OpenArchaeo for usable semantic interoperability

Olivier Marlet
CITERES-LAT, CNRS/University of Tours
Tours, France
olivier.marlett@univ-tours.fr

Thomas Francart
Sparna
Tours, France
thomas.francart@sparna.fr

Béatrice Markhoff
LIFAT, University of Tours
Tours, France
beatrice.markhoff@univ-tours.fr

Xavier Rodier
CITERES-LAT, CNRS/University of Tours
Tours, France
xavier.rodier@univ-tours.fr

Abstract

CIDOC CRM is an ontology intended to facilitate the integration, mediation and interchange of heterogeneous cultural heritage information. The Semantic Web with its Linked Open Data cloud enables scholars and cultural institutions to publish their data in RDF, using CIDOC CRM as an interlingua that enables a semantically consistent re-interpretation of their data. Nowadays more and more projects have done the task of mapping legacy datasets to CIDOC CRM, and successful Extract-Transform-Load data-integration processes have been performed in this way. A next step is enabling people and applications to actually dynamically explore autonomous datasets using the semantic mediation offered by CIDOC CRM. This is the purpose of OpenArchaeo, a tool for querying archaeological datasets on the LOD cloud. We present its main features: the principles behind its user friendly query interface and its SPARQL Endpoint for programs, together with its overall architecture designed to be extendable and scalable, for handling transparent interconnections with evolving distributed sources while achieving good efficiency.

1. Introduction

Since 1990 the Laboratoire d’Archéologie et Territoires (CITERES-LAT) in Tours has stored all its excavations data based on stratigraphic principles (with the dual aims of data management and research) in the ArSol relational database, which is available online[1] since 2014. It belongs to the MASA Consortium[2] of the French TGIR Huma-Num[3] which assists archaeologists in digitizing and making available their excavation archives. Noting the proliferation of archaeological datasets, varying in their structures and in their formats, the objective is to make these datasets interoperable by following the precepts of the Five Stars Linked Open Data and the FAIR principles (Findable, Accessible, Interoperable, Re-usable data). In addition to the essential online availability and usability of datasets, semantic interoperability requires a shared communication layer. For this purpose the MASA consortium chose the standard ontology CIDOC CRM dedicated to the modelling of cultural heritage, which is increasingly adopted by museums such as the British Museum[4], and in European projects such as ARIADNEplus[5]. In the context of ARIADNEplus, the MASA Consortium is engaged to share its datasets and the experimental tools which have been devised for exploiting them. OpenArchaeo is one of those tools and it is intended to evolve according to the needs of users from the ARIADNEplus community.

To meet the requirements of the technical solutions we have chosen to use, we considered that the mapping of archaeological databases to an ontology such as the CIDOC CRM is a prerequisite for their semantic interoperability. Many tools are currently available to perform this operation depending on the existing database format. SPARQL addicts can for example use SPARQL-Generate[6] to export any kind of data to RDF following a given schema. For ArSol we used Ontop[7], an Ontology-Based Data Access tool designed at the University of Bozen-Bolzano which enables to define mappings with a Protégé[8] plugin [2]. Those mappings can be used either to directly querying the CRM ontology and get

[1] Archives du Sol: http://arsol.univ-tours.fr
[2] Mémoire des Archéologues et des Sites Archéologiques: https://masa.hypotheses.org
[3] French Very Large research infrastucture for Digital Humanities: https://www.huma-num.fr
[4] British Museum Sparql Endpoint: https://collection.britishmuseum.org/resource/sparql
[5] Research Infrastructure for archaeologists: https://ariadne-infrastructure.eu/
[6] https://ci.mines-stetienne.fr/sparql-generate/
[7] https://ontop.inf.unibz.it/
[8] https://protege.stanford.edu/

results from the connected relational database, or to export the relational database into an RDF graph of CIDOC CRM instances. For CITERES-LAT's XML databases we also used the Mapping Memory Manager[9], the online visual application provided by ICS-FORTH team in Heraklion to map an XML dataset to the CIDOC CRM. For instance we did it for Aerba[10], which contains the Atlas of rural settlements in ancient Beauce (France). In both cases, a generic model, part of the CIDOC CRM and its extensions has been devised, establishing the minimum elements that can be found in most archaeological datasets, even at different scales. We present this generic model in Section 2. The same generic model is currently used by MOM (Maison de l'Orient et de la Méditerranée, Lyon, France) for mapping their datasets about excavations of the Kition-Pervolia[11] site in Cyprus to the CIDOC CRM, for registering to OpenArchaeo. Interestingly, the British Museum owns artifacts found at Kition so an application could query data both from MOM and British Museum through the CRM.

While programs can be developed to use several RDF datasets, it is commonly noticed that once datasets are in RDF and associated to the CIDOC CRM entities and properties, there still is a need of an application layer for enabling people to use the semantic interoperability provided by the CRM. OpenArchaeo is a semantic mediator to be hosted by a service provider such as Huma-Num. It interconnects datasets with a common generic model dedicated to archaeological excavation data. It offers an online interface for querying registered datasets and it satisfies the following needs:

- a user-friendly query interface,
- the use of external thesauri,
- an API for web services,
- a mean for exploring data from distributed autonomous data providers,
- a solution for not duplicating all datasets by locally loading it,
- also the possibility of loading it when needed,
- the ease of adapting the system by using standard Java API.

OpenArchaeo is developed by CITERES-LAT with the company Sparna for the consortium MASA. It is implemented in Java with RDF4J and it uses the GraphDB triplestore. It can be tested in its current state[12] with two French datasets provided by the LAT (ArSol and Aerba). It is planned to be extended before the end of 2019 for browsing also textual documentation on preventive archaeological excavations which is archived by INRAP[13] and some French regional archaeological services. It is also planned to be adapted and extended within the scope of ARIADNEplus.

In the rest of this paper, in Section 2 we introduce our proposal of a generic model for archaeological excavation datasets interoperability. In Section 3 we focus on its usability by describing human and programming interfaces. In Section 4 we present its internal structure and justify why and how we conceived it, before concluding in Section 5.

2. Generic Model for Archaeological Datasets Interoperability

2.1 Motivation

For presenting how OpenArchaeo can be used we have to consider what kind of users it is intended for. As usual, we devised interfaces for applications, for administrators, and for end users. OpenArchaeo's end users are archaeologists, whether researchers or amateurs alike, i.e. people who know what an excavation is. For such people, we start from the generic model we built as a guide for mapping excavation data to the CIDOC CRM and simplify it for guiding the querying of these data. This model is a selection of CIDOC CRM, CRMsci (for scientific observations and measures), and CRMarchaeo entities and properties that we consider all together necessary and sufficient for representing the core of excavation data.

It is important to notice that our objective is to federate several autonomous datasets. OpenArchaeo platform is not intended to provide access to all specific elements of each corpus. For joining the federation, it is therefore not necessary to perform a completely detailed matching of each dataset to the CIDOC CRM ecosystem. Its purpose is to answer fairly simple queries and to provide, in the answers, the URLs to access the detailed records in each data source. If a source handles specific issues, it is by switching from the results given by OpenArchaeo to the online database of this source that the researcher is able to query these specificities more accurately. Thus, the queries concern a general level that is common to most archaeological datasets. We show in what follows that this deliberate choice enables us to provide a very intuitive querying interface. By the way, our generic model has first been motivated by our past experience in proposing the CIDOC CRM to the research community within the MASA consortium. This has led us to realize that only a framework reduced to the aspects most common to all archaeological data sets will convince everyone to contribute via the ontological level provided by the CIDOC CRM.

The OpenArchaeo model is therefore reduced to a selection of a few CIDOC CRM entities appropriate to represent archaeological entities on an operational level. The requests are made at a general level that is common to most

[9] http://139.91.183.3/3M/

[10] http://aerba.huma-num.fr/

[11] http://chypre.mom.fr/KitionSalamine/home

[12] http://openarchaeo.huma-num.fr/explorateur

[13] https://www.inrap.fr/en

archaeological datasets. This level comprises the site, its location, the person in charge of the operation, the structures, features, walls, burials, stratigraphic units and artifacts. For each of these items, attention has been paid to standard descriptions, possible dates and related documentation. Fig. 1 shows this generic template that we built over time, as we mapped new datasets to the CIDOC CRM for interoperability purposes.

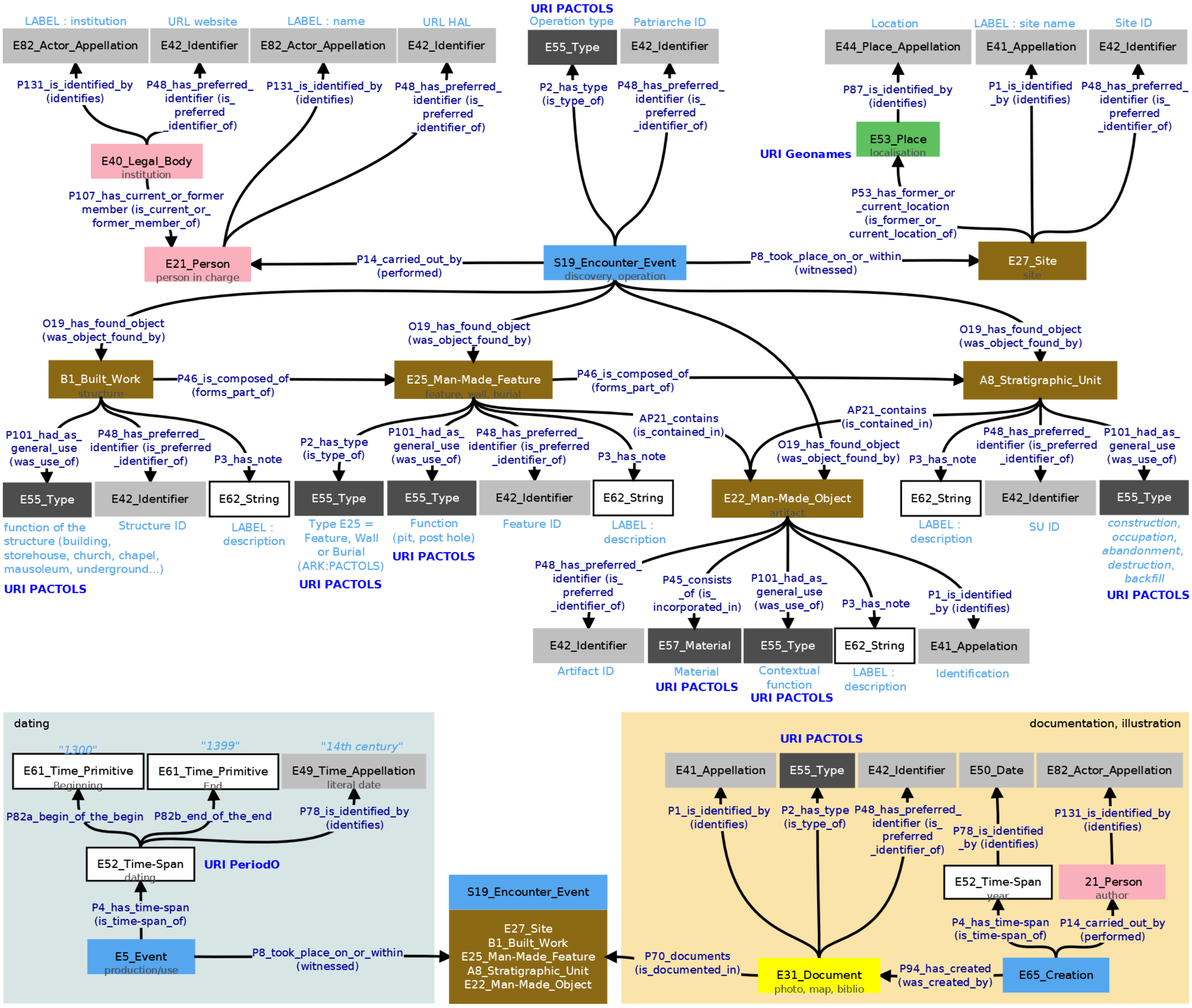

LABEL : institution
URL website
LABEL : name
URL HAL
URI PACTOLS
Operation type
Patriarche ID
Location
LABEL : site name
Site ID
E82_Actor_Appellation
E42_Identifier
E82_Actor_Appellation
E42_Identifier
E55_Type
E42_Identifier
E44_Place_Appellation
E41_Appellation
E42_Identifier
P131_is_identified_by (identifies)
P48_has_preferred_ identifier (is_ preferred _identifier_of)
P131_is_identified_by (identifies)
P48_has_preferred_ identifier (is_ preferred _identifier_of)
P2_has_type (is_type_of)
P48_has_preferred_ identifier (is_ preferred _identifier_of)
P87_is_identified_by (identifies)
P1_is_identified _by (identifies)
P48_has_preferred_ identifier (is_ preferred _identifier_of)
E40_Legal_Body
institution
P107_has_current_or_former member (is_current_or_ former_member_of)
URI Geonames
E53_Place
localisation
P53_has_former_or _current_location (is_former_or_ current_location_of)
E21_Person
person in charge
P14_carried_out_by (performed)
S19_Encounter_Event
discovery, operation
P8_took_place_on_or_within (witnessed)
E27_Site
site
O19_has_found_object (was_object_found_by)
O19_has_found_object (was_object_found_by)
O19_has_found_object (was_object_found_by)
B1_Built_Work
structure
P46_is_composed_of (forms_part_of)
E25_Man-Made_Feature
feature, wall, burial
P46_is_composed_of (forms_part_of)
A8_Stratigraphic_Unit
AP21_contains (is_contained_in)
AP21_contains (is_contained_in)
O19_has_found_object (was_object_found_by)
P101_had_as_ general_use (was_use_of)
P48_has_preferred_ identifier (is_preferred _identifier_of)
P3_has_note
P2_has_type (is_type_of)
P101_had_as_ general_use (was_use_of)
P48_has_preferred_ identifier (is_preferred _identifier_of)
P3_has_note
P3_has_note
P48_has_preferred_ identifier (is_preferred _identifier_of)
P101_had_as_ general_use (was_use_of)
E55_Type
E42_Identifier
E62_String
E55_Type
E55_Type
E42_Identifier
E62_String
E22_Man-Made_Object
artifact
E62_String
E42_Identifier
E55_Type
function of the structure (building, storehouse, church, chapel, mausoleum, underground...)
URI PACTOLS
Structure ID
LABEL : description
Type E25 = Feature, Wall or Burial (ARK:PACTOLS)
URI PACTOLS
Function (pit, post hole)
URI PACTOLS
Feature ID
LABEL : description
LABEL : description
SU ID
construction, occupation, abandonment, destruction, backfill
URI PACTOLS
P48_has_preferred_ identifier (is_ preferred _identifier_of)
P45_consists _of (is_ incorporated_in)
P101_had_as_ general_use (was_use_of)
P3_has_note
P1_is_identified _by (identifies)
E42_Identifier
E57_Material
E55_Type
E62_String
E41_Appelation
Artifact ID
Material
URI PACTOLS
Contextual function
URI PACTOLS
LABEL : description
Identification
dating
"1300"
"1399"
"14th century"
E61_Time_Primitive
Beginning
E61_Time_Primitive
End
E49_Time_Appellation
literal date
P82a_begin_of_the_begin
P82b_end_of_the_end
P78_is_identified_by (identifies)
E52_Time-Span
dating
URI PeriodO
P4_has_time-span (is_time-span_of)
E5_Event
production/use
P8_took_place_on_or_within (witnessed)
S19_Encounter_Event
E27_Site
B1_Built_Work
E25_Man-Made_Feature
A8_Stratigraphic_Unit
E22_Man-Made_Object
documentation, illustration
URI PACTOLS
E41_Appellation
E55_Type
E42_Identifier
E50_Date
E82_Actor_Appellation
P1_is_identified_by (identifies)
P2_has_type (is_type_of)
P48_has_preferred_ identifier (is_ preferred _identifier_of)
P78_is_identified _by (identifies)
P131_is_identified_by (identifies)
E52_Time-Span
year
21_Person
author
P4_has_time-span (is_time-span_of)
P14_carried_out_by (performed)
P70_documents (is_documented_in)
E31_Document
photo, map, biblio
P94_has_created (was_created_by)
E65_Creation

Fig. 1. MASA recommended abstract model for excavation datasets.

2.2 Structuring of Entities and their Relationships

The generic model used for OpenArchaeo is limited to a few entities of the CIDOC CRM with some elements from the CRMsci and CRMarchaeo extensions. These are the elementary archaeological entities that are agreed upon in the field of archaeology: Site, Operation, Structure, Feature/Wall/Burial, Stratigraphic Unit and Artifact. For each of these items, attention has been paid to standard descriptions, possible dates and related documentation.

Following the spirit of CIDOC CRM, the central entity is the event of encountering and excavating a site. This is done under the responsibility of a person. E25_Man-Made_Feature is well adapted to the notion of "Archaeological Feature", it represents the traces of human action. This concept can therefore be adapted to the facts in general - a ditch, a pit, a post hole - as well as to a wall and a burial. In archaeology, these two elements are specific, so they are rarely associated with feature but become distinct concepts for a more precise recording of their specificities. A wall is described by the type of implementation, the materials, the binder, while a burial is described by the container, the orientation and everything about the skeleton and the burial method. So to distinguish walls and burials from more generic features, the E25_Man-Made_Feature must be typed to clearly distinguish these elements. Note that an alternative would have been to create project-specific subclasses of E25_Man-Made_Feature, but it was chosen to use exclusively items already existing in the CRM.

As usual in the CIDOC CRM, each entity can be associated with an identifier, a type (preferably from a standard vocabulary) and possibly an appellation. The identifier is generally the inventory number assigned during recording and which enables to have a unique identifier in the dataset. In addition, when this data is online, it is often the same identifier that is used to access the resource through its URL.

Each of these entities is also linked to a dating module and a documentation module. For the dating module, it is a question of indicating the period of use of the element. A free date is indicated, which may be a number or a string, which corresponds to the information as it was recorded ("end of the 8th century" for example). This period, sometimes fuzzy, is then associated to a machine-readable date range, in the form of a begin and end year (775 / 800, to match with our previous example). For the documentation module, the aim is to link the documentation to these elementary archaeological elements: photographs, field drawings, recording sheets, archive documents, bibliographical references, operating reports, etc. This documentation may itself be dated and associated with an author.

2.3 Use of Standard Vocabularies

To bring data to the semantic web and ensure its interoperability, we encourage data providers of the OpenArchaeo platform to use standards-compliant vocabularies used by the archaeology community.

For all typology (type of operation, type of structure, type of stratigraphic unit, furniture materials), we use the multilingual thesaurus "Subjects" of the PACTOLS[14] managed by the Frantiq network[15] , increasingly used by French archaeologists . As part of the European ARIADNE programme, PACTOLS thesauri are aligned with the controlled vocabulary of the Getty Museum's Art & Architecture Thesaurus and consider integrating the DARIAH Infrastructure BackBones Thesaurus[16] . Since their first aim was initially dedicated to the cataloguing of bibliographical references on antiquity, it can be argued that the PACTOLS are incomplete for archaeologists. However, they are open to enhancements and a web application has been implemented within the MASA Consortium to help align unstructured or poorly structured vocabularies with a standardized thesaurus. This application is called OpenTermAlign and provides an output SKOS file of the aligned vocabulary, as well as a file to submit to the PACTOLS administrators the terms to be added to the thesaurus.

In order to ensure the interoperability of vocabularies, we favour the association of permanent identifiers (ARK or Handle) with the aligned terms. This is the case for vocabularies aligned with PACTOLS but also for the chronological periods that we align with the ARIADNE thesaurus submitted in PeriodO[17].

For localization, within a federated search platform such as OpenArchaeo, it did not seem relevant to us to localize the elements more precisely than the site level. To this end we use Geonames[18]. However, in some cases, it may be necessary to reference a site more precisely than only the centroid of the town. It may be a rural locality or a urban street. While it is possible to enrich Geonames with localities, this is not yet the case for streets or even neighbourhoods. In the case of Geonames, we don’t ask for an ARK identifier but only an URI with a number identifying the geographical entity.

3. OpenArchaeo’s interfaces

3.1 User-friendly Query Interface

We devised the intuitive visual query interface of OpenArchaeo based on the generic model shown in Fig. 1 and by following the main visual guidelines of ResearchSpace[19], to the best of our knowledge the only existing comparable visual querying tool. In our opinion it would have been counterproductive to invent something very different. ResearchSpace is developed at the British Museum on the basis of CIDOC CRM to the aim of connecting researchers, data and practices [1]. While we chose to follow the simple visual elements of ResearchSpace’s user query interface, it was hardly possible to consider developing OpenArchaeo with it for two main reasons: first we tailor OpenArcheo for archaeologists, and second the open source ResearchSpace’s code doesn’t fit our targeted scalable architecture. Once stabilized, the OpenArchaeo’s source code will be provided under the Creative Common BY-SA license as a basis to be adapted for any project of sharing archaeological data via the CIDOC CRM.

Considering the first reason, a system of icons has been set up to identify the main components of the archaeological data: the site, the operation manager, the archaeological structure, the archaeological feature, the wall, the tomb, the stratigraphic unit and the archaeological artifacts. When users wish to connect two elements (artifact and site for example), the interface automatically suggests the available relationships between these two entities, enabling users to formulate their request in a simple way without having to know either the entities and properties of CIDOC CRM, or the structure of the system. Many simple queries can be processed through the interface, Fig. 2 and Fig. 3 illustrate some of them. For instance Fig. 2 shows that the user has first chosen to search for burials, a list of relations describing burials has been

[14] https://www.frantiq.fr/fr/thesaurus

[15] Pactols administration interface : https://pactols.frantiq.fr/opentheso/

[16] https://www.backbonethesaurus.eu/

[17] http://perio.do

[18] http://www.geonames.org/

[19] https://www.researchspace.org/

proposed and she chose the “found in” relation, then she chose “site” in the list of choices. Then she can had conditions on the site, for instance for selecting those sites studied by a given person. Here several persons can be selected using a logical OR operator. Logical AND is also provided, see Fig. 3.

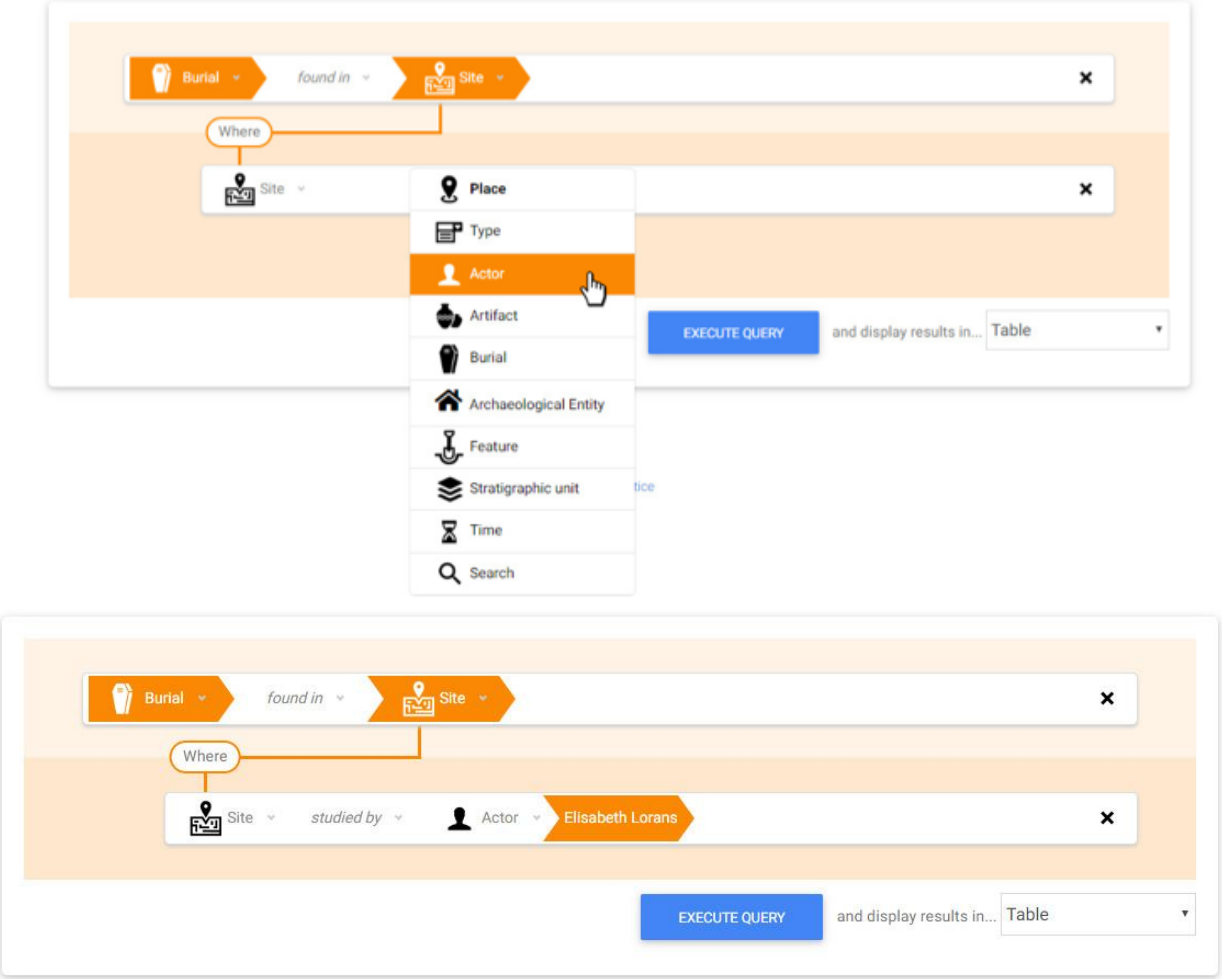


Fig. 2. Querying burials from sites whose excavation has been carried out by Elisabeth Lorans.

OpenArchaeo is able to integrate several external thesauri that are useful for querying excavation datasets. For instance, it enables users to formulate their queries with the vocabulary collected in the PACTOLS thesaurus. It is also possible to use Geonames and Periodo for spatial and temporal searches, as shown in Fig. 3.

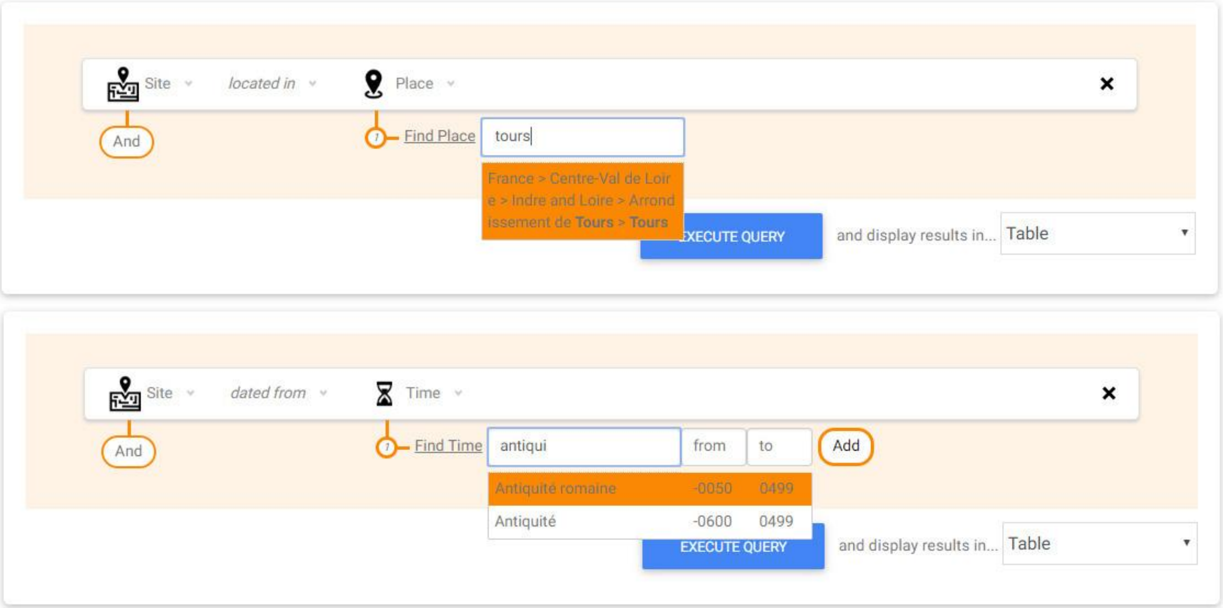
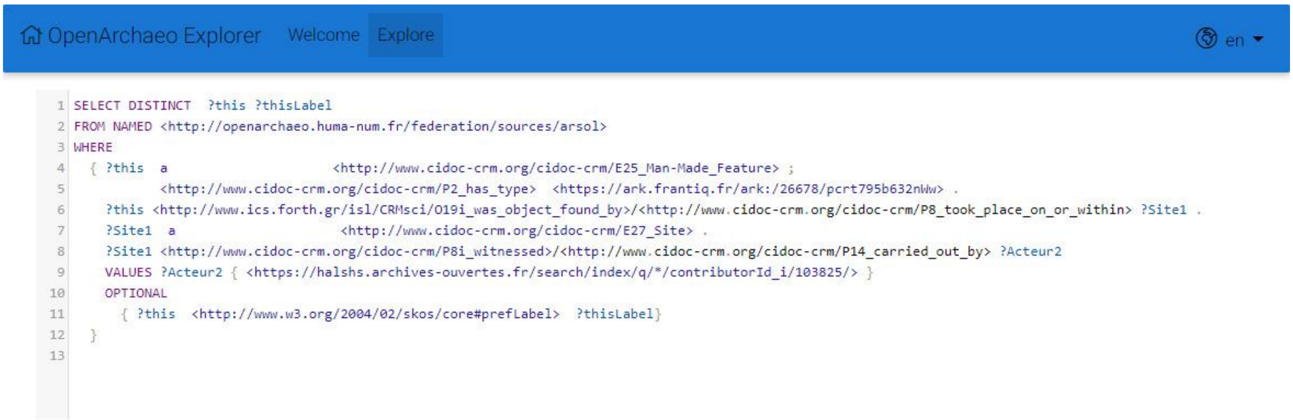

Site
located in
Place
And
Find Place
tours
France > Centre-Val de Loire > Indre and Loire > Arrondissement de Tours > Tours
EXECUTE QUERY
and display results in...
Table
Site
dated from
Time
And
Find Time
antiqui
from
to
Add
Antiquité romaine -0050 0499
Antiquité -0600 0499
EXECUTE QUERY
and display results in...
Table

Fig. 3. Use of external thesauri: Geonames and PeriodO.

As discussed in Section 4, the system is devised in such a modular way that the visual query interface can be reused in other contexts, with other high level selecting entities and relationships: this has been tested for instance on a part of DBpedia. This aspect of OpenArcheo is related to the domain of Visual Query Systems (VQS), which is currently revisited with the principles of Ontology-Based Data Access, as analyzed in a recent survey [3]. The SPARQL queries that correspond to the sentences visually built by users are automatically computed, for instance the query in Fig. 4 is the SPARQL counterpart of the screenshot in Fig. 2.



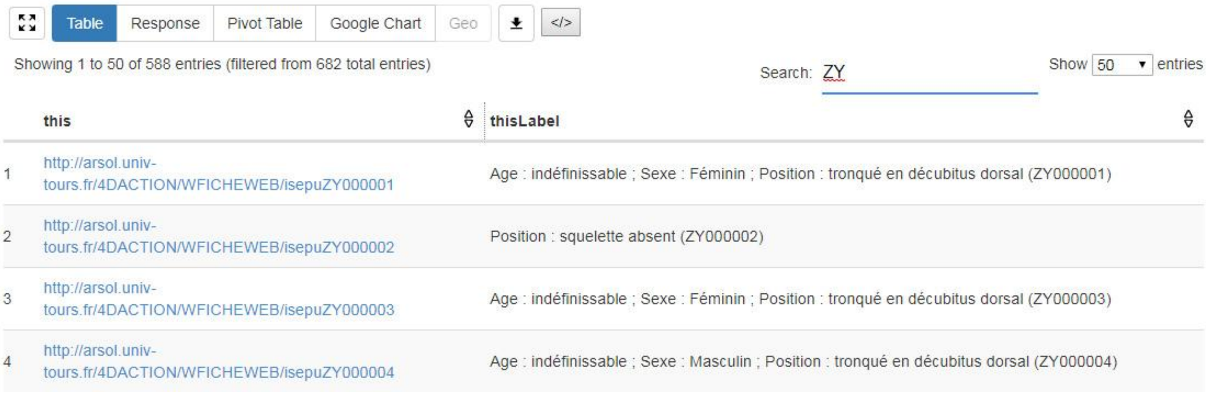
```
SELECT DISTINCT  ?this ?thisLabel
FROM NAMED <http://openarchaeo.huma-num.fr/federation/sources/arsol>
WHERE
  { ?this  a                     <http://www.cidoc-crm.org/cidoc-crm/E25_Man-Made_Feature> ;
           <http://www.cidoc-crm.org/cidoc-crm/P2_has_type>  <https://ark.frantiq.fr/ark:/26678/pcrt795b632nWw> .
    ?this <http://www.ics.forth.gr/isl/CRMsci/O19i_was_object_found_by>/<http://www.cidoc-crm.org/cidoc-crm/P8_took_place_on_or_within> ?Site1 .
    ?Site1  a                     <http://www.cidoc-crm.org/cidoc-crm/E27_Site> .
    ?Site1 <http://www.cidoc-crm.org/cidoc-crm/P8i_witnessed>/<http://www.cidoc-crm.org/cidoc-crm/P14_carried_out_by> ?Acteur2
    VALUES ?Acteur2 { <https://halshs.archives-ouvertes.fr/search/index/q/*/contributorId_i/103825/> }
    OPTIONAL
      { ?this  <http://www.w3.org/2004/02/skos/core#prefLabel>  ?thisLabel}
  }

```

Fig. 4. SPARQL query generated from the visual query in Fig. 2.

The visual query interface is mandatory to assist the beginners, the casual users and those who do not want to know the model behind or learn the SPARQL query language. But OpenArchaeo also enables experienced users to interactively explore the datasets with their own SPARQL queries, while benefiting from the various possible visualizations of the responses to their requests that we present in the next section.

3.2 Presentations of search results

OpenArchaeo is compliant with semantic web technologies and standards. In particular, its query federation is seen from the outside like a plain SPARQL service. This enables us to propose to users several kinds of visualization tools from a standard SPARQL results visualisation library, YASR[20]. The default visualisation is a table of results (Fig. 5), if the results are geolocalized they can be shown in a map, Google charts can be used for statistical views, etc.



Showing 1 to 50 of 588 entries (filtered from 682 total entries)

Search: ZY

Show 50 entries

	this	thisLabel
1	http://arsol.univ-tours.fr/4DACTION/WFICHEWEB/isepuZY000001	Age : indéfinissable ; Sexe : Féminin ; Position : tronqué en décubitus dorsal (ZY000001)
2	http://arsol.univ-tours.fr/4DACTION/WFICHEWEB/isepuZY000002	Position : squelette absent (ZY000002)
3	http://arsol.univ-tours.fr/4DACTION/WFICHEWEB/isepuZY000003	Age : indéfinissable ; Sexe : Féminin ; Position : tronqué en décubitus dorsal (ZY000003)
4	http://arsol.univ-tours.fr/4DACTION/WFICHEWEB/isepuZY000004	Age : indéfinissable ; Sexe : Masculin ; Position : tronqué en décubitus dorsal (ZY000004)

Fig. 5. Results for the query from the visual sentence query in Fig. 2.

3.3 OpenArchaeo API

If enabling end users to explore datasets integrated with OpenArchaeo is of primary importance, the Linked Open Data cloud actually exists for developers to build innovative applications based on the published linked datasets. The full potential of the CIDOC CRM integrating capabilities is not necessarily limited to the visual querying provided by OpenArchaeo. To let it be usable in other ways by everyone, we also provide means for applications to query the datasets via the integrated access point implemented by OpenArchaeo. In this way applications can use OpenArchaeo's SPARQL

[20] http://yasr.yasgui.org/

Endpoint to perform any query that could be tuned up in the user interface but also, for instance, to automatically compute statistics on the participating datasets, or verify periodically the updates in datasets, or more generally, to use Ontology-Based data Access principles to write queries that link the datasets registered in OpenArcheo with other ones in the LOD. Note however that, while a SPARQL service has been put into place, no content negotiation mechanism is provided to disseminate the metadata of each URI in the data graph.

4. OpenArchaeo's Architecture and Handling of Distributed Data Providers

OpenArchaeo's implementation is carefully conducted in order to maintain a modular structure, for easing the change of some parts without re-engineering the whole system. In particular, we can test several solutions of Federated Query Systems as we explain in Section 4.3. We will summarise how datasets are provided in the Linked Open Data cloud and how they can be consumed in an integrated way, before presenting the choices implemented in OpenArchaeo. But first of all it is important to understand its overall architecture.

4.1 General Architecture

Fig. 6 shows how the OpenArcheo architecture is decomposed into two main software components : the front-end application, including the visual query builder, entitled *OpenArchaeo Explorer*, and the federated query engine and administration interfaces, entitled *OpenArchaeo Federation*. The *OpenArchaeo Explorer* is composed of 4 main parts:

1. the first is a source selection screen that enables the user to select which underlying sources she wants to query.
2. the second is the query builder JavaScript component enabling easy creation of SPARQL queries. It should be noted that the queries constructed at this stage are not expressed in terms of the same CIDOC-CRM graph structure described in section 2, but rather using a simpler structure; for example, the "S19_Encounter_Event" entity, describing the activity of excavating a site under the responsibility of a chief archaeologist at a given date and place, is not presented to the user in the query builder, but instead a direct link from the Site to the Person, "studied by" is shown.
3. the third component is a query expansion algorithm, which translates the query structure from the one expressed by the user using the visual component into the actual CIDOC-CRM graph structure. In the previous example, the direct link "Site studied by Actor" is translated into a path of two edges : "Site witnessed an Encounter Event / Encounter Event was carried out the Actor". The translation is performed based on an ontology alignment file that defines the mapping from the "end user ontology" to the CIDOC-CRM ontology. At the end of the algorithm the final SPARQL query is sent to the *OpenArchaeo Federation* for execution.
4. upon the successful execution of the query, the results are displayed in the fourth component, responsible for displaying search results, and implemented using the YASR SPARQL Results display library.

The OpenArchaeo Federation is the foundational layer responsible for three functionalities:

A. Execution of federated SPARQL queries; the queries built in the *Explorer* on selected sources are executed using a federated query library (see below); the criterias in the query are sent to the underlying data repositories, and results are joined to create a unified result set.

B. In order to populate the lists, autocomplete and date input fields in the visual query builder, the *Federation* layer offers specific APIs on which the graphical widgets on the frontend rely; these APIs read data from centralized search indexes (implemented with Lucene), in which the labels from all federated data sources are indexed; this provides very efficient search in the frontend, without the need to go through a federated query for value selection, but requires regular update of the index when the underlying data changes.

C. Some values in the data can refer to URI identifiers of entities that are not described in the data sources federated by OpenArchaeo: for instance concepts from the PACTOLS thesaurus, or places from the Geonames database. When such external URIs are detected, the OpenArchaeo Federation fetches their associated RDF machine-readable description, and stores it locally, in order to populate the search indexes; the Federation thus maintains a local cache of the structured description of external URIs.

One key aspect of the OpenArchaeo Federation is that it is presented to the outside world as a *virtual RDF database, exposed in SPARQL*. Each federated data source is considered and exposed as an RDF Named Graph: SPARQL queries can be sent to the Federation using the "FROM" keyword to indicate which federated data source(s) should be queried. The underlying federation for query execution is built dynamically from these sources (contrary to most federated query approaches where the federation is created once, and every query runs on the same federated data source). Although the Federation does not maintain local data, it offers a SPARQL API (and web form) for developers.

The Federation System can query the registered remote data sources, but we are aware of the importance to also provide a data hosting solution for those partners who want to share their dataset but cannot implement and maintain their own triplestore and SPARQL Endpoint. To this end, OpenArchaeo offers its own RDF data-hosting solution, in which the data coming from partners can be stored. This OpenArchaeo triplestore, implemented using GraphDB, simply acts as any other federated RDF data source.

The overall architecture offers what we think are key differentiating features:

- decoupling of end user query model from the actual graph structure, thus enabling user-friendly classes and links in the visual query builder, without compromise on the expressivity of the underlying data;
- exposition of federated data sources as a virtual RDF graph that can be queried with SPARQL;
- dynamic aggregation of structured description of external URIs in a local cache, enabling the system to be agnostic on the vocabularies used (as long as the machine-readable description of URIs is accessible), and avoiding the need to manually import vocabularies in a local triplestore;
- centralized search and list indexes to perform efficient value selection in the front-end query builder, avoiding the need to perform potentially long federated queries during this step where strong user interaction is required.

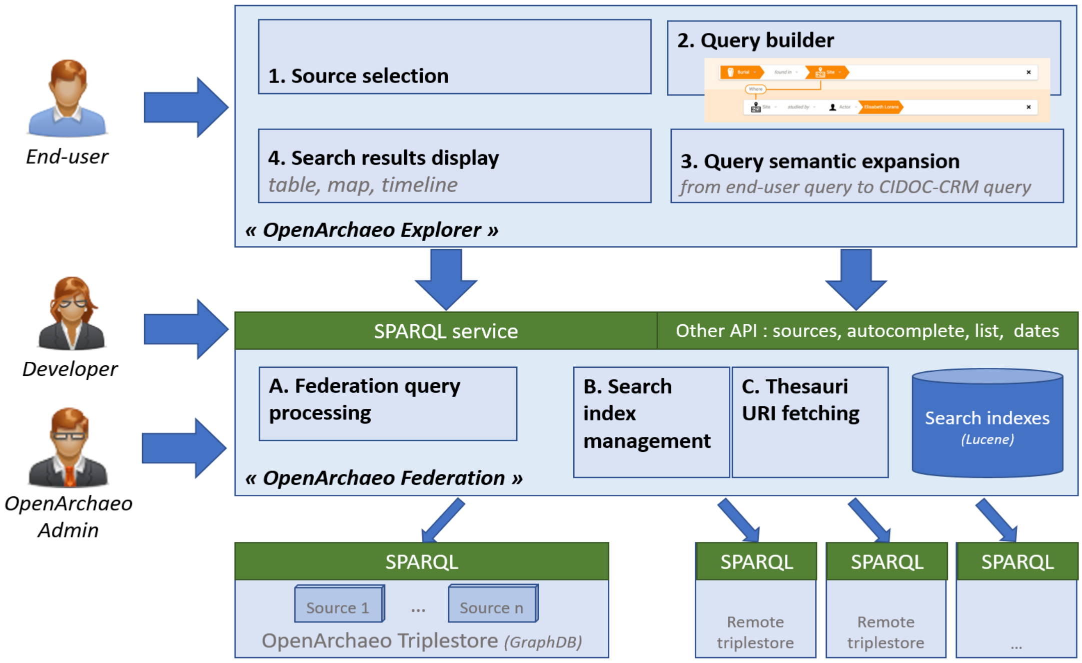


Fig. 6. Overall OpenArchaeo Architecture.

4.2 Semantic Web Data Querying and Integration

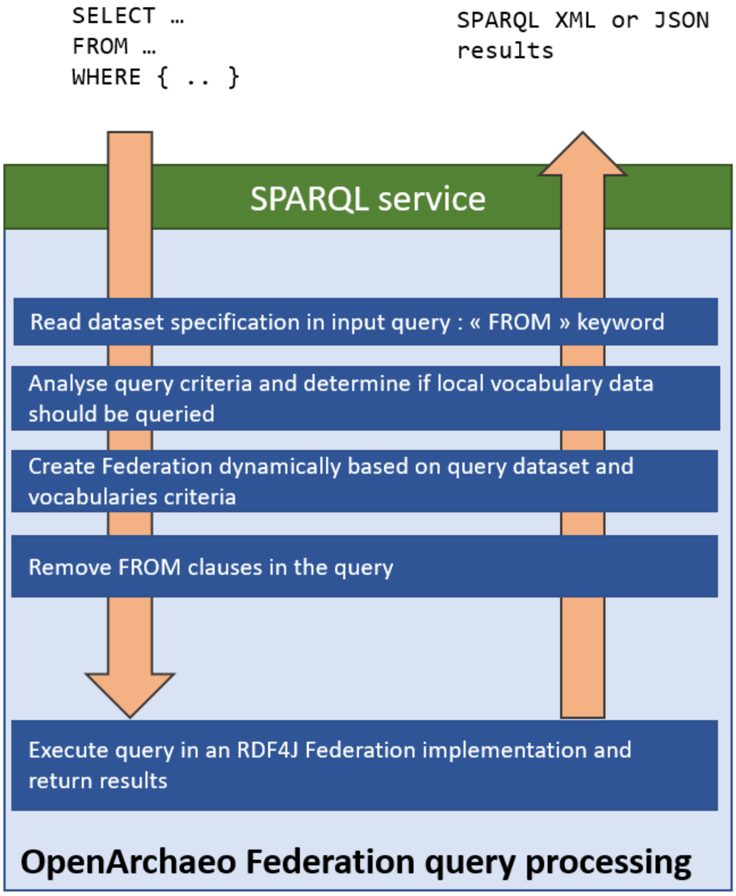
The Linked Open Data graph offer large quantities of data to be reused in other applications. However these sources use diverse data models, slightly different semantics, or different levels of details. Hence there is a need for an upper level for applications, for safely relying on a tailored integrated view of the sources. Offering a defined view and a single access point to several sources is the purpose of data integration systems, among them Ontology-Based Data Integration (OBDI) systems, and this is the very purpose [4] of the CIDOC CRM.

Based on the CIDOC CRM as a common global schema, many institutions have produced integrated datasets following OBDI processes, see for instance [5], and in most of the cases these datasets are extracted and loaded in a triplestore. This is the case for [5], also for work done at the British Museum [1]. The major advantage of such central repository infrastructures is the direct availability of locally stored data, which enables optimized query evaluation techniques. However, when harvested from autonomous sources, the queried data may become out of date. An OBDI system can also be built as a mediator, offering a single query model to perform queries on a set of autonomous data sources, as demonstrated in [6] and [7]. In the mediator solution, the integration is virtual, data is not downloaded from sources, but the user who interacts with the mediator feels like interacting with a single database. This is possible thanks to the definition of mappings between the local schemas and the global schema. The challenges of this solution are related to its query-answering process: when a query is posed in terms of the global schema presented by the mediator, the system must reformulate it in terms of a suitable set of queries posed to the sources, send each computed subquery to the involved sources, and compose the received results into a final global answer for the user. In [7] the authors use Ontop, that we mentioned in Introduction, and rely on its mappings to query local and remote datasets, while the system developed in [6] manages the query-rewriting in a rather simple way because all data sources use the same CRM-based model. Notice that this is also the case for OpenArcheo.

Semantic Web Federated Query systems [8] are an alternative solution, between central repositories and decentralized mediator systems. They propose a unique interface to perform a query on a fixed set of SPARQL Endpoints, but these endpoints have not been integrated according to a common ontology. Based on dynamically built knowledge about the source's content, the federation query engine decomposes the incoming user query into sub-queries, distributes them to data sources and constructs the final result by combining answers from each source. All datasets queried by OpenArchaeo are semantically integrated with the generic model described in Section 2, but we decided to rely on a federated query system solution for scalability.

4.3 OpenArchaeo's Distributed Query Solution

Fig. 7 represents the distributed query solution of OpenArchaeo.



```
SELECT …
FROM …
WHERE { .. }
```

SPARQL XML or JSON results
SPARQL service
Read dataset specification in input query : « FROM » keyword
Analyse query criteria and determine if local vocabulary data should be queried
Create Federation dynamically based on query dataset and vocabularies criteria
Remove FROM clauses in the query
Execute query in an RDF4J Federation implementation and return results
OpenArchaeo Federation query processing

Fig. 7. Query processing in OpenArchaeo Federation

The steps are the following:

1. A SPARQL query is sent to the SPARQL service of the Federation, including "FROM" keywords to specify which data sources should be queried.
2. The FROM clauses are parsed and read from the query to determine the list of sources to be queried; a query without FROM clauses is assumed to query all known federated sources.
3. The query criteria themselves are also analyzed to determine if the local cache of remote URIs needs to added to the federation; in particular, if the query contains criteria on Geonames latitude and longitude, the local cache of remote URIs is added to the data sources to be queried; otherwise it is not, to save some time on query execution.
4. Based on the previous step, the Federation to be queried is built dynamically.
5. The FROM clauses in the initial query are removed.
6. The SPARQL query is executed through an RDF4J Federation repository.

The execution of the SPARQL query is performed by a Federated Query System and this can be easily modified in configuration files without impacting the rest of the query processing. We have tested several solutions of Federated Query Systems:

- Simple Federation implementation provided in RDF4J[21];

[21] http://docs.rdf4j.org/programming/#_creating_a_federation

- CostFed[22];
- FedX[23].

CostFed [9] is a federation implementation that relies on local statistics of the data in each federated data source to plan query execution. It thus requires the computation of these statistics as a prerequisite to work. The two other implementations work without computation of statistics. CostFed relies on FedX [10], a more robust solution of Federated Query System. Our conclusion is that CostFed, while theoretically promising and faster for most of the queries, is still limited in the expressivity of SPARQL, in particular the VALUES keyword is not managed. In its current state it is also too unstable with query crashes, and not maintained for now. The simple Federation implementation provided in RDF4J runs too slowly, so our current choice is to work with FedX, which is production-ready, with a good maintenance from the developer, good coverage of SPARQL features, and ability to create dynamic federations at query-time.

5. Conclusion

As more and more projects have done the task of mapping legacy datasets to the CIDOC CRM for semantic interoperability purposes, there is a need for intuitive query tools for exploring those semantically interconnected datasets. In this paper we presented OpenArchaeo, a service for querying autonomous archaeological datasets in this way. It is intended to be hosted, parameterized and maintained by a community manager, as is currently done for French archaeologists by the MASA Consortium of the CNRS TGIR Huma-Num: data providers can register their datasets to OpenArchaeo, either by providing their own SPARQL endpoint or by giving their dataset to be stored in the OpenArchaeo's internal triplestore. We described its current implementation, designed to be extendable and reusable in other contexts. We drew the principles behind its intuitive visual user query interface, based on a simplified view of the generic model used to map datasets to the CRM. This query interface enables archaeologists to use external vocabularies and to visualize the output in different formats. We sketched its overall architecture for handling both local datasets in its internal triplestore and remote datasets, while achieving a good query efficiency. In the near future, we will extend OpenArchaeo for also browsing textual documentation on preventive archaeological excavations stored by INRAP and French regional archaeological services. It is also planned to be reused, tailored and extended at the European level within the scope of ARIADNEplus. In addition, a Franco-German research project on medieval written sources will exploit the same icon query interface.

Acknowledgements
This research was carried out with the support of the MASA Consortium (https://masa.hypotheses.org/) from the TGIR HUMA-NUM (https://www.huma-num.fr/), and partly supported by ARIADNEplus H2020-INFRAIA-2018-1-823914.

References

1. Oldman D., Tanase D. Reshaping the Knowledge Graph by Connecting Researchers, Data and Practices in ResearchSpace. In: Vrandečić D. et al. (eds) The Semantic Web – ISWC 2018. LNCS, vol 11137. pp. 325-340. Springer, Cham (2018).
2. Marlet, O., Curet, S., Rodier, X., Markhoff, B. Using CIDOC CRM for dynamically querying ArSol, a relational database, from the semantic web. In: Campana, S., et al. (eds) CAA 2015 Keep the revolution going. pp. 241–250. Archaeopress Archaeology, Oxford (2016).
3. Soylu, A., Giese, M., Jimenez-Ruiz, E. et al. Ontology-based End-user Visual Query Formulation: Why, What, Who, How, and Which?. Univ Access Inf Soc (2017) 16: 435. https://doi.org/10.1007/s10209-016-0465-0
4. Doerr, M., Iorizzo, D. The dream of a global knowledge network – a new approach. J. Comput. Cult. Herit. 1(1), 5:1–5:23 (2008).
5. Felicetti A., Gerth P., Meghini C., Theodoridou M. Integrating heterogeneous coin datasets in the context of archaeological research. In Proc. of the Workshop on Extending, Mapping and Focusing the CRM, co-located with 19th ICTPDL conference, p. 13–27. CEUR-WS.org (2015).
6. Niang, X., Marinica, C., Markhoff, B., Leboucher, E., Laissus, F., Malavergne, O., Bouiller, L., Darrieumerlou, C., Capderou, C. Supporting Semantic Interoperability in Conservation-Restoration domain the PARCOURS project. ACM Journal on Computing and Cultural Heritage (JOCCH) - Special Issue on Digital Infrastructure for Cultural Heritage 10, 16 (2017).
7. Calvanese, D., Liuzzo, P., Mosca, A., Remesal, J., Rezk, M., Rull, G. Ontology Based Data Integration in EPNet: production and distribution of food during the Roman Empire. Eng. Appl. Artif. Intell. 51, 212–229 (2016).
8. Ozsu, M.T.: A survey of RDF data management systems. Front. Comput. Sci. 10(3), p. 418–432 (2016).
9. Saleem M., Potocki A., Soru T., Hartig O., and Ngonga Ngomo A. C. CostFed: Cost-Based Query Optimization for SPARQL Endpoint Federation. SEMANTICS 2018, p. 163-174 (2018).
10. Schwarte, A., Haase, P., Hose, K., Schenkel, R., Schmidt, M.: FedX: optimization techniques for federated query processing on linked data. In: Aroyo et al. (eds.) ISWC 2011. LNCS, vol. 7031, pp. 601–616. Springer, Heidelberg (2011).

[22] https://github.com/dice-group/CostFed

[23] https://github.com/VeritasOS/fedx