimed at researchers from all disciplines and sectors, research infrastructure builders and users, as well as IT personnel. It aims to include information about tools, services, methodologies, and current standards and makes use of existing social media for dissemination and discussions. TERESAH is open source software and has been developed with a reusability plan in mind, meaning that anyone can install and run a TERESAH instance of their own with minimal effort required. Partners who has contributed to TERESAH include King's College London, Swedish National Data Service, Finnish Social Science Data Archive, University of Tartu, Universitat Pompeu Fabra and CentERdata. It provides functions of search and browsing in different ways, access to metadata via an API, import and export features in RDF or JSON formats. The metadata format used to describe the tools is the Dublin Core.

4.9. CulturaItalia catalogue

CulturaItalia is the Portal of Italian Culture, managed by the Central Institute for the Union Catalogue of Italian Libraries (ICCU) of Ministry of cultural heritage, activities and tourism (MiBACT). CulturaItalia, as national aggregator, plays an important role for the development of European RIs on Cultural Heritage such as ARIADNE, DARIAH and Europeana, making available cooperative networks and agreements and coordinating technical activities.

The CulturaItalia's catalogue, currently manages 3,160,965 metadata records from museums, libraries, foundations and institutions both public and private. CulturaItalia contains a section devoted to open data (http://dati.culturaitalia.it), where it's also possible to search via SPARQL END POINT, a datasets selection available in Linked Open Data format. The CulturaItalia's catalogue is continuously enriched by new metadata through the OAI-PMH servers of the data providers.

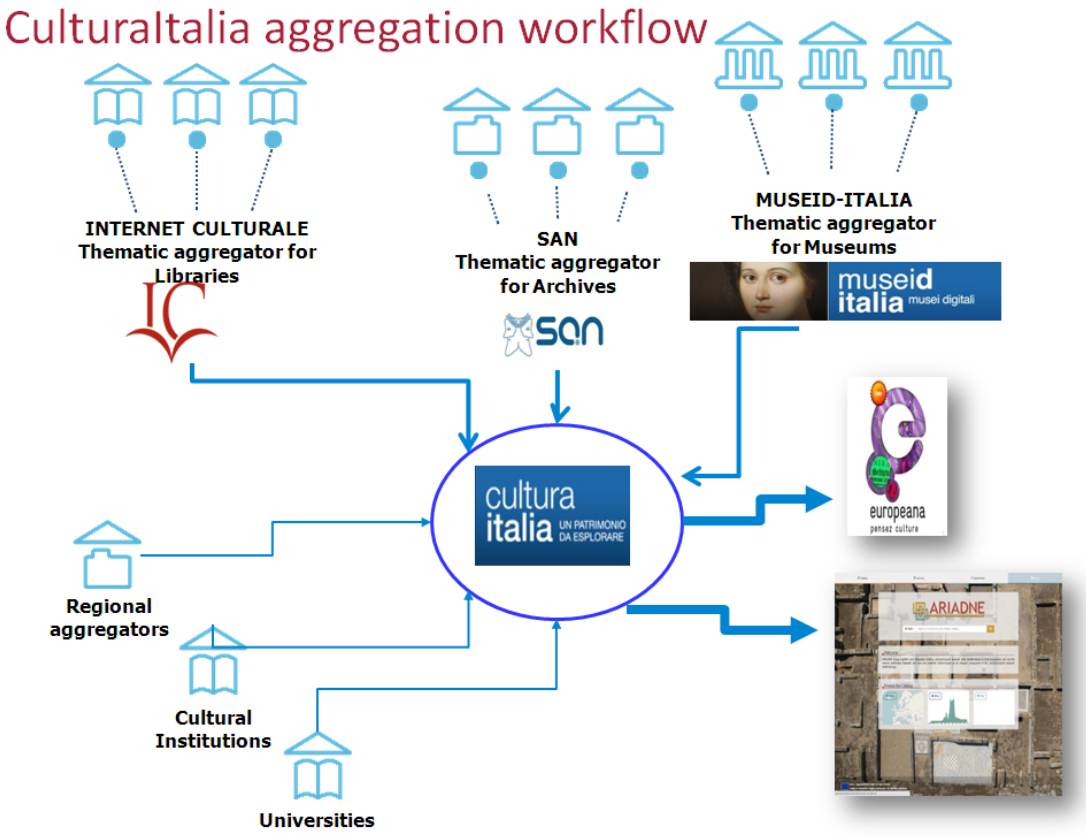

Figure 15. CulturaItalia infrastructure

4.9.1. Application profile

A specific Dublin Core Application Profile has been designed in order to cover the complex domain of the “Italian Culture” and to guarantee the interoperability of various kinds of cultural resources. This application profile is called PICO AP from the name of the Project in whose context the CulturaItalia portal was developed.

The PICO AP combines in one metadata schema all DC Elements, all DC Element Refinements and Encoding Schemes from the Qualified DC and other refinements and encoding schemes specifically conceived to retrieve information pertaining to Italian culture.

In particular, it has been integrated in the PICO AP different systems of cataloguing of cultural heritage (works and objects of art, books and archival documents) by enabling through a single point of access, their research and browsing.

The PICO AP can be consulted at http://purl.org/pico/picoap1.0.xml. Schemas used for the PICO AP are published on a PURL, under the domain PICO: http://purl.org/pico/1.1/pico.xsd and http://purl.org/pico/1.1/picotype.xsd.

The decision to avoid the introduction of new elements, adding solely new element refinements and encoding schemes, was due to the priority of assuring total interoperability with system based on DC.

To ensure the quality of data, some mandatory field values are required to the data providers (*dc:title, dc:identifier, dc:type* and *dc:subject*, with a value extracted by the encoding PICO Thesaurus). Recommended field values to be supplied are: *pico:author, dc:creator, dcterms:temporal, dcterms:spatial, dc:date, dcterms:DateCreated, dc:description, dc:publisher, dc:language, pico:preview*.

4.9.2. Linked Open Data

The Section dati.culturaitalia.it started in 2012 to build up a Linked Open Data (LOD) Service that makes available open data sets from the web-portal CulturaItalia.

The CulturaItalia repository allows the semantic enrichment with four types of reference resources:

- authority files like the VIAF (Virtual International Authority File: www.viaf.org)
- GeoNames (www.geonames.org/)
- PICO Thesaurus in SKOS

- DCMI Type vocabulary

The SPARQL endpoint provides access to RDF metadata structured according to the CIDOC - Conceptual Reference Model in the implementation of Erlangen CRM/OWL. Data can be searched over three querying interfaces, corresponding to three sections of dati.culturaitalia.it:

- Text search: here it is possible to perform free text searches over all triples contained in dati.culturaitalia.it.
- SPARQL query: here you can try your hand at a SPARQL query. There are also some examples of queries.
- iSPARQL query: here there is an even more complex querying interface for advanced users.

In Dati.CulturaItalia an OAI Provider is available that makes available XML or RDF metadata structured according to the following different schemas:

- oai-dc (xml): OAI-PMH schema adopted by Open Archives Initiative Protocol for Metadata Harvesting
- pico (xml): PICO Application Profile, the CulturaItalia Application Profile
- edm (rdf): Europeana Data Model, adopted by the portal Europeana EDM27
- cidoc (rdf): CIDOC - Conceptual Reference Model in the implementation of Erlangen CRM / OWL

4.10. The European Holocaust Research Infrastructure (EHRI) portal

The EHRI online portal is one of the key outcomes of the EHRI project. It is an online environment that provides users with free access to rich information about Holocaust-related archival institutions and their collections across Europe and beyond. It further provides users with a range of tools to find, explore, organize and share such information.

The EHRI portal is trans-national in scope, containing information about Holocaust-related archival institutions in more than fifty countries. Both the content and the functionality of the EHRI portal are constantly evolving. The main information elements of the EHRI portal conform to the conceptual standards proposed by the International Council on Archives[28]:

- ISAD(G): General International Standard Archival Descriptions

[28] http://www.ica.org

- ISDIAH: International Standard for Describing Institutions with Archival Holdings
- ISAAR(CPF): International Standard Archival Authority Record for Corporate Bodies, Persons and Families.

And the related following encoding standards:

- EAD: Encoded Archival Description
- EAG: Encoded Archival Guide
- EAC(CPF): Encoded Archival Context (Corporate Bodies, Persons, Families)

Additionally, thesaurus data is encoded in a manner aligned with the Simple Knowledge Organisation System (SKOS).

The EHRI catalogue currently contains more than 150,000 descriptions of archival materials, 474 descriptions of archival institutions that hold archival materials and authority files on 3,231 Corporate Bodies and 620 Personalities related to the history of the Holocaust.

In addition to providing search and browse functionality, an administrative interface allows EHRI editors to add new descriptions/amend existing descriptions. Data ingest is currently available internally via an HTTP-based web service interface accepting JSON, SKOS RDF, and XML (EAD, EAC) formats. Data is currently exportable as structured formats EAD, EAC, and EAG XML. Additionally, a system of pre-set database queries allows ad-hoc data export as JSON or CSV, though currently the management interface for this facility is restricted to administrative users. A JSON-based HTTP API is currently under development but its precise mechanism is still under review.

4.11. Huma-Num

Huma-Num is a major research infrastructure (TGIR) aimed at facilitating the turning of digital research in the humanities and social sciences.

4.11.1. Huma-Num's Infrastructure and Services

The TGIR Huma-Num offers services dedicated to the production and reuse of scientific data. To do this, Huma-Num supports research teams throughout their digital projects to allow the sharing, reuse and preservation of data thanks to a chain of devices focused on interoperability. The aim is to foster the exchange and dissemination of metadata, but also of data itself via standardized tools and lasting, open formats. These tools, developed by

Huma-Num, are all based on Semantic Web technologies, mainly for their auto-descriptive features and for the enrichment opportunities they enable. Other interoperability technologies complement them, such as the OAI-PMH. Interoperability is used internally to allow Huma-Num services to communicate with one another and externally to let users plug their tools into Huma-Num services (http://www.huma-num.fr/).

Another important point is to make the storage of data **independent** of the device used to disseminate the data. These services embrace the research data life cycle.

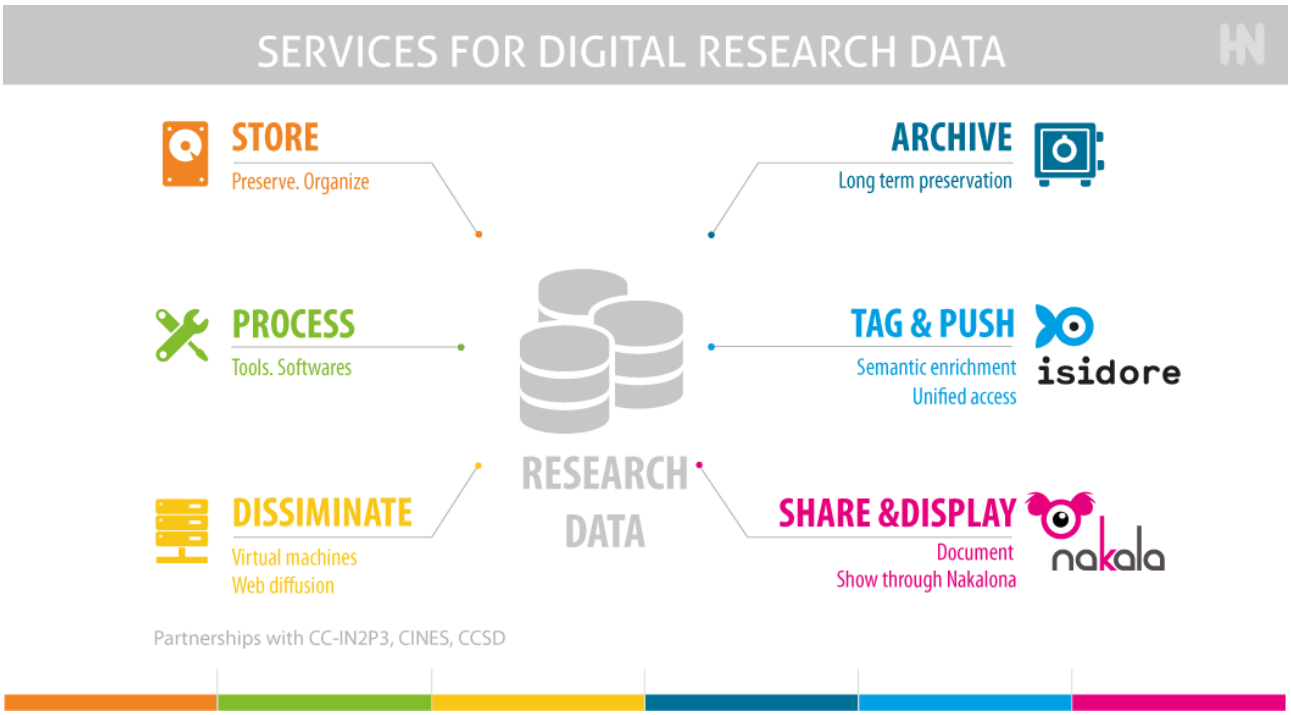

Figure 16. Huma-Num Services

4.11.2. NAKALA service: Share and Disseminate research data

Noting that many teams and research projects do not have the necessary digital infrastructure that will provide a persistent and interoperable access to their digital data, Huma-Num has implemented a service called NAKALA exposure. NAKALA offers three types of services: one to give access to the data, another one to expose metadata and one to give PID to both access DATA & METADATA. By relieving scholars of technical management, it enables them to concentrate on the scientific value of their data. MetaData can be shared via a SparqlEndPoint or via the OAI-PMH protocol and searched.

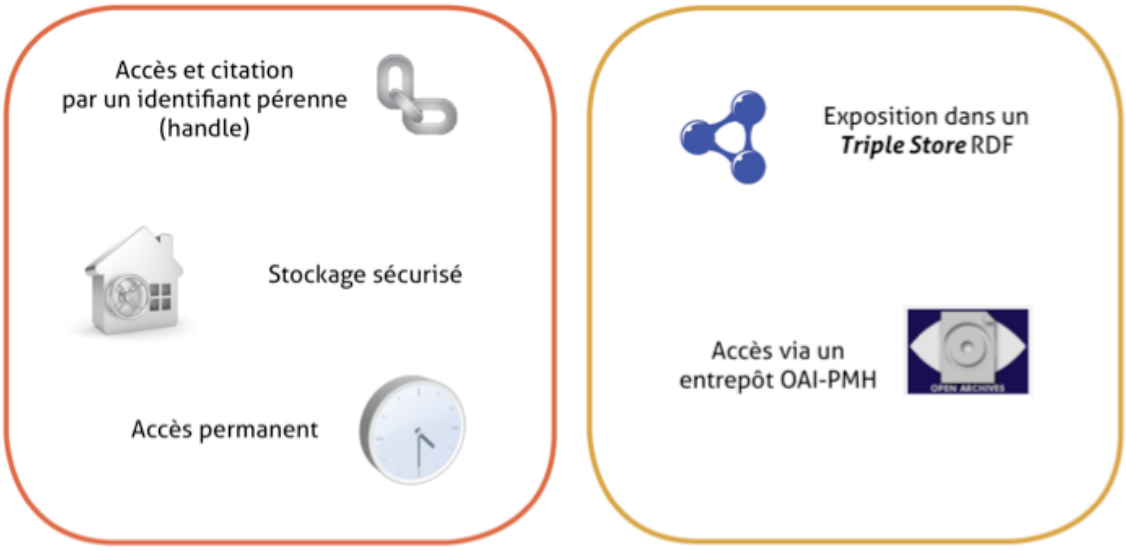

Figure 17. Nakala Services

The key points of NAKALA are the following:

- Use of W3C standard languages;
- Secure servers located in Europe;
- Sustainability. NAKALA is managed by the very large Facility Huma-Num which is a unit of the CNRS, guaranteeing continuity of service;
- PID management based on Handle;
- Open source development based on reliable and proven components (Handle server, OAI PMH server PROAI and Triple Store server Virtuoso)

Data hosted by Nakala may be editorialized with the NAKALONA pack [29] (combining Omeka and Nakala).

[29] http://www.huma-num.fr/services-et-outils/diffuser

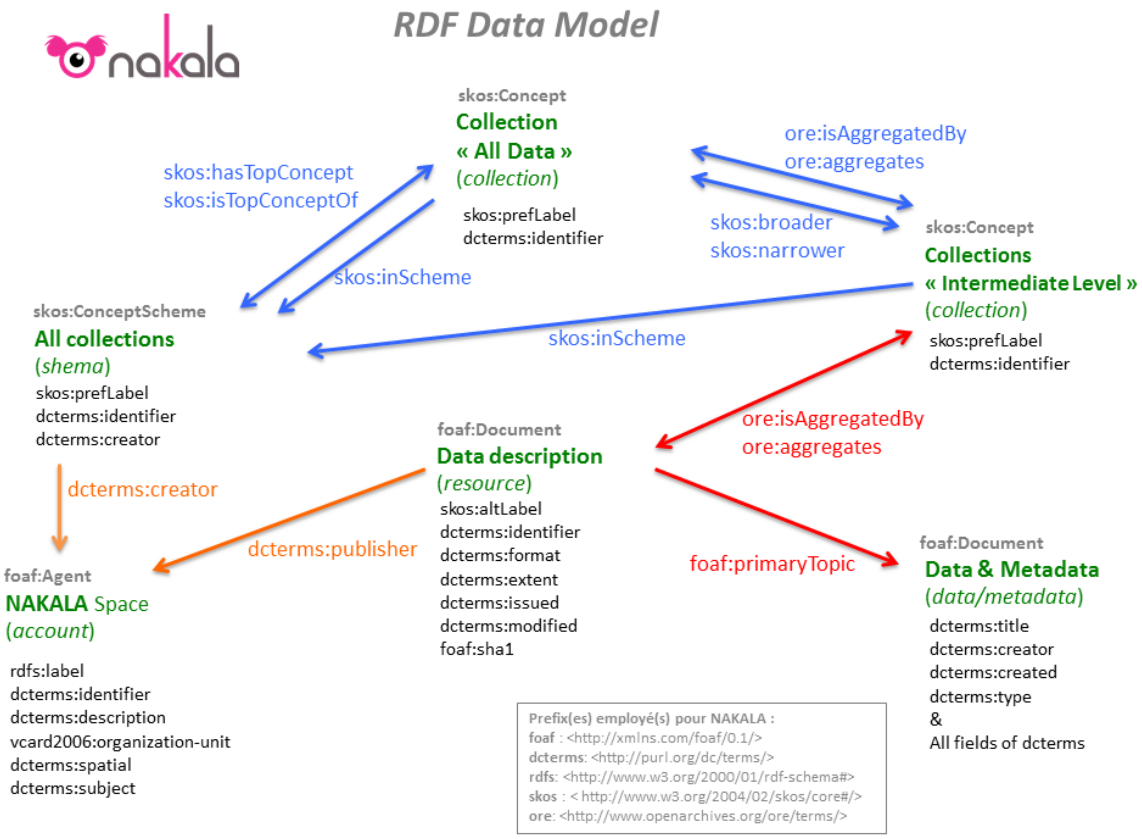

Figure 18. NAKALA RDF model

4.11.3. ISIDORE service: Tag and push Data

ISIDORE is a platform allowing access to digital data in the Humanities and Social Sciences. Its architecture relies on the languages of the semantic web (RDF/RDFS/OWL) and provides open access to data.

The key points of the ISIDORE platform are the following:

- ***Targeted harvesting*** of metadata and scientific data structured according to international standards available in open access;
- ***Indexing of unstructured data*** (full text of a scientific article, for example) and of structured data (documentary metadata, for example);
- ***Standardization*** of metadata and enrichment of data relying on vocabularies recognized in the community (DC, DCterms, FOAF, ORE, RDFS, SKOS);
- ***Multilingual (English, French, Spanish) search GUI*** exploiting the richness of structured data and vocabularies;
- ***SparqlEndPoint*** on sources and indexed data: In 2013, the TGIR Huma-Num and DANS developed a prototype (Proof of Concept) in order to show the connection

between two repositories NARCIS (DANS) and ISIDORE. The connections relied on the two SparqlEndpoints and the alignment of the disciplinary vocabularies used by these systems (cf. Annex). This proof of concept demonstrates the compatibility between ISIDORE and OPENAIRE. In other words, a query in OPENAIRE can dynamically search in ISIDORE and/or in other SparqlEndPoints managed by other stakeholders. This kind of decentralized architecture, based on several SparqlEndpoints, is more resistant to failure and ensures scalability and sustainability.

- ***Smartphone and responsive design*** applications;
- ***Supplying*** metadata enriched by several multilingual thesauri;
- ***Possible integration of the search engine Isidore*** in another environment by providing API and widgets.

Datasets typology & census:

- Archival materials (54467)
- Art exhibition (12896)
- Articles (1288281)
- Bibliography (50320)
- Blog posts (180873)
- Books and book chapters (476423)
- Books collection (1361)
- Conferences and symposiums (109019)
- Images and photos (625419)
- Learning Object (721)
- Manuscripts (60626)
- Maps (58060)
- Memorandum (8586)
- Multimedia materials (104219)
- Others (230790)
- Periodicals (19152)
- Preprint (10579)
- Public slideshow (1897)
- Recensions (41798)
- Scores (17191)
- Seminars (8593)
- Survey data (7230)
- Textual materials (662309)
- Thesis (46536)
- Web Page (557129)
- Website (11)

Table 7. Huma-Num Dataset types

Figure 19 shows a simplified Data Model of ISIDORE.

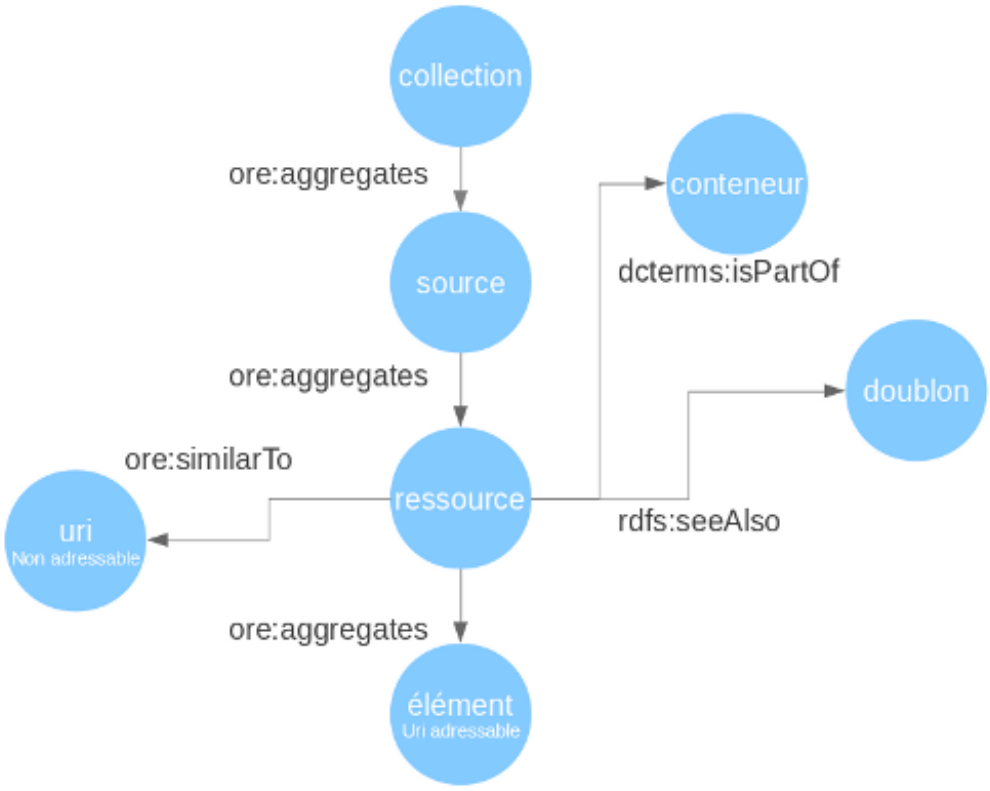

Figure 19. ISIDORE simplified Data Model

5. Analysis of registries survey

Starting from the detailed registries descriptions of the previous section, in the following paragraphs we report an analysis of the results of the survey, to identify the main entities and functionalities that form the basis from which to start to define a suitable data model for the PARTHENOS Joint Resource Registry.

5.1. Entities analysis

Table 8. Entities in the surveyed registries" overleaf summarizes the significant entities in the surveyed registries, as provided by the project partners, by filling in Table 4 and by our subsequent investigation into the registries web sites.

	Dataset	Institution / Organization	Person / Actor	Edu. Packages	Service	Ontology / Vocabulary / Schema	Software	Topics	Events	Date	Location	Procedure
CLARIN Concept Registry (CCR)	YES (concepts)											
CLARIN NL Resources list	YES (Text, pdf, xml)	YES	YES	YES								
CLARIN Component Registry (CMDI)	YES (metadata)											
CLARIN Virtual Language Observatory (VLO)	YES	YES	YES									
LRE MAP	YES	YES	YES		YES	YES	YES					
Metashare Repository	YES				YES		YES					
LingHub	YES	YES	YES		YES	YES	YES					
CENDARI ARCHIVAL Descriptions	YES (Places)	YES (Agents)	YES (Agents)					YES (concepts)	YES	YES (time span)		
DARIAH collection registry	YES (collection)		YES (Agents)		YES						YES	
DARIAH schema registry	YES (metadata schema)											
DARIAH-GR/DYAS	YES (collection)	YES	YES									
CulturaItalia Catalogue	YES (museum, archive, library)	YES										
ARIADNE	YES (database, collection)	YES	YES		YES	YES						
EHRI	YES (archival descr.)	YES	YES		YES	YES						
Huma-Num	YES (archival descr., books, images, media)					YES						

Table 8. Entities in the surveyed registries

5.2. Function analysis

Following the same approach as in the previous paragraph, we analysed the features offered by the various registries, by considering the information provided by our partners using Table 5 and our more in-depth investigation into documents found on the web.

	Browsing	Search	Import (OAI-PMH)	API (Rest)	Online GUI for Editing	Export (XML)	Export (RDF)	SPARQL Endpoint	Export via Web Service	Import (Various Formats)	Export (Various Format)	Tools for Editing and Enrichment
CLARIN Concept Registry (CCR)	YES	YES	YES	YES								
CLARIN NL Resources list	YES	YES	YES	YES								
CLARIN Component Registry (CMDI)	YES	YES	YES	YES	YES							
CLARIN Virtual Language Observatory (VLO)	YES	YES	YES	YES								
LRE MAP	YES	YES										
Metashare Repository	YES	YES			YES	YES						
LingHub	YES	YES	YES	YES			YES	YES				
CENDARI ARCHIVAL Descriptions	YES	YES	YES	YES	YES							
DARIAH collection registry	YES	YES	YES	YES								
DARIAH schema registry	YES											
DARIAH-GR/DYAS	YES	YES	YES		YES				YES			
LifeWatch Greece Directory	YES	YES	YES	YES	YES		YES					
CulturaItalia Catalogue	YES	YES	YES		YES			YES				
ARIADNE	YES	YES	YES	YES	YES					YES	YES	
EHRI	YES	YES		YES	YES					YES	YES	
Huma-Num	YES	YES	YES	YES				YES				YES

Table 9. Surveyed registries functions

6. The PARTHENOS Joint Resource Registry Data Model

At the PARTHENOS WP5 & WP6 Kick-off Meeting (Crete June 9-11, 2015), attendees agreed to define the features of a Joint Resource Registry that will expose a minimal set of metadata describing datasets, software, services, mappings, and more.

The definition of the main entities for the PARTHENOS's Joint Resource Registry starts from the analysis of the user's requirements with respect to the information acquired by the census and from the general semantic framework defined in T5.1. The PARTHENOS's Joint Resource Registry Data Model has been designed to enable the cross-discipline resource discovery and integrated service, in particular to:

- Describe uniformly data presented to the infrastructure thus contributing to the reuse of data and its interoperability,
- Answer research questions through resource discovery,
- Pilot "on the fly" mapping processes in case it will be used in more than one archive.

Starting from the results of the census, whose entities are summarized in Table 8, the main concepts identified has been grouped into five entities, able to manage both the process of knowledge generation and resource discovery:

1. **Dataset** *is a set or collection of data, records or information that is kept as a persistent unit of information in the knowledge generation process.*

   This concept occurs in different forms or denominations in the various registries and includes:

   a. The CLARIN Dataset of concepts (CCR), Dataset of metadata (CMDI), Educational package and textual file of different format (NL Resource List).

   b. The ARIADNE *DataResource*, a class describing resources that are data containers (such as databases, GIS, collections or datasets) and the ARIADNE *LinguisticResources,* a class that has as instances resource of a linguistic nature, whether in natural language (such as a gazetteer) or in a formal language (such as a vocabulary, metadata schema and mapping between schemas). *LinguisticResources* (vocabulary, authority files) are also in the EHRI registry and in the LRE Map and LingHub registries (ontology).

c. The Dataset of archival descriptions (EHRI, CulturaItalia), Datasets from museum, library (CulturaItalia), Dataset of collections (DARIAH-GR/DYAS).
d. The CENDARI Archival descriptions: Dataset of Places, Topics (Concepts), Events and Dates (Timespans).
e. All Huma-Num datasets (see Table 7), vocabularies and collections.

2. **Actor** *is an institution, a team or an individual person that participates in the research infrastructure as partner providing data and/or services*.
   This entity is almost present in every surveyed registry with the name: Person, Organization, Institution, or Agent.
3. **Service** *is defined as the continued, declared willingness and ability of an actor to execute on demand certain activities of specific benefit to the client*.
   Most of the surveyed registries contain such an entity without any specification, while ARIADNE classifies services as *StandAloneService*, *WebService*, *ServiceForHumans*, *InstitutionalService*.
4. **Software** *is an artefact that can be executed on a computer to perform specific operations.*
   This entity is in *LRE Map* and *LingHub* registries. It is also present in the ARIADNE registry as a specialization of the Service class with the name of *StandAloneService*.
5. **Knowledge generation process** *represents the workflow of the processes used to produce specific datasets.*
   This entity is found in the LifeWatch registry under the name Procedure.

To implement these entities it was decided to define a basic model to model, describe, and manage entities and relations. It defines basic descriptions and operations that are common to all entities. This model has been named Information System Model, or simply IS Model, and is described in Section 6.1. Then the basic entities defined by the IS Model has been specialized to model, describe, and manage all entities and relations identified by the general semantic framework defined in T5.1. This model is reported in Section 6.2.

6.1. IS Model

6.1.1. Basic Concept

IS Model identifies common properties of Entities and Relations.

Two typologies of **Entities** are envisaged:

- **Resources**, i.e. entities representing a description of "thing" to be managed;
  Every Resource is characterised by a number of Facets.
- **Facets**, i.e. entities contributing to "build" a description of a Resource. Every facet, once attached to a Resource profile captures a certain aspect / characterization of the resource;
  Every facet is characterised by a number of properties;

Two typologies of **Relations** are envisaged:

- **isRelatedTo**, i.e. a relation linking any two Resources.
- **consistsOf**, i.e. a relation connecting each Resource with one of the Facets characterizing it.

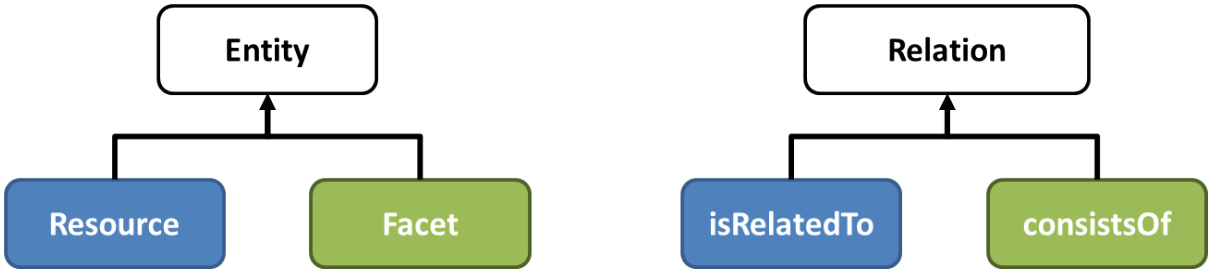

Figure 20. Entity and Relation Typologies

Relations are characterized by the following properties:

- Any relation has a direction, i.e. a "source" (**out** bound of the relation) and a "target" (**in** bound of the relation). The relation can be also navigated in the opposite direction;
- It is not permitted to define a **Relation** having a **Facet** as "source". This also means that it is not permitted to define a **Relation** connecting a **Facet** with another one;
- It is not permitted to define a **Relation** connecting a **Facet** with a **Resource** (as target);
- The same **Facet** instance can be linked (by **consistsOf** or any specialization of it) from different **Resources**.

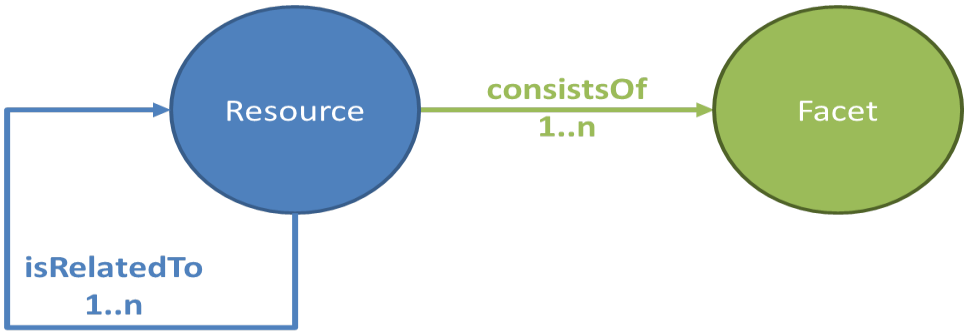

Figure 21. Relation Characterizations

Each **Entity** and **Relation:**

- has an **header** automatically generated for the sake of identification and provenance of the specific information;
- can be **specialized.**

The Header automatically filled by the System has the following attributes.

Name	Type	Attributes	Description
uuid	UUID	Mandatory=true NotNull=true ReadOnly=true	This uuid can be used to univocally identify the Entity or the Relation
creator	String	Mandatory=true NotNull=true ReadOnly=true	Filled at creation time. The creator is retrieved using the authorization token
creationTime	Date	Mandatory=true NotNull=true ReadOnly=true	Creation time in milliseconds. Represent the difference, measured in milliseconds, between the creation time and midnight, January 1, 1970 UTC
lastUpdateTime	Date	Mandatory=true NotNull=true	Last Update time in milliseconds. Represent the difference, measured in milliseconds, between the last update time and midnight, January 1, 1970 UTC

A number of specializations are identified below. Such specializations are managed by the gCube Core services, i.e. Core services builds upon these specialization to realize its management tasks. Authorized clients at runtime can define other specializations since the system makes it possible to store these additional typologies of relations and facets and to discover them.

Facet and **Relation** instances can have additional properties, which are not defined in the schema (henceforth schema-mixed mode) and are defined at runtime.

Any Property can be enriched with the following attributes:

- **Name**: Property Name
- **Type**: The Type of the Property (e.g. String, Integer, ...). See *Property Type*
- **Description**: The description of the Property. default=null.
- **Mandatory** (**M**): Indicate if the Property is mandatory. default=false.
- **ReadOnly** (**RO**): The Property cannot change its value. default=false.
- **NotNull** (**NN**): Whether the property must assume a value diverse from 'null' or not. default=false
- **Max** (**Max**): default=null
- **Min** (**Min**): default=null
- **Regexpr** (**Reg**)): A Regular Expression to validate the property value, default=null.

Properties can be specified using the following Property Types.

Basic Property Type

Type	Java type	Description
Boolean	java.lang.Boolean or boolean	Handles only the values *True* or *False*.
Integer	java.lang.Integer or int or java.math.BigInteger	32-bit signed Integers.
Short	java.lang.Short or short	Small 16-bit signed integers.
Long	java.lang.Long or long	Big 64-bit signed integers.
Float	java.lang.Float or float	Decimal numbers.
Double	java.lang.Double or double	Decimal numbers with high precision.
Date	java.util.Date	Any date with the precision up to milliseconds.
String	java.lang.String	Any string as alphanumeric sequence of chars.
Embedded	? extends org.gcube.informationsystem.model.embedded.Embedded	This is an Object contained inside the owner Entity and has no Header. It is reachable only by navigating the owner Entity.
Embedded list	List<? extends org.gcube.informationsystem.model.embedded.Embedded>	List of Objects contained inside the owner Entity and have no Header. They are reachable only by navigating the owner Entity.
Embedded set	Set<? extends org.gcube.informationsystem.model.embedded.Embedded>	Set (no duplicates) of Objects contained inside the owner Entity and have no Header. They are reachable only by navigating the owner Entity.

Embedded map	Map<String, ? extends org.gcube.informationsystem.model.embedded.Embedded>	Map of Objects contained inside the owner Entity and have no Header. They are reachable only by navigating the owner Entity.
Byte	java.lang.Byte or byte	Single byte. usesful to store small 8-bit signed integers.
Binary	java.lang.Byte[] or byte[]	Can contain any value as byte array.

Derived Property Type

The following are obtained using a String as real type and adding a validation regex.

Type	Java type	Description
Enum	java.lang.Enum or enum	by default it is represented using the String representation of the Enum. So that the primitive type used will be String. The enumeration is checked by setting Regexpr property. The Regular Expression is auto-generated and it will be something like ^(FIRST-ENUM-STRING_REPRESENTATIONISECOND-ENUM-STRING_REPRESENTATIONI ...ILAST_ENUM_STRING_REPRESENTATION)$ Otherwise (if indicated using an annotation), it can be represented using the Integer value of the Enum. So that the primitive type used will be Integer. The enumeration is checked using Max and Min properties.
UUID	java.util.UUID	String representation of the UUID. The check is obtained using the regular expression ^([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}){1}$
URL	java.net.URL	String representation of the URL. No check actually.
URI	java.net.URI	String representation of the URI. No check actually.

Embedded Types

ValueSchema

Name	Type	Description
value	String	
schema	URI	

AccessPolicy

Name	Type	Description
policy	Embedded ValueSchema	
note	String	

RelationProperty

Name	Type	Description
referentialIntegrity	Enum	onDeleteCascadeWhenOrphan, onDeleteCascade, onDeleteKeep. The meaning is related to the relation direction.
accessPolicy	Embedded. AccessPolicy	A policy is characterized by a name, a description, and the period ([start], [end]) when the policies apply
expiryTime	Long	The expiry date can be used to model the time until the relationship is valid, Expiry time in milliseconds. Represent the difference, measured in milliseconds, between the creation time and midnight, January 1, 1970 UTC

6.1.2. Entity

6.1.2.1. Resource

Scope: This entity is conceived to describe every "main thing" to be registered and discovered by the Information System.				
Source	Relation	Multiplicity	Target	Description
Resource	isIdentifiedBy	1..n	Facet	Any Resource has at least one Facet which in some way allow to identify the Resource per se.
Resource	consistsOf	0..n	Facet	Any Resource consist of zero or more Facets which describes the different aspects of the facet.
Resource	isRelatedTo	0..n	Resource	Any Resource can be related to any other resource.

6.1.2.2. Facet

Facets are collections of attributes conceived to capture a certain feature / aspect of the Resource they are associated with.

Every Facet has:

- A Header automatically generated to capture identification- and provenance-related aspects of the facet once it is instantiated;

- Zero or more properties. Besides the per-facet envisaged properties, clients can add new ones.

6.1.2.3. Relation

Every relation has:

- A Header
- A Relation Property
- Zero or More properties (not necessarily predefined, similarly to Facets).

isRelatedTo

Source	Relation	Multiplicity	Target	Description
Resource	isRelatedTo	0..n	Resource	A relation linking any two Resources.

consistsOf

Source	Relation	Multiplicity	Target	Description
Resource	consistsOf	1..n	Facet	A relation connecting each Resource with one of the Facet characterizing it.

isIdentifiedBy (def)

Source	Relation	Multiplicity	Target	Description
Resource	isIdentifiedBy	1..n	Facet	A relation connecting each Resource with one of the Facet which can be used to identify the Resource.

6.2.

6.3. PARTHENOS Entities Model

The PARTHENOS Entities Model has been modelled by specializing Entities defined in previous section.

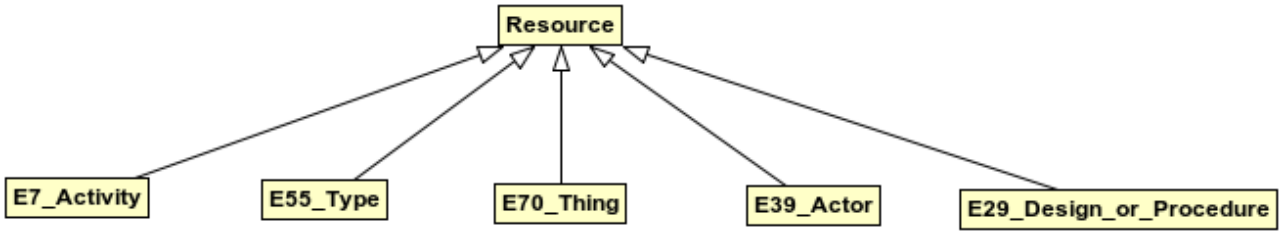

Figure 22. PARTHENOS Entities Class Diagram

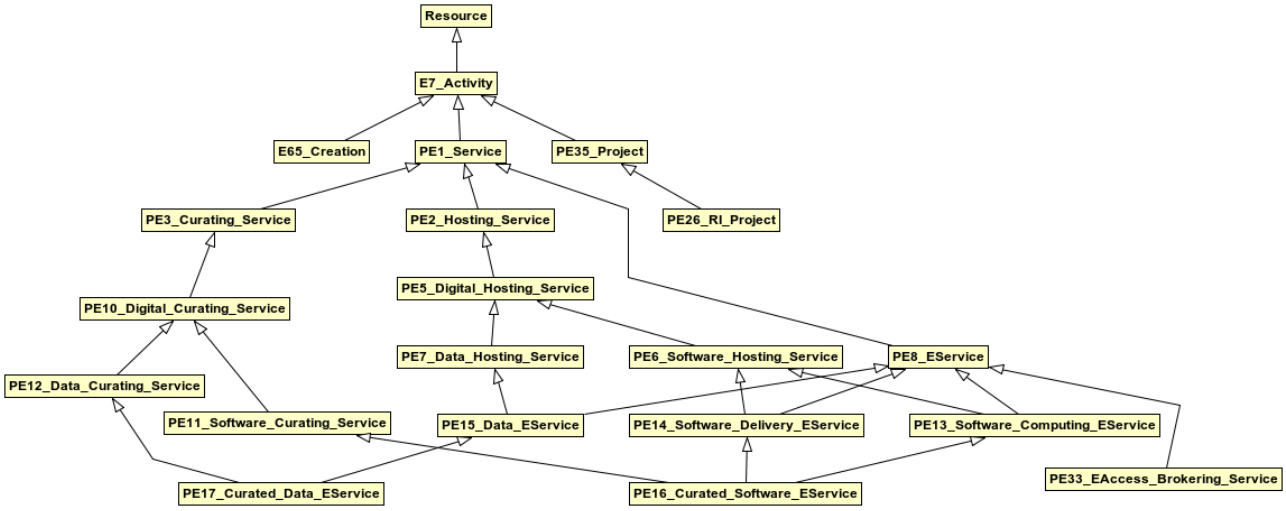

Figure 23. PARTHENOS Activity Entity Class Diagram

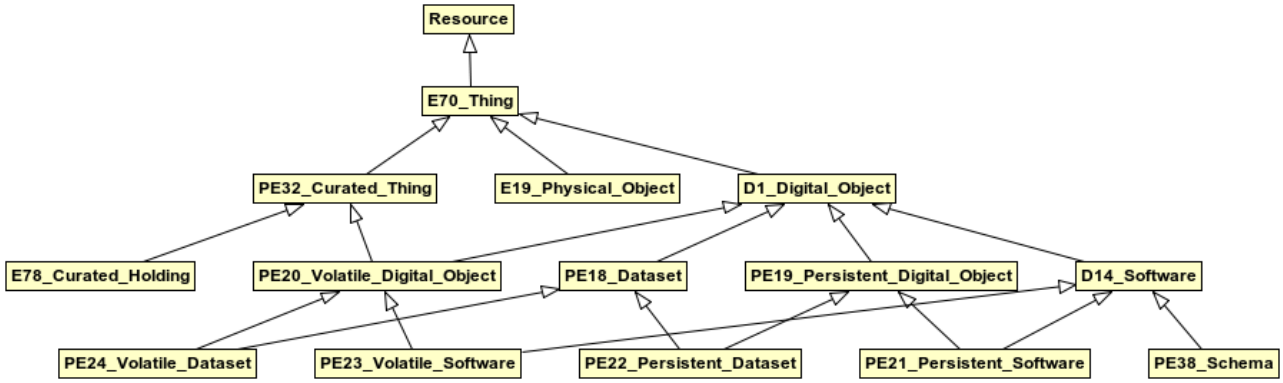

Figure 24. PARTHENOS Thing Entity Class Diagram

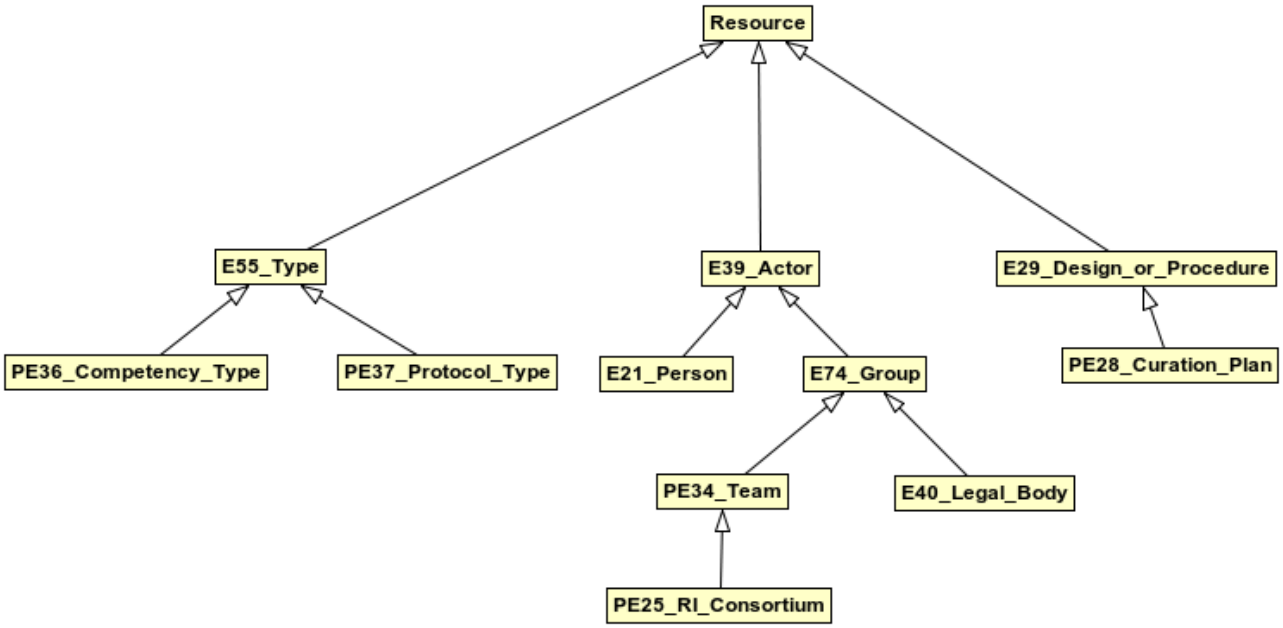

Figure 25. PARTHENOS Type, Actor and Design or Procedure Entities Class Diagram

6.3.1. Facets

The Facets reported in this Section are the ones that have been identified. Each of the tailored PARTHENOS Facet extends the Facet Entity defined in the IS Model. For each of the identified Facets the properties and their main known usages are reported.

6.3.1.1. E51_Contact_Point *extends* Facet

E51 Contact Point is modelled as Facet. It does not add any specific properties. It is needed for hierarchical consistency.

6.3.1.2. LicenseFacet *extends* Facet

Goal: This facet captures information on any licence associated with the resource to capture the policies governing its exploitation and use.				
Properties				
Name	Type	Attributes	Description	
name	String	Mandatory=true NotNull=true	The common name of the licence. E.g. EUPL 1.1, GPLv2, BSD.	
textURL	URL	Mandatory=true NotNull=true	The URL to the actual text of the licence.	
Known Usage				
Source	Relation	Multiplicity	Target	Description

PE18_Dataset	consistsOf	0..n	Licence Facet	The duration of licence - if any - can be captured by the expiry date defined in the consistsOf relation.
PE8_E_Service	consistsOf	0..n	Licence Facet	Licence for the Eservice
D14_Software	consistsOf	1..n	Licence Facet	Licence of the Software

6.3.1.3. E30_Right *extends* LicenseFacet

E30_Right is declared as specialization of LicenseFacet. It does not add any specific properties. It is needed for hierarchical consistency.

6.3.1.4. PE_Basic_Info_Facet *extends* Facet

Goal: This facet is expected to capture title and description metadata for PARTHENOS Entities. It is used as base class of P_Info Facet				
Properties				
Name	Type	Attributes	Description	
title	String	Mandatory=false	The title	
description	String	Mandatory=true NotNull=true	The Description	
Known Usage				
Source	Relation	Multiplicity	Target	Description
E70_Thing	consistsOf	1..n	P_Basic_Info_Facet	

6.3.1.5. ContactReferenceFacet

Goal: This facet captures information on the primary and authoritative contact for the resource it is associated with.			
Properties			
Name	Type	Attributes	Description.
website	URL		The main website.
address	String		A physical address.
phoneNumber	String		A phone number.
Known Usage			

Source	Relation	Multiplicity	Target	Description
E39_Actor	consistsOf	0..n	Contact Reference Facet	The primary contact information for the Actor it is attached to.
E40_Legal_Body	consistsOf	0..n	Contact Reference Facet	The primary contact information for the Legal Body it is attached to.
E21_Person	consistsOf	0..n	Contact Reference Facet	The primary contact information for the Person it is attached to.

6.3.1.6. PE_Contact_Reference_Facet extends ContactReferenceFacet

Goal: This facet is expected to capture minimal metadata for E39_Actor.				
Properties				
Name	Type	Attributes	Description	
appellation	String		The name by which the *'Resource'* is known or referred to	
description	String		A textual description	
legalAddress	String		The legal address	
eMail	String	Regex=^[a-z0-9._%+-]{1,128}@[a-z0-9.-]{1,128}	A restricted range of RFC-822 compliant email address. Please note that just domain based email address are accepted (not IP based). Please also note that new TLD are also accepted (e.g luca@google without .com which is a valid email address).	
Known Usage				
Source	Relation	Multiplicity	Target	Description
E39_Actor	consistsOf	1..n	P_Contact_Reference_Facet	Metadata for E39_Actor

6.3.1.7. PE_Info_Facet *extends* P_Basic_Info_Facet

Goal: This facet is expected to capture minimal metadata for PE1_Service.
Properties

Name	Type	Attributes	Description	
competence	ValueSchema			
availability	ValueSchema			
Known Usage				
Source	Relation	Multiplicity	Target	Description
PE1_Service	consistsOf	1..n	P_Info_Facet	Capture basic information of PE1_EService

6.3.1.8. Access Point Facet

Goal: This facet captures information on an “access point” of a resource, i.e. any web-based endpoint to programmatically interact with the resource via a known protocol.			
Properties			
Name	Type	Attributes	Description
entryName	String		A distinguishing string to be used by clients to identify the access point of interest.
endpoint	URI	Mandatory=true ReadOnly=true NotNull=true	The URI which characterizes the specific endpoint instance.
protocol	String		The high-level protocol used by the access point. The String could contains the version if needed. E.g. WMS not http which is already contained in URI.
description	String		A human oriented text accompanying the access point.
authorization	ValueSchema		Contains authorization information. E.g.: a token, username:password. By relying on schema it should be sufficient to capture also whether the content is encrypted or not. URI of the Schema to be used for a proper interpretation of the authorization value.
properties	List<ValueSchema>		This can be an arbitrarily complex element whose "structure" is defined by the associated schema. URI of the Schema to be used for a proper interpretation of the properties value.
Known Usage			

Source	Relation	Multiplicity	Target	Description
PE18_Dataset	consistsOf	0..n	Access Point Facet	Each access point captures a possible web-based method for accessing the dataset. Any embargo-related information can be captured by the access policy property of the consistsOf.
D14_Software	consistsOf	1..n	Access Point Facet	Each access point captures a possible web-based method for accessing a software manifestation. Examples are links to maven artifact on nexus, javadoc, wiki, svn, etc.

6.3.1.9. PE29_Access_Point *extends* E51_Contact_Point, AccessPointFacet

Goal: This facet is expected to capture PE29 Access Point Parthenos Entity				
Known Usage				
Source	Relation	Multiplicity	Target	Description
PE8_EService	PP28_has_designated_access_point	0..n	PE29_Access_Point	Capture metadata for PE8_EService

6.3.2. Relations

The Relation reported in this Section are the ones that have been identified. Each of the tailored PARTHENOS Relation extends the Relation Entity defined in the IS Model.

isRelatedTo PARTHENOS relations.

IsRelatedTo relation modelling CDOC-CRM and CRMdig relations.

6.3.2.1. P106_is_composed_of *extends* IsRelatedTo

Source	Relation	Multiplicity	Target
Resource	P106_is_composed_of	0..n	Resource

6.3.2.2. P125_used_object_of_type extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P125_used_object_of_type	0..n	E55_Type

6.3.2.3. P129_is_about extends IsRelatedTo

Source	Relation	Multiplicity	Target
Resource	P129_is_about	0..n	Resource

6.3.2.4. P130_shows_features_of extends IsRelatedTo

Source	Relation	Multiplicity	Target
E70_Thing	P130_shows_features_of	0..n	E70_Thing

6.3.2.5. P14_carried_out_by extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P14_carried_out_by	0..n	E39_Actor

6.3.2.6. P15_was_influenced_by extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P15_was_influenced_by	0..n	Resource

6.3.2.7. P16_used_specific_object extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P16_used_specific_object	0..n	E70_Thing

6.3.2.8. P17_was_motivated_by extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P17_was_motivated_by	0..n	Resource

6.3.2.9. P21_had_general_purpose extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P21_had_general_purpose	0..n	E55_Type

6.3.2.10. P33_used_specific_technique extends IsRelatedTo

Source	Relation	Multiplicity	Target
E7_Activity	P33_used_specific_technique	0..n	E29_Design_or_Procedure

6.3.2.11. P9_consists_of *extends* IsRelatedTo

Source	Relation	Multiplicity	Target
Resource	P9_consists_of	0..n	Resource

IsRelatedTo relation modelling tailored PARTHENOS relations.

6.3.2.12. PP1_currently_offers extends P9_consists_of

Source	Relation	Multiplicity	Target
PE26_RI_Project	PP1_currently_offers	0..n	PE1_Service

6.3.2.13. PP2_provided_by extends P14_carried_out_by

Source	Relation	Multiplicity	Target
PE1_Service	PP2_provided_by	0..n	E39_Actor

6.3.2.14. PP4_hosts_object extends P16_used_specific_object

Source	Relation	Multiplicity	Target
PE2_Hosting_Service	PP4_hosts_object	0..n	E70_Thing

6.3.2.15. PP6_hosts_digital_object extends PP4_hosts_object

Source	Relation	Multiplicity	Target
PE5_Digital_Hosting_Service	PP6_hosts_digital_object	0..n	D1_Digital_Object

6.3.2.16. PP7_hosts_software_object extends PP6_hosts_digital_object

Source	Relation	Multiplicity	Target
PE6_Software_Hosting_Service	PP7_hosts_software_object	0..n	D14_Software

6.3.2.17. PP8_hosts_dataset extends PP6_hosts_digital_object

Source	Relation	Multiplicity	Target
PE7_Data_Hosting_Service	PP8_hosts_dataset	0..n	PE18_Dataset

6.3.2.18. PP11_curates_volatile_digital_object extends PP32_curates

Source	Relation	Multiplicity	Target
PE10_Digital_Curating_Service	PP11_curates_volatile_digital_object	0..n	PE20_Volatile_Digital_Object

6.3.2.19. PP12_curates_volatile_software extends PP11_curates_volatile_digital_object

Source	Relation	Multiplicity	Target
PE11_Software_Curating_Service	PP12_curates_volatile_software	0..n	PE23_Volatile_Software

6.3.2.20. PP13_curates_volatile_dataset extends PP11_curates_volatile_digital_object

Source	Relation	Multiplicity	Target
PE12_Data_Curating_Service	PP13_curates_volatile_dataset	0..n	PE24_Volatile_Dataset

6.3.2.21. PP14_runs_on_request extends P16_used_specific_object

Source	Relation	Multiplicity	Target

PE13_Software_Computing_E_Service	PP14_runs_on_request	0..n	D14_Software

6.3.2.22. PP15_delivers_on_request extends P16_used_specific_object

Source	Relation	Multiplicity	Target
PE14_Software_Delivery_E_Service	PP15_delivers_on_request	0..n	D14_Software

6.3.2.23. PP16_has_persistent_digital_object_part extends P106_is_composed_of

Source	Relation	Multiplicity	Target
PE19_Persistent_Digital_Object	PP16_has_persistent_digital_object_part	0..n	PE19_Persistent_Digital_Object

6.3.2.24. PP17_has_snapshot extends P130_shows_features_of

Source	Relation	Multiplicity	Target
PE20_Volatile_Digital_Object	PP17_has_snapshot	0..n	PE19_Persistent_Digital_Object

6.3.2.25. PP18_has_digital_object_part extends P106_is_composed_of

Source	Relation	Multiplicity	Target
PE20_Volatile_Digital_Object	PP18_has_digital_object_part	0..n	D1_Digital_Object

6.3.2.26. PP19_has_persistent_software_part extends PP16_has_persistent_digital_object_part

Source	Relation	Multiplicity	Target
PE21_Persistent_Software	PP19_has_persistent_software_part	0..n	PE21_Persistent_Software

6.3.2.27. PP20_has_persistent_dataset_part extends PP16_has_persistent_digital_object_part

Source	Relation	Multiplicity	Target
PE22_Persistent_Dataset	PP20_has_persistent_dataset_part	0..n	PE22_Persistent_Dataset

6.3.2.28. PP21_has_software_part extends PP18_has_digital_object_part

Source	Relation	Multiplicity	Target
PE23_Volatile_Software	PP21_has_software_part	0..n	D14_Software

6.3.2.29. PP22_has_release extends PP17_has_snapshot

Source	Relation	Multiplicity	Target
PE23_Volatile_Software	PP22_has_release	0..n	PE21_Persistent_Software

6.3.2.30. PP23_has_dataset_part extends PP18_has_digital_object_part

Source	Relation	Multiplicity	Target
PE24_Volatile_Dataset	PP23_has_dataset_part	0..n	PE18_Dataset

6.3.2.31. PP24_has_dataset_snapshot extends PP17_has_snapshot

Source	Relation	Multiplicity	Target
PE24_Volatile_Dataset	PP24_has_dataset_snapshot	0..n	PE22_Persistent_Dataset

6.3.2.32. PP25_is_maintained_by extends P15_was_influenced_by

Source	Relation	Multiplicity	Target
PE26_RI_Project	PP25_is_maintained_by	0..n	PE25_RI_Consortium

6.3.2.33. PP29_uses_access_protocol extends P16_used_specific_object

Source	Relation	Multiplicity	Target
PE8_E_Service	PP29_uses_access_protocol	0..n	D14_Software

6.3.2.34. PP31_used_curation_plan extends P33_used_specific_technique

Source	Relation	Multiplicity	Target
PE3_Curating_Service	PP31_used_curation_plan	0..n	PE28_Curation_Plan

6.3.2.35. PP32_curates extends IsRelatedTo

Source	Relation	Multiplicity	Target
PE3_Curating_Service	PP32_curates	0..n	PE32_Curated_Thing

6.3.2.36. PP39_is_metadata_for extends P129_is_about

Source	Relation	Multiplicity	Target
PE22_Persistent_Dataset	PP39_is_metadata_for	0..n	D1_Digital_Object

6.3.2.37. PP40_created_successor_of extends P16_used_specific_object

Source	Relation	Multiplicity	Target
E65_Creation	PP40_created_successor_of	0..n	PE22_Persistent_Dataset

6.3.2.38. PP41_is_index_of extends IsRelatedTo

Source	Relation	Multiplicity	Target
PE24_Volatile_Dataset	PP41_is_index_of	0..n	D1_Digital_Object

6.3.2.39. PP43_supported_project_activity extends P9_consists_of

Source	Relation	Multiplicity	Target
PE35_Project	PP43_supported_project_activity	0..n	E7_Activity

6.3.2.40. PP44_has_maintaining_team extends P17_was_motivated_by

Source	Relation	Multiplicity	Target
PE35_Project	PP44_has_maintaining_team	0..n	PE34_Team

6.3.2.41. PP45_has_competence extends P21_had_general_purpose

Source	Relation	Multiplicity	Target
PE1_Service	PP45_has_competence	0..n	PE36_Competency_Type

6.3.2.42. PP46_brokers_access_to extends IsRelatedTo

Source	Relation	Multiplicity	Target
PE33_EAccess_Brokering_Service	PP46_brokers_access_to	0..n	PE8_EService

6.3.2.43. PP47_has_protocol_type extends P125_used_object_of_type

Source	Relation	Multiplicity	Target
PE8_EService	PP47_has_protocol_type	0..n	PE37_Protocol_Type

6.3.2.44. PP48_uses_protocol_parameter extends P16_used_specific_object

Source	Relation	Multiplicity	Target
PE8_EService	PP47_has_protocol_type	0..n	PE38_Schema

6.3.2.45. PP48_uses_protocol_parameter extends P16_used_specific_object

Source	Relation	Multiplicity	Target
PE8_EService	PP47_has_protocol_type	0..n	PE38_Schema

ConsistsOf

6.3.2.46. IsIdentifiedBy *extends* ConsistsOf

Source	Relation	Multiplicity	Target
Resource	isIdentifiedBy	1..n	Facet

6.3.2.47. P1_is_identified_by *extends* IsIdentifiedBy

Source	Relation	Multiplicity	Target
Resource	P1_is_identified_by	0..n	Facet

6.3.2.48. PP28_has_designated_access_point *extends* P1_is_identified_by

Source	Relation	Multiplicity	Target
PE8_E_Service	PP28_has_designated_access_point	0..n	PE29_Access_Point

6.3.3. Resources

Every Resource has:

- A Header
- One or more Facets characterizing it
- Zero or More relation with other Resources

A class can be identified as Abstract. This means that cannot be instantiated. It is expected that one of its specializations are instantiated. It is not required that an Abstract class establishes an ***IsIdentifiedBy*** relation with a Facet.

The following Entities are used from PARTHENOS model but are defined in CIDOC-CRM and CRMdig. They are the minimal entities to be declared for hierarchical consistency. Please note that not all the CIDOC-CRM and CRMdig hierarchy has been defined.

- D1_Digital_Object *extends* E70_Thing
- D14_Software *extends* D1_Digital_Object
- E19_Physical_Object *extends* E70_Thing
- E21_Person *extends* E39_Actor
- E29_Design_or_Procedure *extends* Resource
- E39_Actor *extends* Resource
- E40_Legal_Body *extends* E74_Team
- E55_Type *extends* Resource
- E65_Creation *extends* Resource
- E7_Activity *extends* Resource
- E70_Thing *extends* Resource
- E74_Group *extends* E39_Actor
- E78_Curated_Holding *extends* E70_Thing

6.3.3.1. PE1_Service *extends* E7_Activity

Source	Relation	Multiplicity	Target	Description
Facets				
PE1_Service	P1_is_identified_by	1..n	IdentifierFacet	The identifier used to indicate the service. PE1->P1->E42
PE1_Service	ConsistsOf	0..n	PE_Info_Facet	Map the following minimal metadata: Title (PE1->P1->E41); Description (PE1->P3->E62); Competence (PE1->P2->E55); Availability (PE1->P2->E55).
PE1_Service	PP42_has_declarative_time	0..n	EventFacet	Declared Begin / End of operation
PE1_Service	ConsistsOf	0..n	E30_Right	Maps the following metadata: - Conditions of use/rights - Type Conditions of Use / Rights Text
PE1_Servic	ConsistsOf	1..n	PE_Contact_	Maps the

e			Reference_Facet	Communication address metadata E.g., the contact method for this particular service (regardless of providers address) "Follow path of service provider and then add from E39: E39->p76->E51" n.b. E39 is the service provider
Relations				
PE26_RI_Project	PP1_currently_offers	0..n	PE1_Service	
PE1_Service	PP2_provided_by	1..n	E39_Actor	The actor that provides the service, e.g., for a curating service we keep the curator. N.B. the semantic path will differ based upon our leve of knowledge
PE1_Service	IsRelatedTo	0..n	E39_Actor	E.g., the contact person of the actor that provides the service "Follow path of service 'Provided by' and switch E39 for E21: E21->p76->E51"

PE1_Service	PP45_has_competence	0..n	PE36_Competency_Type	

6.3.3.2. PE2_Hosting_Service *extends* PE1_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE2_Hosting_Service	PP4_hosts_object	0..n	E70_Thing	Indicate the object hosted by the hosting service PE2->PP4->E70 If hosting service has objects, display these under hosting service, hierarchically.

6.3.3.3. PE3_Curating_Service *extends* PE1_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE3_Curating_Service	PP31_uses_curation_plan	0..n	PE28_Curation_Plan	
PE3_Curating_Service	PP32_curates	0..n	PE32_Curated_Thing	Link the curation service to the general object it curates PE3-

				>PP32->PE32 If curation service is service for some curated holding, display it.

6.3.3.4. PE5_Digital_Hosting_Service *extends* PE2_Hosting_Service

Source	Relation	Multiplicity	Target	Description
Facets				
PE5_Digital_Hosting_Service	ConsistsOf	0..n	DescriptiveMetadataFacet	It maps Preservation Activity Type metadata. Indicate the type of preservation activity undertaken on hosted digital object PE5->P9->D12->P2->E55 Snapshot, Backup, Give Copy
Relations				
PE5_Digital_Hosting_Service	PP6_hosts_digital_object	0..n	D1_Digital_Object	Indicate the digital object hosted PE5->PP6->D1 If hosting service has objects, display these under hosting service, hierarchically.

6.3.3.5. PE6_Software_Hosting_Service *extends* PE5_Digital_Hosting_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE6_Software_Hosting_Service	PP7_hosts_software_object	0..n	D14_Software	Indicate the software object hosted PE6->PP7->D14 If hosting service has objects, display these under hosting service, hierarchically.

6.3.3.6. PE7_Data_Hosting_Service *extends* PE5_Digital_Hosting_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE7_Data_Hosting_Service	PP8_hosts_dataset	0..n	PE18_Dataset	Indicate the dataset hosted PE6->PP8->PE18 If hosting service has objects, display these under hosting service, hierarchically.

6.3.3.7. PE8_E_Service *extends* PE1_Service

Source	Relation	Multiplicity	Target	Description
Facets				
PE8_E_Service	PP28_has_designated_access_point	1..n	PE29_Access_Point	It maps the following metadata: - Online Access Point - Authorization - Protocol - Protocol Parameters
PE8_E_Service	consistsOf	0..n	License Facet	Licence for the Eservice
Relations				
PE8_E_Service	PP29_uses_access_protocol	1..n	D14_Software	Links the service to the access protocol, considered as a form of software, which it invokes
PE8_EService	PP47_has_protocol_type	0..n	PE37_Protocol_Type	
PE8_EService	PP48_uses_protocol_parameter	0..n	PE38_Schema	
PE33_EAccess_Brokering_Service	PP46_brokers_access_to	0..n	PE8_EService	

6.3.3.8. PE10_Digital_Curating_Service *extends* PE3_Curating_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE10_Digital_Curating_Service	PP11_curates_volatile_digital_object	0..n	PE20_Volatile_Digital_Object	Link the curation service to the volatile digital object that it manages PE10->PP11->PE20 If curation service is service for some curated holding, display it.

6.3.3.9. PE11_Software_Curating_Service *extends* PE10_Digital_Curating_Service

Scope:				
Source	Relation	Multiplicity	Target	Description
Relations				
PE11_Software_Curating_Service	PP12_curates_volatile_software	0..n	PE23_Volatile_Software	Link the curation service to the volatile software that it manages PE11->PP12->PE23 If curation service is service for some curated holding, display it.

6.3.3.10. PE12_Data_Curating_Service *extends* PE10_Digital_Curating_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE12_Data_Curating_Service	PP13_curates_volatile_dataset	0..n	PE24_Volatile_Dataset	Link the curation service to the volatile dataset that it manages PE12->PP13->PE24 If curation service is service for some curated holding, display it.

6.3.3.11. PE13_Software_Computing_E_Service *extends* PE8_EService, PE6_Software_Hosting_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE13_Software_Computing_E_Service	PP14_runs_on_request	1..n	D14_Software	Indicate the software object the service runs on request PE13->PP14->D14

6.3.3.12. PE14_Software_Delivery_EService *extends* PE8_EService, PE6_Software_Hosting_Service

Source	Relation	Multiplicity	Target	Description
Relations				
PE14_Software_Deliver	PP15_delivers_on_request	1..n	D14_Software	Indicate the software object the service delivers on request PE14-

y_EService				>PP15->D14

6.3.3.13. PE15_Data_E_Service *extends* PE8_EService, PE7_Data_Hosting_Service

No additional minimal metadata defined.

6.3.3.14. PE16_Curated_Software_E_Service *extends* PE11_Software_Curating_Service, PE14_Software_Delivery_E_Service, PE13_Software_Computing_E_Service

No additional minimal metadata defined.

6.3.3.15. PE17_Curated_Data_E_Service *extends* PE12_Data_Curating_Service, PE15_Data_E_Service

No additional minimal metadata defined.

6.3.3.16. PE18_Dataset *extends* D1_Digital_Object

Source	Relation	Multiplicity	Target	Description
Facets				
PE18_Dataset	HasTemporalCoverage	0..n	CoverageFacet	Here we indicate the geographic scope for which the dataset has relevance. PE18->E2
PE18_Dataset	consistsOf	0..n	License Facet	The duration of license - if any - can be captured by the expiry date defined in the consistsOf relation.
PE18_Dataset	consistsOf	0..n	Access Point Facet	Each access point captures a possible web-based method for accessing the dataset. Any embargo-related information can be captured by the access policy property of the consistsOf.
Relations				
PE7_Data_Hosting_Service	PP8_hosts_dataset	1..n	PE18_Dataset	Here we indicate the data hosting service responsible for the hosting of dataset PE18->PP8i->PE7
PE24_Volatile_Dataset	PP23_has_dataset_part	0..n	PE18_Dataset	

PE18_Data set	P129_is_about	0..n	E55_Type	Here we indicate the role that the dataset can play in research PE18->P129->E55

6.3.3.17. PE19_Persistent_Digital_Object *extends* D1_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE19_Persistent_Digital_Object	PP16_has_persistent_digital_object_part	0..n	PE19_Persistent_Digital_Object	"Here we indicate the persistent data object that forms a distinct part of the overall persistent data object in question. N.B. a persistent data object can have as part any other type of persistent digital object. It cannot have a volatile data object as part." PE19->PP16->PE19
PE20_Volatile_Digital_Object	PP17_has_snapshot	0..n	PE19_Persistent_Digital_Object	If the persistent data object stands as the identifying snapshot for some volatile data object, this can be indicated here. PE19->PP17i->PE20

6.3.3.18. PE20_Volatile_Digital_Object *extends* PE32_Curated_Thing, D1_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE10_Digital_Curating_Service	PP11_curates_volatile_digital_object	1..n	PE20_Volatile_Digital_Object	Here we indicate the digital curating service responsible for the curation of this object. PE20->PP11i->PE10
PE20_Volatile_Digital_Object	PP17_has_snapshot	0..n	PE19_Persistent_Digital_Object	Here we indicate the snapshot that gives the identity to a volatile data object. In order for a volatile data object to have proper provenance it must at any time have one official snapshot that is known to the curator of the object. PE20->PP17->PE19
PE20_Volatile_Digital_Object	PP18_has_digital_object_part	0..n	D1_Digital_Object	Here we can indicate the parts of a volatile data object. A volatile data object can be made up of volatile as much as persistent data objects. If it has as component as

				volatile data object, this object in turn, in order to have proper provenance must have its own snapshot. PE20->PP18->D1

6.3.3.19. PE21_Persistent_Software *extends* D14_Software, PE19_Persistent_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE21_Persistent_Software	PP19_has_persistent_software_part	0..n	PE21_Persistent_Software	Here we link the persistent software to its component parts. PE21->PP19->PE21
PE23_Volatile_Software	PP22_has_release	0..n	PE21_Persistent_Software	Here we link to the volatile software of which this persistent software is a release. PE21->PP22i->PE23

6.3.3.20. PE22_Persistent_Dataset *extends* PE18_Dataset, PE19_Persistent_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE22_Persistent_Dataset	PP20_has_persistent_dataset_part	0..n	PE22_Persistent_Dataset	
PE24_Volatile_Dataset	PP24_has_dataset_snapshot	0..n	PE22_Persistent_Dataset	Here we indicate the volatile dataset of which this persistent dataset was or is a snapshot. PE22->PP24i->PE24
PE22_Persistent_Dataset	PP39_is_metadata_for	0..n	D1_Digital_Object	

6.3.3.21. PE23_Volatile_Software *extends* D14_Software, PE20_Volatile_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE11_Software_Curatin	PP12_curates_volatile_softwa	1..n	PE23_Volatile_Software	A link between the volatile software object and the software

g_Service	re			curation service that is responsible for its curation. PE23->PP12i->PE11
PE23_Volatile_Software	PP21_has_software_part	0..n	D14_Software	Here we link to the distinct parts of the software that can be identified whether also volatile or persistent. PE23->PP21->D14
PE23_Volatile_Software	PP22_has_release	0..n	PE21_Persistent_Software	Here we link to the official release of the volatile software. PE23->PP22->PE21

6.3.3.22. PE24_Volatile_Dataset *extends* PE18_Dataset, PE20_Volatile_Digital_Object

Source	Relation	Multiplicity	Target	Description
Relations				
PE12_Data_Curating_Service	PP13_curates_volatile_dataset	1..n	PE24_Volatile_Dataset	A link between the volatile dataset object and the data curation service that is responsible for its curation. PE24->PP13i->PE12
PE24_Volatile_Dataset	PP23_has_dataset_part	0..n	PE18_Dataset	Here we link to the parts of this volatile dataset. These parts can be persistent or volatile, dataset or software. PE24->PP23->PE18
PE24_Volatile_Dataset	PP24_has_dataset_snapshot	0..n	PE22_Persistent_Dataset	Here we link to the dataset which is the snapshot of this volatile dataset. PE24->PP24->PE22
PE24_Volatile_Dataset	PP41_is_index_of	0..n	D1_Digital_Object	

6.3.3.23. PE25_RI_Consortium *extends* E40_Legal_Body

Source	Relation	Multiplicity	Target	Description
Relations				
PE26_RI_Project	PP25_has_maintaining_RI	1..n	PE25_RI_Consortium	Here we indicate the project that the RI is responsible for maintaining. PE25->PP25->PE26

6.3.3.24. PE26_RI_Project *extends* E7_Activity

Source	Relation	Multiplicity	Target	Description

Relations				
PE26_RI_Project	PP25_has_maintaining_RI	1..n	PE25_RI_Consortium	Here we indicate the project that the RI is responsible for maintaining. PE25->PP25->PE26

6.3.3.25. PE28_Curation_Plan *extends* E29_Design_or_Procedure

Source	Relation	Multiplicity	Target	Description
Relations				
PE3_Curating_Service	PP31_uses_curation_plan	0..n	PE28_Curation_Plan	

6.3.3.26. PE32_Curated_Thing *extends* E70_Thing

Source	Relation	Multiplicity	Target	Description
Relations				
PE3_Curating_Service	PP32_curates	0..n	PE32_Curated_Thing	

6.3.3.27. PE33_EAccess_Brokering_Service extends PE8_EService

Source	Relation	Multiplicity	Target	Description
Relations				
PE33_EAccess_Brokering_Service	PP46_brokers_access_to	0..n	PE8_EService	

6.3.3.28. PE34_Team extends E74_Group

Source	Relation	Multiplicity	Target	Description
Relations				
PE35_Project	PP44_has_maintaining_team	0..n	PE34_Team	

6.3.3.29. PE35_Project extends E7_Activity

Source	Relation	Multiplicity	Target	Description
Relations				
PE35_Project	PP43_supported_project_activity	0..n	E7_Activity	
PE35_Project	PP44_has_maintaining_team	0..n	PE34_Team	

6.3.3.30. PE36_Competency_Type extends E55_Type

Source	Relation	Multiplicity	Target	Description
Relations				
PE1_Service	PP45_has_competence	0..n	PE36_Competency_Type	

6.3.3.31. PE37_Protocol_Type extends E55_Type

Source	Relation	Multiplicity	Target	Description
Relations				
PE8_EService	PP47_has_protocol_type	0..n	PE37_Protocol_Type	

6.3.3.32. PE38_Schema extends D14_Software

Source	Relation	Multiplicity	Target	Description
Relations				
PE8_EService	PP48_uses_protocol_parameter	0..n	PE38_Schema	

7. The Joint Resource Registry Architecture

The **PARTHENOS Joint Resource Registry** aims to guarantee resource discovery and integration. To do this, we identify some basic requirements, which are:

- A persistent solution.
- Search/retrieval capabilities.
- Inbound and outbound interfaces for data and metadata ingestion.
- An API for data and metadata management.

Starting from the results of the census whose entities are summarized in Table 9 and from the aforementioned requirements, the main concepts identified has been grouped into four functions:

1. Browse features: can include browse by archive, institution, deposit date, author, subjects or broad topical categories, resource type, equations/formulae, latest updates.
2. Simple Search or Advanced Search to facilitate quick access and efficient retrieval of records.
3. Export of the data in different formats.
4. The ability of managing and searching data via an API.

The PARTHENOS Joint Resource Registry is part of the PARTHENOS Cloud Infrastructure Enabling Framework [10].

The Enabling Framework is realized as a combination of services and libraries that contribute and extend the gCube System open-source project. These services promote the optimal exploitation of the resources available in the PARTHENOS Cloud Infrastructure and the integration of technology operated and maintained by external resource providers. They insulate, as much as possible, the management of the infrastructure from the data and the data management services that are hosted in or accessible through the infrastructure itself.

The motto at the heart of the management facilities is *fewer dependencies for more management* meaning that the requirements imposed on resources (even resources

operated by third-party providers) to be managed are minimal, close to zero in some cases. All the implemented solutions are prioritized in order to pursue this goal.

The Enabling Framework is composed of three main systems: Resource Management System, Information System, and Security System. These are complex ICT systems that exploit tailored persistence technologies managed via web services.

The Resource Management System supports the creation of a Virtual Research Environment and its exploitation via the registration, management, and utilization of the resources assigned to it. The resources managed by the Resource Manager are compliant with the PARTHENOS Entities Model as described in Section 6.2.

The Information System supports the registration, discovery, and access of the resources profile. All the implemented basic functions have been designed to manage resources compliant with the basic model as described in Section 6.1.

The Security System ensures the correct exploitation, auditing, and accounting of the resources under the policies defined at registration time and customized at VRE definition time. It is orthogonal to all services operating in the infrastructure and its components are deployed on all computing nodes.

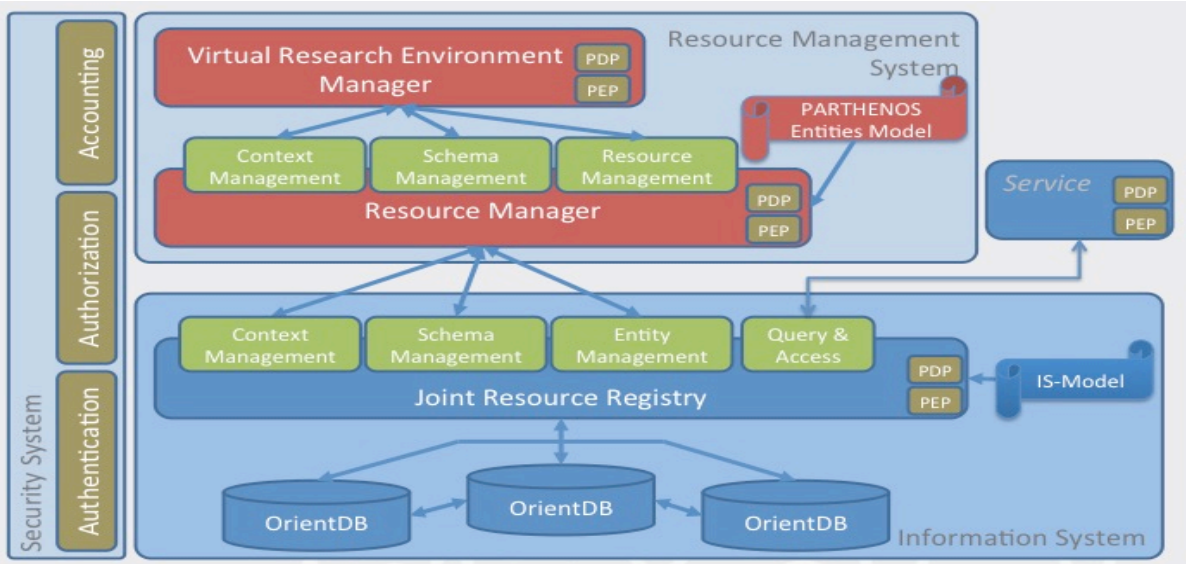

Figure 26. PARTHENOS Cloud Infrastructure Enabling Framework

The Joint Resource Registry is the core subsystem connecting producers and consumers of resources. It acts as a registry of the infrastructure by offering global and partial views of its resources and their current status and notification instruments.

The approach provided by the Resource Registry is of great support for the dynamic allocation of resources and the interoperability solutions offered by the Resource Manager system.

7.1. Key Features

Resource Publication, Access and Discovery	The Joint Resource Registry is functionally complete offering Java and WEB APIs to register new resources, to discover, and access them
Consistency with the new Resource Model	The Joint Resource Registry grants publication and access to resources compliant with the IS Model
Production level QoS - Responsiveness	Each query served in milliseconds, thousands of queries served each hour
Production level QoS - Scalability	Infrastructures with more than 100K of resources successfully powered
Production level QoS - Permanent and Uninterrupted Functioning	The Joint Resource Registry instances have been continuously up for more than one year without human intervention
Flexible deployment scenarios	The Joint Resource Registry components can be deployed in several ways, to best fit the needs of the infrastructure or a specific community

7.2. Requirements

The design of the Joint Resource Registry has been driven by requirements elicited by the design of the Joint Resource Registry Data Model, by the survey of the existing registries, and by the management needs of the PARTHENOS Cloud Infrastructure. We can divide those requirements in two main families:

- **Functional Requirements**: defines a function of a system or its component;
- **Non-Functional Requirements**: specifies criteria that can be used to judge the operation of a system, rather than specific behaviour.

Functional Requirements

- Data Definition Language (DDL) for schema definition, henceforth Schema DDL;
- Entities and Relations must be:
    - Univocally identifiable;
    - Selective/Partial updatable;
    - Validated against the Schema;
    - Guarantee Referential Integrity.

- Support Dynamic Queries: support queries built dynamically from clients rather than provided as an explicit access pattern from the system.
- Support Subscription Notification: provide the possibility for a client to subscribe for certain event and being notified when the event occurs.

Non-Functional Requirements

- High-Availability (HA): High availability is a characteristic of a system, which aims to ensure an agreed level of operational performance for a higher than normal period;
- Eventual Consistency: is a consistency model used in distributed computing to achieve high availability that informally guarantees that, if no new updates are made to a given data item, eventually all accesses to that item will return the last updated value;
- Horizontal Scalability: is the capability of a system, network, or process to handle a growing amount of work, or its potential to be enlarged in order to accommodate that growth by adding more nodes to (or remove nodes from) a system;
- Workload to be supported: 1 million facets, in average 10 facets per profile.

7.3. Joint Resource Registry Components

The design of the Joint Resource Registry supports distribution and replication wherever it is possible while abstracting clients from the deployment scenario. It exploits HAProxy for proxying requests to the deployed instances of the Joint Resource Registry web service. HAProxy is a free, very fast, and reliable solution offering high availability and load balancing for very high traffic web applications. Over the years it has become the de-facto standard open-source load balancer and it is now shipped with most mainstream Linux distributions. For these reasons, it is deployed by default in the PARTHENOS Cloud Infrastructure.

The Joint Resource Registry web service is stateless making it possible to replicate it horizontally.

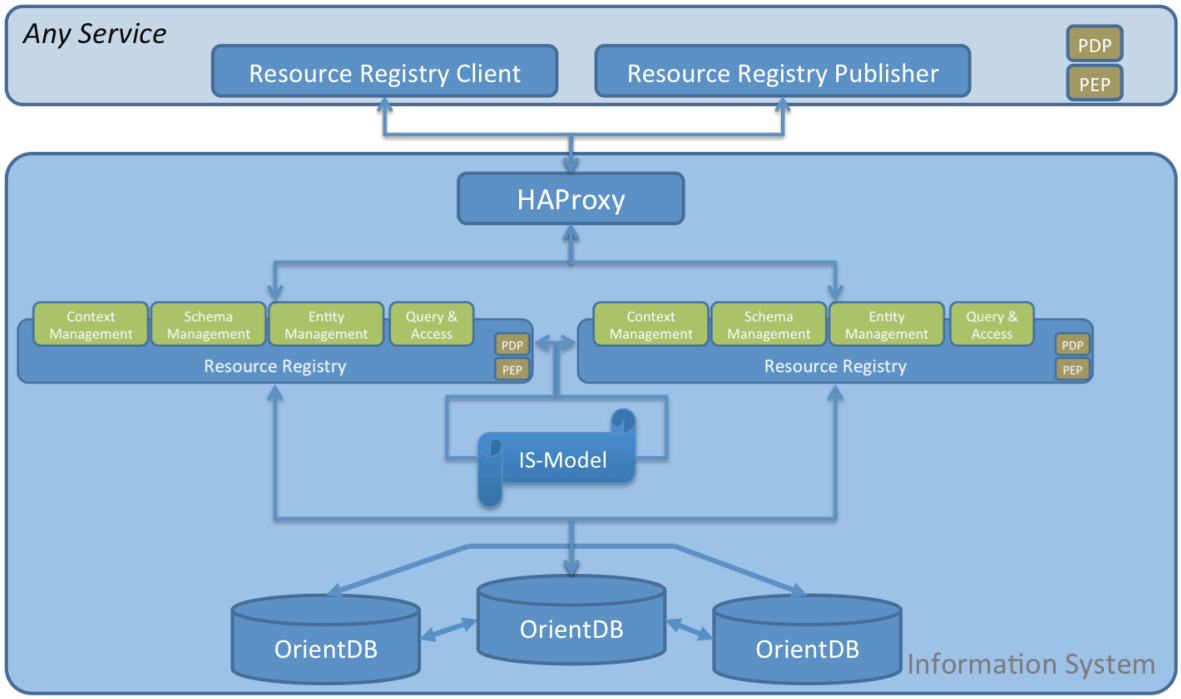

Figure 27. Joint Resource Registry Architecture

In the infrastructure are deployed many instances of the Joint Resource Registry. To load balance these instances there are several HA-Proxy instances. The client sends and receives messages to/from the Joint Resource Registry but is proxied instead by HA-Proxy. Even though HA-Proxy instances are more than one, this is hidden through the use of a Round Robin DNS (not shown in the picture for simplicity).

From the other side, the Joint Resource Registry communicates with a database cluster. OrientDB has been selected as the backend because it perfectly fits the requirements. The selection of the backend database was performed on non-functional requirements very similar with the same requirements identified for the Resource Registry. OrientDB is a Graph Database (but also Document Store and Key-Value Store). Moreover, it supports Apache TinkerPop™ standard. *Apache TinkerPop™ is a graph-computing framework for both graph databases (OLTP) and graph analytic systems (OLAP). When a data system is TinkerPop-enabled, its users are able to model their domain as a graph and analyse that graph using the Gremlin graph traversal language.*

7.4. Port types

The Joint Resource Registry web service has 4 port-types:

- **Context Management**: managing hierarchical Context. A VRE is a typical context managed by the Resource Registry;
- **Schema Management**: registering and defining Entities and Relations schema. This port-type is used by the Resource Manager to register the PARTHENOS Entities Model, defined in Section 6.2, to the Joint Resource Registry web service. This choice allows easy extension and support modification to the resource model and this is a key factor for the sustainability of the service and the Cloud infrastructure that have to last for several years;
- **Entity Management**: managing Entities and Relations instances compliant with registered schemas;
- **Query and Access**: supporting discovery and access of instances of registered entities and schema of registered types

Every port-type is exposed with a REST API.

7.4.1. Context Management

The Context Management is responsible for managing the Context belonging to the same Application Domain. Security configuration based on Authorization Framework makes this port type accessible only from the Resource Manager (see Deliverable 6.1). In other words, no other client is allowed to manage Context other than Resource Manager.

Context requirements

- No predefined hierarchies of contexts.
- Possibility to change the name of the Context with no impact for any component.
- Possibility to move a Context from a parent Context to another.

Available Methods

- **Create**: it allows creation of a new Context as child of another Context (if any).
- **Rename**: it allows renaming a Context.
- **Move**: it allows moving a context as child of another Context (different from the previous one, if any).
- **Delete**: it allows deletion of a Context.

Any action made to a Context succeeds if the following requirements are guaranteed:

- Two Contexts with same name can exist but only if they have different parents. The operations that will try to create a Context with the same name of the parent Context will fail with no effect.

- Any operation made in any Context has an effect only on the Context. In other words, there will be no effect on the associated Entity and Relations.
- When a Context is deleted, the children of that Context remain as orphans. It is the responsibility of Resource Manager to deny the Context delete or reallocate the children, depending on the policy of the infrastructure.

7.4.2. Schema Management

The Schema Management is responsible for the management of the registration of the type of entities and relations and their schema. In particular, it manages the creation of Entities sub-types, namely **Resource** and **Facet**, and Relation sub-types, namely **isRealtedTo** and **consistsOf**.

This port type is accessible through the Resource Manager.

It offers an API to:

- Create: register a new type and its own schema.
- Update: change the schema definition.
- Delete: delete a registered type.

The system offers the following consistency checks:

- On Creation and Update, it ignores all the properties the client is trying to register for a certain Resource type.
- On Update, it checks if the change can be applied to all available instances. If it fails, (e.g. the client is trying to change a property from string to integer and the instances have text string which cannot be automatically converted), it results in an error.
- On Delete, this succeeds only if there are no instances for such a type.

Please note that it is responsibility of the Resource Manager to make the update or delete feasible (e.g. through a chain of actions).

Any registered type is described by the following attributes:

- **name** (String): Type Name
- **description** (String): The description of the Type. default=null.
- **abstractType** (Boolean): Indicate if the type is abstract so that it cannot be instantiated. In other words only subtypes of this type can be instantiated. default=false.

- **superclasses** (List<String>): The list of all supertypes of this type. Multiple Inheritance is supported.
- **properties** (0..n): Resources do not have any property except the header which is managed by the system.

Any Property is described by the following attributes:

- **name**: Property Name
- **type**: The Type of the Property (e.g. String, Integer, ...).
- **description**: The description of the Property. default=null.
- **mandatory**: Indicate if the Property is mandatory. default=false.
- **readOnly**: The Property cannot change its value. default=false.
- **notNull**: Whether the property must assume a value diverse from null or not. default=false
- **max**: default=null
- **min**: default=null
- **regexpr**: A Regular Expression to validate the property value, default=null.

7.4.3. Entity Management

The Entity Management allows a user to:

- **create** a new instance of registered type in a certain Context
- **update** an instance
- **delete** an instance
- **add** an instance **to** a **Context** (different from the one it was initially created)
- **remove** an instance **from** a **Context** (if an instance is present only in such a context the instance is not deleted but it is marked as such and it becomes accessible only for management purposes).

Please note that:

- Headers are automatically managed from the Registry
- Every instance can be identified by using the UUID specified in the Header
- **add/remove to/from Context** and the **delete** operations are managed by the Joint Resource Registry to check and enforce the propagation constraint.

7.4.4. Query and Access

This Query And Access port-type provides idempotent (they don't change the actual state of the registry) methods. It provides the following methods:

- query
- read type schema
- read instances
    - Resource Instance: The result is a (or a list of) Resource(s) with all consisting Facets.
    - Facet Instance: Just the Facet instance is returned.
    - Relation Instances: The result is a (or a list of) Resource(s) containing the relations starting/arriving from/to an entity specified as parameter.

As you can see from the read methods, the design is Resource centric. This is based on the consideration that Relations and Facets do not have any reason to exist without the associated Resource.

7.4.5. Subscription Notification

The Resource Registry notifies subscribed clients of changes made to the resource (in other words it does not generate a notification for read operations).
It is compliant with the standard support for Topic as defined by the standard Java Messaging System (JMS). In JMS, the Topic specification provides the following guarantees:

- any subscribed clients receive a copy of the message in the topic
- the clients receive the messages only for the periods of they are subscribed in
- the clients have the possibility to filter the messages to be received from a topic through a query SQL like.

The Joint Resource Registry is responsible for creating and submitting the changes through a list of topic. It particular, it uses the following topic for:

- Context topic: The Resource Registry uses one single topic for all actions made on Context: Create, Rename, Move, and Delete;
- Schema Topic: The Resource Registry uses one topic all actions made on Schema: Create, Update, and Delete;

- Instance Topic: The Resource Registry uses three different topics for operation changing instances of Entities: Create, Update, and Delete.

Due to the fact that Context and Schema Management operations are sporadic, the design choice was to use a single topic of all actions for each of this port type. On the other side, the changes to instances managed from the Entity Management port-type occur frequently and so a different topic is used for each different action. This design choice simplifies the interaction with the clients and guarantees scalability and robustness.

8. Conclusions

The results of the survey on the registries existing in the infrastructures of interest of the PARTHENOS project, has been presented. The aim of the census was to identify the main entities and functionalities that form the basis on which to start the definition of a suitable data model for the PARTHENOS Joint Resource Registry. We thank all the partners who contributed to the writing of this report.

The census and the general semantic framework defined in T5.1 significantly drove the design of the Joint Resource Registry. The designed system is scalable and its initial validation proved this; it is affordable since the model can easily be extended and modified according to the evolving needs of the communities; it is sound since it satisfies all the identified requirements; and it is robust since it is compliant with standards; it is secure since it fully supports the security framework for Authorization and Policies management defined in the PARTHENOS Cloud Infrastructure.

9. References

[1] ISO/IEC 11179-1:1999 Information technology -- Specification and standardization of data elements -- Part 1: Framework for the specification and standardization of data elements, 1999.

[2] ISO/IEC 11179-1:2004. Information technology -- Metadata registries (MDR) -- Part 1: Framework, 2004

[3] Data Catalog Vocabulary (DCAT) https://www.w3.org/TR/vocab-dcat/

[4] Broeder, Daan, Menzo Windhouwer, Dieter Van Uytvanck, Twan Goosen, and Thorsten Trippel. 2012. “CMDI: A Component Metadata Infrastructure.” In Describing LRs with Metadata: Towards Flexibility and Interoperability in the Documentation of LR Workshop Programme, 1.

[5] Gavrilidou, Maria, Penny Labropoulou, Elina Desipri, Stelios Piperidis, Haris Papageorgiou, Monica Monachini, Francesca Frontini, et al. 2012. “The META-SHARE Metadata Schema for the Description of Language Resources.” In, 1090–97. http://www.lrec-conf.org/proceedings/lrec2012/index.html.

[6] Soria, Claudia, Núria Bel, Khalid Choukri, Joseph Mariani, Monica Monachini, Jan Odijk, Stelios Piperidis, Valeria Quochi, Nicoletta Calzolari, and others. 2012. “The FLaReNet Strategic Language Resource Agenda.” In LREC, 1379–86. http://lrec.elra.info/proceedings/lrec2012/pdf/777_Paper.pdf.

[7] Calzolari, Nicoletta, Riccardo Del Gratta, Gil Francopoulo, Joseph Mariani, Francesco Rubino, Irene Russo, and Claudia Soria. 2012. “The LRE Map. Harmonising Community Descriptions of Resources.” In *Proceedings of LREC 2012, Eighth International Conference on Language Resources and Evaluation*, 1084–89. Istanbul, Turkey. http://lrec.elra.info/proceedings/lrec2012/pdf/769_Paper.pdf.

[8] Calzolari, Nicoletta, Riccardo Del Gratta, Gil Francopoulo, Joseph Mariani, Francesco Rubino, Irene Russo, and Claudia Soria. 2012. “The LRE Map. Harmonising Community Descriptions of Resources.” In *Proceedings of LREC 2012, Eighth International Conference on Language Resources and Evaluation*, 1084–89. Istanbul, Turkey. http://lrec.elra.info/proceedings/lrec2012/pdf/769_Paper.pdf.

[9] McCrae, John P, Philipp Cimiano, Victor Rodrıguez Doncel, Daniel Vila-Suero, Jorge Gracia, Luca Matteis, Roberto Navigli, Andrejs Abele, Gabriela Vulcu, and Paul Buitelaar. 2015. “Reconciling Heterogeneous Descriptions of Language Resources.” ACL-IJCNLP 2015, 39.

[10] Pasquale Pagano, Leonardo Candela, Massimiliano Assante, Luca Frosini, Paolo Manghi, Alessia Bardi, Fabio Sinibaldi. “PARTHENOS Cloud Infrastructure”. Deliverable D6.1, 2016.

10. Appendix A - List of CLARIN centres

Center	City	Description
Collections de corpus oraux numeriques	Paris (FR)	Speech recordings repository
ASV Leipzig	Leipzig (DE)	
Bayerisches Archiv für Sprachsignale	München (DE)	
Berlin-Brandenburg Academy of Sciences and Humanities	Berlin (DE)	
Center of Estonian Language Resources	Tartu (EE)	
CLARIN Centre Vienna	Wien (AT)	Main Centre in CLARIN-AT, providing infrastructural services and access to CLARIN-AT language resources.
CLARIN-PL Language Technology Centre	Wrocław (PL)	Main Centre in CLARIN-PL, providing infrastructural services and access to CLARIN-PL language resources.
CLARIN.SI Language Technology Centre	Ljubljana (SL)	
CLARINO Bergen Center	Bergen (NO)	CLARINO Bergen Centre offers a repository, a corpus management and query system, a treebanking infrastructure and a CMDI editor.
CMU-TalkBank	Pittsburgh (US)	TalkBank data and tools.
Data Archiving and Networked Services	Den Haag (NL)	
Dutch-Flemish Human Language Technology Agency	Den Haag (NL)	The HLT Agency manages, maintains, distributes and supports digital Dutch language resources such as corpora, lexica and tools. The HLT Agency is the Dutch Language Union's CLARIN Centre.
Eberhard Karls Universität Tübingen	Tübingen (DE)	
Forschungzentrum Jülich	Jülich (DE)	

Gesellschaft für wissenschaftliche Datenverarbeitung Göttingen	Göttingen (DE)	
Hamburger Zentrum für Sprachkorpora	Hamburg (DE)	
Huygens ING	Den Haag	
Institut für Deutsche Sprache	Mannheim (DE)	Providing long-term storage of Germanic language resources.
Institut für Maschinelle Sprachverarbeitung	Stuttgart (DE)	
Instituut voor Nederlandse Lexicologie	Leiden (NL)	
IULA-UPF-CC-CLARIN	Barcelona (ES)	We offer harvestable machine readable metadata about resources and web services for text enrichment and quantitative text analysis services.
LINDAT/CLARIN	Praha (CZ)	
Max Planck Computing and Data Facility	Garching (DE)	
Meertens Instituut	Amsterdam (NL)	
MPI for Psycholinguistics	Nijmegen (DE)	
National Library of Norway	Oslo (NO)	Providing metadata services and access to language resources
Oxford Text Archive	Oxford (UK)	
Spanish CLARIN K-Centre	Barcelona (ES)	Distributed CLARIN K Centre consisting of CLARIN Competence Center IULA-UPF and HDLab@UPF-Department of Humanities (Barcelona), UNED – LINHD: Laboratorio de innovación en Humanidades Digitales (Madrid), and UPV – Grupo IXA (San Sebastián).
Speech & Language Data Repository	Aix-en-Provence (FR)	SLDR is a repository for oral and linguistic resources aimed at their long-term preservation and sharing
The CLARIN Centre at University of Copenhagen	København (DK)	

The Language Bank of Finland	Helsinki (FI)	Offers Finnish META-SHARE content.
Universität des Saarlandes	Saarbrücken (DE)	
Utrecht Institute of Linguistics OTS	Utrecht (NL)	