A hands-on introduction to querying evolutionary relationships across multiple data sources using SPARQL [version 1; peer review: awaiting peer review]

Ana Claudia Sima¹-³, Christophe Dessimoz²-⁶, Kurt Stockinger¹, Monique Zahn-Zabal²,³, Tarcisio Mendes de Farias²-⁴,⁷

¹ZHAW Zurich University of Applied Sciences, Winterthur, Zurich, Switzerland
²Department of Computational Biology, University of Lausanne, Lausanne, Vaud, Switzerland
³SIB Swiss Institute of Bioinformatics, Lausanne, Vaud, Switzerland
⁴Center for Integrative Genomics, University of Lausanne, Lausanne, Vaud, Switzerland
⁵Department of Computer Science, University College London, London, UK
⁶Department of Genetics, Evolution, and Environment, University College London, London, UK
⁷Department of Ecology and Evolution, University of Lausanne, Lausanne, Vaud, Switzerland

Abstract
The increasing use of Semantic Web technologies in the life sciences, in particular the use of the Resource Description Framework (RDF) and the RDF query language SPARQL, opens the path for novel integrative analyses, combining information from multiple sources. However, analyzing evolutionary data in RDF is not trivial, due to the steep learning curve required to understand both the data models adopted by different RDF data sources, as well as the SPARQL query language. In this article, we provide a hands-on introduction to querying evolutionary data across multiple sources that publish orthology information in RDF, namely: The Orthologous MAtrix (OMA), the European Bioinformatics Institute (EBI) RDF platform, the Database of Orthologous Groups (OrthoDB) and the Microbial Genome Database (MBGD). We present four protocols in increasing order of complexity. In these protocols, we demonstrate through SPARQL queries how to retrieve pairwise orthologs, homologous groups, and hierarchical orthologous groups. Finally, we show how orthology information in different sources can be compared, through the use of federated SPARQL queries.

Keywords
Orthology, Comparative Genomics, Sequence Homology, Resource Description Framework (RDF), SPARQL
Corresponding author: Tarcisio Mendes de Farias (tarcisio.mendesdefarias@unil.ch)

Author roles: Sima AC: Conceptualization, Methodology, Software, Writing – Original Draft Preparation, Writing – Review & Editing; Dessimoz C: Funding Acquisition, Writing – Review & Editing; Stockinger K: Funding Acquisition, Writing – Review & Editing; Zahn-Zabal M: Writing – Review & Editing; Mendes de Farias T: Conceptualization, Formal Analysis, Investigation, Methodology, Project Administration, Software, Supervision, Validation, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This work was funded by the Swiss National Research Programme 75 “Big Data” (Grant 167149) and a Swiss National Science Foundation Professorship grant to CD (Grant 150654). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2019 Sima AC et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Sima AC, Dessimoz C, Stockinger K et al. A hands-on introduction to querying evolutionary relationships across multiple data sources using SPARQL [version 1; peer review: awaiting peer review] F1000Research 2019, 8:1822 (https://doi.org/10.12688/f1000research.21027.1)

First published: 29 Oct 2019, 8:1822 (https://doi.org/10.12688/f1000research.21027.1)
Introduction

Gene classification based on evolutionary history is essential for many aspects of comparative and functional genomics - reviewed in (Gabaldón & Koonin, 2013); (Glover et al., 2019). On the one hand, evolutionary relations are often described as binary relations. Two genes that share a common ancestor are defined as homologs. We can classify homologs into orthologs, which originate from a speciation event; paralogs, which originate from a gene duplication; and xenologs, which originate from horizontal gene transfer (Fitch, 1970). On the other hand, Hierarchical Orthologous Groups (HOGs) are hierarchical clusters of corresponding genes where each level in the hierarchy refers to a common ancestral gene at a taxonomic level of reference (Altenhoff et al., 2013). Identifying orthologs and HOGs is valuable in several contexts such as gene function inference, gene evolution dynamics and comparative genomics.

To query and interoperate biological databases, Semantic Web Technologies are being increasingly adopted, in particular the use of the Resource Description Framework (RDF) and SPARQL protocol and RDF query language (SPARQL Query Language for RDF, n.d.). However, despite the progress they have enabled in several fields, particularly in the life sciences (Duek et al., 2018), (Iyappan et al., 2016) (Williams et al., 2012), there are still significant challenges that limit their use for the larger scientific community. In particular, analysing evolutionary relationship data in RDF poses the following challenges:

1) complex data models - for example, while storing data in a hierarchical structure (HOGs) results in significant performance benefits for common analyses, such as computing orthologs of a specific gene in a different model organism, the hierarchy also results in requiring advanced knowledge of the SPARQL language (in particular, recursivity) in order to benefit from the RDF representation of HOGs. In this article, we present a series of hands-on examples, in increasing order of complexity, to familiarise the reader with the basic concepts needed to query evolutionary relationships in orthology databases.

2) heterogeneous data models - understanding the data model of a single orthology database might not be sufficient in general, since different databases have made different design decisions. We help overcome this challenge by depicting how the following major Orthology Databases structure their data in RDF, as well as how they can be queried using SPARQL: the Orthologous MAtrix (OMA) (Altenhoff et al., 2018), the European Bioinformatics Institute (EBI) RDF platform (Jupp, et al., 2014), the Database of Orthologous Groups (OrthoDB) (Zdobnov et al., 2017) and the Microbial Genome Database (MBGD) (Uchiyama et al., 2018).

3) overhead of integration into existing analysis pipelines. The limited rate of adoption of Semantic Web Technologies can be explained by the reluctance of bioinformaticians to change their existing workflows in order to accommodate new data formats based on the RDF framework. For example, retrieving orthology information using public SPARQL endpoints instead of the more traditional file-based data exchange or full database dumps. A SPARQL endpoint is an access point for receiving and processing SPARQL protocol requests. In this article, we show through concrete examples that integrating the results of SPARQL queries into existing analyses is a straightforward task - more specifically, we show how to transform the results into regular Pandas dataframes in Python. Furthermore, we provide an accompanying Jupyter notebook where all the examples presented in this article can be directly tested and further refined.

This article has several goals:

1) Understanding Orthology Data Models. Become familiar with how evolutionary relationships are represented in RDF across several databases. Learn about the data modelling decisions: common points as well as differences between these sources to support the choice of one or more of them for a given analysis.

2) Understanding how to query orthology data using SPARQL. To this end, we extend the introduction and examples in (Chiba et al., 2015) (de Farias et al., 2017), while also covering multiple, distributed orthology data sources.

3) Integrating external sources. Leverage connections to other external bioinformatics resources that make their data available in public SPARQL endpoints based on cross-references. In particular, learn about the role of UniProt cross-references as a bridge between different data sources in integrative analyses.

In addition, we show how to use SPARQL to make meta-analyses combining multiple orthology databases. For instance, for a given gene, which are the orthologs in a given database which are not present in another one? Finally, we show how to leverage SPARQL aggregations in order to get useful statistics about orthology data available in the sources.

Finally, learn how to leverage SPARQL results in downstream analyses by converting them to Pandas dataframes. This is illustrated through a series of hands-on exercises in the accompanying Jupyter Notebook (exercises provided in Python).

The protocols presented in this article are aimed at bioinformaticians who are already familiar with the basics of SPARQL and wish to learn how orthology data can be integrated in their research analyses programmatically, through the use of (federated) SPARQL queries.

Materials
In the following paragraphs, we briefly describe the orthology databases considered in this article.

OrthoDB (Zdobnov et al., 2017) contains orthologous genes along with evolutionary and functional annotations. It relies on HOGs to enable different orthology information resolutions with regards to more closely related species. The 2018 OrthoDB
version covers thousands of eukaryotes, prokaryotes, and viruses. OrthoDB data is available in RDF through the public SPARQL endpoint at https://sparql.orthodb.org/sparql. We note here that the timeout for the public SPARQL endpoint is limited to 100 seconds - more precisely, queries with longer estimated execution time will not be allowed to run.

MBGD (Uchiyama et al., 2018) is a comparative genomics database that contains orthology information about bacteria, archaea and unicellular eukaryotes. The 2018 MBGD version has more than six thousand genomes. The MBGD SPARQL endpoint is available online at http://mbgd.genome.ad.jp/sparql/.

OMA (Altenhoff et al., 2018) provides orthologous gene inferences covering all three domains of life: Archaea, Bacteria, and Eukarya. Although mainly focusing on orthology information, OMA also provides paralogy information (i.e. genes related by duplication). Other homology information is not explicitly available but might be manually or automatically extracted from HOGs (de Farias et al., 2017). The 2018 OMA version has 2167 species and can be queried through the SPARQL endpoint at https://sparql.omabrowser.org/lode/sparql/.

EBI is one of the largest bioinformatics resource providers in Europe (Brooksbank et al., 2014). The EBI RDF platform includes pairwise orthologous genes information from Ensembl database (Zerbino et al., 2018). The SPARQL endpoint to access the EBI data is available at https://www.ebi.ac.uk/rdf/services/sparql/. For further details, see https://www.ebi.ac.uk/rdf/documentation/ensembl/.

We group the aforementioned databases based on the orthology information type they provide as follows:

I. Hierarchical Orthologous Groups (HOGs). The three data sources that provide evolutionary relationship data in RDF to represent HOGs are OrthoDB, MBGD and OMA. Although the RDF data models of MBGD and OMA both rely on the ORTH ontology (Fernández-Breis et al., 2016), they use different ORTH versions. However, SPARQL queries running over either of the two sources can be formulated in a very similar manner. In the case of OrthoDB, data are organised according to their own internal data model, while also providing cross-references to the UniProt RDF store.

II. Homologous groups are sets of homologous genes without any hierarchical grouping (“flat”). All members are homologous to all other members, with no distinction of paralogy or orthology. However, each homologous group can still be associated with a taxonomic level, which indicates to which species clade its members belong. Example of orthology databases from which we can extract these homologous groups are OMA, OrthoDB and MBGD.

III. Pairwise orthology. Apart from the aforementioned orthologous groups, evolutionary data can also be provided in the form of pairwise orthologous genes. Among the sources that provide this type of information in RDF, we consider in this article EBI, OMA and MBGD.

Data models

In this section we provide a brief introduction to the data models of the orthology databases considered in this article, in order to facilitate the understanding of the SPARQL queries presented in the Protocol Section.

Figure 1 illustrates a few of the members of a HOG, the main data structure in MBGD. In particular, this MBGD cluster has the identifier 28799. Members of an MBGD orthologous cluster can be either genes, domains or other clusters. These nested orthologous clusters are built at specific taxonomic levels in the hierarchy. For example, the cluster highlighted in blue in Figure 1 was built at taxonomic level 32, Myxococcus. The hierarchy needs to be traversed in order to reach genes, such as mxa:PL1911 that is highlighted in red in Figure 1, or domains (sub-gene level) which belong to an orthologous cluster at a given taxonomic level.

The RDF model is more suitable for representing such hierarchical data structures than the relational data model (Sima et al., 2019) given that RDF is a graph data model. Moreover, querying orthology RDF data can benefit from SPARQL 1.1 recursive graph patterns such as property paths1. The main construct in SPARQL required to retrieve the orthologous genes of a gene of interest X will then be the following recursive pattern:

?hog_cluster a orth:OrthologsCluster.

?hog_cluster orth:hasHomologous* ?gene_X.

?hog_cluster orth:hasHomologous* ?orthologous_gene_Y.

For example, we can replace ?gene_X with the URI of the human Hemoglobin Subunit Beta (HBB) gene, namely: <http://mbgd.genome.ad.jp/rdf/resource/gene/hsa:HSA_4504349>, which would enable retrieving all orthologs of human HBB through the ?orthologous_gene_Y variable. The asterisk (*) following the “orth:hasHomologous” property indicates that this property should be matched recursively.

A graphical abstraction of the RDF data structure in MBGD is given in Figure 2. SPARQL queries can be formulated by following the direction and labels of arrows in order to formulate triple patterns. For example, to retrieve all genes (i.e. instances of the orth:Gene class) of a given HOG, we can follow the graph structure from root to leaf members by performing the query in the following code fragment. In other

---

1. https://www.w3.org/TR/sparql11-query/#propertypaths
A fragment of the hierarchical orthologous cluster no. 28799 in MBGD. A cluster can consist of genes, domains (sub-genes) or further nested orthologous clusters. Multiple levels of the hierarchy may need to be traversed recursively in order to reach a given orthologous gene. For example, the gene mxa:PL1911 (highlighted in red) can be reached through the member orthologous cluster 2018-01_tax32_8537 (shown in blue). This can be achieved in SPARQL through a recursive graph pattern, using the hasHomologous* property path. A graphical abstraction of the RDF representation is provided in Figure 2. This SPARQL query will retrieve all the genes in the MBGD Hierarchical Orthologous Group represented with the identifier 28799.

![OrthologsCluster](http://mbgd.genome.ad.jp/rdf/resource/cluster/2018-01_default_28799)

```sparql
PREFIX orth: <http://purl.org/net/orth#>

SELECT ?hog_cluster ?gene1 WHERE {
VALUES ?hog_cluster {cluster-id:2018-01_default_28799}
?hog_cluster a orth:OrthologsCluster.
?gene1 a orth:Gene. }
```

Figure 2. Directed-graph abstraction of a portion of the MBGD RDF graph related to hierarchical orthologous groups. Edges are RDF properties and nodes are either classes or variables. The terms preceded by a question mark (e.g. ?gene1) represent variables assigned with either zero or more literals or URIs. Dashed edges illustrate the orth:hasHomologous property that can be stated zero or more times, recursively. URI prefixes were omitted. MBGD is gene-centric and taxonomic ranges where HOGs are built are not directly available in RDF, in some cases these can be extracted from the cluster URI (e.g. http://mbgd.genome.ad.jp/rdf/resource/cluster/2018-01_tax32_8537 corresponds to taxonomic identifier 32, Myxococcus). By contrast, the taxonomic information per gene entry is richer in MBGD than in OMA, including explicit Superkingdom and Phylum information. Example SPARQL queries based on this graph abstraction are provided in the Protocols, as well as in the accompanying Jupyter Notebook. The pairwise orthology information is not directly available (e.g. through an RDF property), but can be extracted from the Orthologs Cluster (to highlight this, the “isPairwiseOrthologous” is shown in green with a dotted arrow).
Similarly, the HOG structure in OMA is abstracted in Figure 3. Both figures can be used as a guide in formulating SPARQL queries, by following the directions of the arrows in order to formulate triple patterns. Since both the MBGD and the OMA models rely on the ORTH ontology (Fernández-Breis et al., 2016), the two graph structures are very similar and therefore SPARQL queries can be formulated with only minor differences for both data sources.

Figure 4 illustrates the data model of the portion of the EBI RDF graph describing orthology information. In contrast to OMA and MBGD, EBI only provides pairwise orthologous genes.

Figure 5 illustrates the structure of Orthologous Groups in the OrthoDB RDF. Here, genes are direct members of OrthoGroups built at a given taxonomic level (Clade), e.g. Cyanobacteria. We mention that OrthoDB provides richer information in RDF, including sequence length, number of exons for gene members, as well as evolutionary rates, functional category and others for orthologous groups (for more details see Extended data).

Protocols

In this section, we provide four protocols to (i) retrieve pairwise orthologs through SPARQL queries from EBI, OMA, MBGD, as well as (ii) homologous groups from OMA, MBGD and OrthoDB (iii) restrict the search to a given taxonomic level (iv) perform meta-analyses across multiple data sources providing orthology information, as well as aggregations using the entire data available in a given source. All protocols presented below are included in the accompanying Jupyter notebook.

For the sake of simplicity, genes are identified with either their Ensembl identifiers or their cross-reference to the

---

**Figure 3.** Directed-graph abstraction of a portion of the OMA RDF graph related to hierarchical orthologous groups. Edges are RDF properties and nodes are either classes or variables. The terms preceded by a question mark (e.g. ?protein1) represent variables. Dashed edges illustrate the orth:hasHomologousMember property that can be stated zero or more times, recursively. OMA is protein-centric, however the corresponding genes that encode the proteins are also available in RDF through the “is encoded by” property (a cross-reference to Ensembl identifiers is also provided). Furthermore, the taxonomic ranges where HOGs were built are asserted through the “hasTaxonomicRange” property. The pairwise orthology information is not directly available (e.g. through an RDF property), but can be extracted from the Orthologs Cluster (to highlight this, the “isPairwiseOrthologous” is shown in green with a dotted arrow). Note: URI prefixes were omitted.
Figure 4. Directed-graph abstraction of a portion of the EBI RDF graph related to pairwise orthologous genes. Edges are RDF properties and nodes are either classes or variables. The terms preceded by a question mark (i.e. ?) represent variables assigned with either zero or more literals or URIs. As opposed to the RDF representations in OMA and MBGD, here the pairwise orthology is explicitly asserted through the “is orthologous to” property (more precisely, http://semanticscience.org/resource/SIO_000558). However, there is no information available regarding orthologous clusters. Moreover, the Gene class here is in fact the OBO (not ORTH) class, i.e. http://purl.obolibrary.org/obo/SO_0000704. Instances of these genes can be specified either through their cross-reference to UniProt (the http://rdf.ebi.ac.uk/terms/ensembl/DEPENDENT property) or directly through their ENSEMBL identifier, by fixing the value of ?gene to the concatenation of http://rdf.ebi.ac.uk/resource/ensembl/ and the corresponding Ensembl identifier. Finally, the taxonomic identifiers are provided via instances of the BioSource class, http://www.biopax.org/release/biopax-level3.owl#BioSource.

Figure 5. Directed-graph abstraction of a portion of the OrthoDB RDF graph related to orthologous groups. Edges are RDF properties and nodes are either classes or variables. The terms preceded by a question mark (i.e. ?) represent variables assigned with either zero or more literals or URIs. Genes are direct members of OrthoGroups built at a given taxonomic level (Clade), e.g. Cyanobacteria, available through the ogBuiltAt property. The cross-references to UniProt (as well as Ensembl and Entrez) are available through a 2-triple pattern (for examples see Protocols section).
UniProt accession number. In this article, we assume the reader already knows the UniProt primary accession number of the searched gene. In general, this number can be found by searching for the corresponding gene name in the UniProt webpage, for example, “HBB” (i.e. “hemoglobin subunit beta”). The UniProt protein identifier in RDF is a Uniform Resource Identifier (URI) composed of the UniProt accession number (e.g. P68871) appended to the UniProt namespace prefix: http://purl.uniprot.org/uniprot/. For instance, in the case of the human HBB gene, the URI identifier is http://purl.uniprot.org/uniprot/P68871.

Protocol 1: Retrieve pairwise orthologs (EBI, OMA, MBGD)

In this protocol we illustrate the basic task of retrieving the pairwise orthologs of a given gene, for example the HBB (Hemoglobin subunit beta) human gene. This is illustrated on the three orthology databases that provide pairwise orthology information in RDF: EBI, OMA and MBGD. The corresponding SPARQL queries to retrieve the pairwise orthologs can be formulated as shown below. We note that the resulting orthologs are also provided using their “clickable” cross-reference link to UniProt. This can directly be used to find out more information about the resulting genes (e.g. name, location, expression) and has the added advantage that results originating from different orthology databases can then be compared against each other.

a) Retrieving EBI pairwise orthologs

The following code fragment depicts a SPARQL query to retrieve pairwise orthologs of the human HBB gene from Ensembl dataset at the EBI RDF platform. To execute this query, copy and paste the it into the Web interface of the EBI SPARQL endpoint at https://sparql.ensembl.org.

```sparql
PREFIX ensembl: <http://rdf.ebi.ac.uk/resource/ensembl/>
PREFIX ensemblterms: <http://rdf.ebi.ac.uk/terms/ensembl/>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX obo: <http://purl.obolibrary.org/obo/>

SELECT DISTINCT ?gene ?ortholog_uniprot_uri {
  FILTER(?taxon != ?ortholog_taxon)
  FILTER(?gene_uniprot_uri != ?ortholog_uniprot_uri)
  FILTER(STRSTARTS(STR(?ortholog_uniprot_uri),"http://purl.uniprot.org/uniprot/"))
}
```

The HBB gene is represented with the UniProt URI http://purl.uniprot.org/uniprot/P68871. To retrieve the orthologs of other genes, we can replace this URI with one that corresponds to another gene such as human INS (i.e. http://purl.uniprot.org/uniprot/P01308). We can also provide a set of URIs enclosed with parentheses such as follows: VALUES(?gene_uniprot_uri) {
(http://purl.uniprot.org/uniprot/P68871)
(http://purl.uniprot.org/uniprot/P01308)
}. The sio:SI0_000558 is the « is orthologous to » property, while the obo:RO_0002162 represents the « in taxon » property (for a graphical abstraction, see Figure 4).

We note here that not all EBI gene entries have an assigned cross-reference to UniProt. For example, “ENSG00000139618” identifies an Ensembl gene for which the UniProt cross-reference is missing from the EBI RDF platform. In this case, the previous SPARQL query can be adapted, by assigning in the VALUES statement of the query, the ?gene variable to the corresponding Ensembl identifier, as depicted in the following code fragment:

```sparql
PREFIX ensembl: <http://rdf.ebi.ac.uk/resource/ensembl/>
PREFIX ensemblterms: <http://rdf.ebi.ac.uk/terms/ensembl/>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX obo: <http://purl.obolibrary.org/obo/>

VALUES(?gene) {
  (ensembl:ENSG00000139618)
  (ensembl:ENSG00000254647)
}

SELECT DISTINCT ?gene ?ortholog_uniprot_uri {
  FILTER(?taxon != ?ortholog_taxon)
  FILTER(?gene != ?ortholog)
  FILTER(STRSTARTS(STR(?ortholog_uniprot_uri),"http://purl.uniprot.org/uniprot/"))
}
```

This code fragment illustrates the SPARQL query to retrieve orthologs for the human BRCA2 gene from the Ensembl dataset. The BRCA2 gene is represented with the UniProt URI ensemble: ENSG00000139618 where ensemble is a prefix that replaces http://rdf.ebi.ac.uk/resource/ensembl/. To retrieve the orthologs of other genes, we can replace ensemble:ENSG00000139618 with a URI that corresponds to another gene such as human INS (i.e. ensemble:ENSG00000254647). We can also provide a set of URIs enclosed with parentheses such as follows: VALUES(?gene) {
(ensemble:ENSG00000139618)
(ensemble:ENSG00000254647)
}.

b) Retrieving OMA pairwise orthologs

The following code fragment shows a SPARQL query to retrieve pairwise orthologs of the human HBB gene which are derived from the HOGs in the OMA database. To execute the query, copy and paste it in the OMA SPARQL endpoint webpage: https://sparql.omabrowser.org/lode/sparql.

```sparql
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX ensembl: <http://rdf.ebi.ac.uk/resource/ensembl/>
PREFIX ensemblterms: <http://rdf.ebi.ac.uk/terms/ensembl/>

SELECT DISTINCT ?gene ?ortholog_uniprot_uri {
  FILTER(?taxon != ?ortholog_taxon)
  FILTER(STRSTARTS(STR(?ortholog_uniprot_uri),"http://purl.uniprot.org/uniprot/"))
}
```
The HBB gene is represented in this code fragment with the UniProt URI http://purl.uniprot.org/uniprot/P68871. More precisely, in OMA the lscr:xrefUniprot represents the cross reference to UniProt.

c) Retrieving MBGD pairwise orthologs
In a similar manner to the previous code fragment, the following depicts a SPARQL query to retrieve pairwise orthologs of the human HBB gene which are derived from the HOGs in the MBGD database. To execute the query, copy and paste it in the MBGD SPARQL endpoint webpage: http://mbgd.genome.ad.jp/sparql/.

This code fragment illustrates the SPARQL query to retrieve homologous groups (i.e. clusters) that contains the human HBB gene in the OMA database. The HBB gene is represented with its related UniProt entry (i.e. the UniProt URI http://purl.uniprot.org/uniprot/P68871). To retrieve the clusters that have other genes, we can replace this URI with one that corresponds to another gene such as human INS (i.e. http://purl.uniprot.org/uniprot/P01308). We can also provide a set of URIs enclosed with parentheses such as follows:

VALUES(?protein1_uniprot_URI) {(<http://purl.uniprot.org/uniprot/P68871>) (<http://purl.uniprot.org/uniprot/P01308>)}. Similarly, we can change the taxonomic level of reference as follows: VALUES(?tax_name) {("Hominoidea")}.

b) MBGD Homologous Groups derived from the HOGs
The HOGs in MBGD do not provide explicit taxonomic levels at the root level of a HOG. However, the taxon NCBI identifiers of subHOGs (i.e. sublevels) can be extracted in some cases from the cluster URI. Since this requires more advanced knowledge of SPARQL (in particular, for parsing the cluster URIs), we only make it available as part of the Extended data.
c) OrthoDB Homologous Groups

The following code can be executed at the OrthoDB SPARQL endpoint webpage at https://sparql.orthodb.org.

```sparql
PREFIX orthodb: <http://purl.orthodb.org/>
PREFIX up: <http://purl.uniprot.org/core/>

VALUES ?protein2_uniprot {<http://purl.uniprot.org/uniprot/P68871>}
VALUES ?taxLevel {"Primates"}
?gene2 a orthodb:Gene.
?gene2 orthodb:memberOf ?groups.
?gene1 a orthodb:Gene.
?gene1 orthodb:memberOf ?groups.
?organism a ?taxon.
?taxon up:scientificName ?species_name.
?groups orthodb:ogBuiltAt ?taxLevel_uniprot.
?taxLevel_uniprot up:scientificName ?taxLevel.
?protein2_uniprot a orthodb:Uniprot.
?xref a orthodb:Xref.
OPTIONAL{
?protein1_uniprot a orthodb:Uniprot.}
} ORDER BY ?groups, ?taxLevel
```

This SPARQL query will retrieve flat homologous groups (i.e. clusters) that contains the human HBB gene in OrthoDB. The HBB gene is represented with its related UniProt entry (i.e. the UniProt URI http://purl.uniprot.org/uniprot/P68871).

Protocol 3: Retrieve Hierarchical Orthologous Groups (HOG)

In this protocol we show how to retrieve the HOGs containing a target gene, such as the human HBB gene, in the three orthology databases OMA, MBGD and OrthoDB. The Ensembl dataset in the EBI RDF platform is not considered because it does not provide HOG information.

a) Retrieving HOGs from OMA

The following code fragment can be executed at the OMA SPARQL endpoint webpage at https://sparql.omabrowser.org/lode/sparql.

```sparql
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX orth: <http://purl.org/net/orth#>
PREFIX taxon: <http://purl.uniprot.org/taxonomy/>
PREFIX up: <http://purl.uniprot.org/core/>
PREFIX lscr: <http://purl.org/lscr#>

SELECT DISTINCT ?root_hog ?species_name ?protein1_uniprot (?protein1 as ?protein1_OMA) ?taxLevel {
VALUES ?protein2_uniprot {<http://purl.uniprot.org/uniprot/P68871>}
?root_hog obo:CDAO_0000148 ?hog_cluster. #has_Root
?hog_cluster orth:hasHomologousMember* ?node1.
?node1 a orth:OrthologsCluster.
?node1 orth:hasTaxonomicRange ?level.
?node1 orth:hasHomologousMember* ?protein1.
?hog_cluster orth:hasHomologousMember* ?protein2.
?protein1 a orth:Protein.
?organism a ?taxon.
?taxon up:scientificName ?species_name.
?groups orthodb:ogBuiltAt ?taxLevel_uniprot.
?taxLevel_uniprot up:scientificName ?taxLevel.
?protein2_uniprot a orthodb:Uniprot.
?xref a orthodb:Xref.
OPTIONAL{
?protein1_uniprot a orthodb:Uniprot.}
} ORDER BY ?taxLevel
```

This SPARQL query will retrieve hierarchical orthologous groups that contain the human HBB gene in the OMA database. The HBB gene is represented with its related UniProt entry (i.e. the UniProt URI http://purl.uniprot.org/uniprot/P68871).

b) Retrieving HOGs from MBGD

The SPARQL query to retrieve HOGs from MBGD is similar to the previous query over OMA and therefore we make it available as Extended data. As a reminder, although both the OMA and MBGD databases rely on different versions of the ORTH ontology, they structure their HOG data similarly.

c) Retrieving HOGs from OrthoDB

The following code fragment can be executed at the OrthoDB SPARQL endpoint webpage at https://sparql.orthodb.org. Note that the OrthoDB HOGs often do not contain all taxonomic levels between the orthologous cluster at the highest taxonomic level (i.e. the root) and genes (i.e. leaves).
This SPARQL query will retrieve hierarchical orthologous groups that contains the human HBB gene in the OrthoDB database. The HBB gene is represented with its related UniProt entry (i.e. the UniProt URI http://purl.uniprot.org/uniprot/P68871).

Protocol 4: Meta-analysis - comparing data across OMA and MBGD orthology

In this protocol, we show how to compare orthology information across multiple databases with SPARQL 1.1. Although the example in the following code fragment is restricted to OMA and MBGD, similar queries over different combinations of the orthology databases mentioned in this article can be derived based on the Code Fragments in Protocols 1, 2 and 3.

For a given UniProt entry such as the accession number K9Z723, retrieve orthologs that are only in MBGD, but not in OMA. Alternatively, to retrieve only those that appear in both sources, simply remove the “NOT” keyword in the FILTER clause below. To execute the query, copy and paste it in the SPARQL endpoint page of OMA: https://sparql.omabrowser.org/lode/sparql.

This federated SPARQL query will retrieve pairwise orthologous genes of the Cyanobacterium-aponinum psb27- gene that are found in the MBGD database but are not present in OMA. The psb27 gene is represented with its related UniProt entry, thus the UniProt URI http://purl.uniprot.org/uniprot/K9Z723.

Aggregations in SPARQL: Combining data from multiple resources

In the Extended data, we provide additional examples showing how to retrieve the top 10 entries with most orthologs in OMA and MBGD for a given species, e.g. ‘Drosophila melanogaster’. These examples illustrate a few more advanced SPARQL features, such as aggregation and ordering by a criterion in order to select the top N results.

Conclusion

We provide four protocols that show how to query evolutionary relationships (pairwise orthologs, as well as HOGs) across four major databases available through SPARQL 1.1 endpoints: EBI, OMA, MBGD and OrthoDB. The protocols presented
can serve as a useful starting point for readers interested in an introduction to the RDF data models of these sources, as well as the basics of retrieving orthology information through SPARQL queries. Finally, we have shown how aggregations in SPARQL can be used to quickly generate an overview of the data available in each considered database, and how this data can be compared across the sources.

To sum up, we hope these protocols provide a useful introduction into analysing evolutionary relationships among genes with SPARQL, as well as enriching these analyses by integrating information from external sources, through federated queries. We encourage readers to experiment with the examples presented in this article, which are provided in the accompanying Jupyter notebook, to be directly re-used or integrated into existing research analysis pipelines. As future work, we plan to integrate the queries in this protocol in the BioQuery search interface (Sima et al., 2019) already available for OMA data at http://biosoda.expasy.org/ in order to enable researchers to directly execute or further refine them in a user-friendly environment.

**Data availability**

**Underlying data**


Archived protocols as at time of publication: http://doi.org/10.5281/zenodo.3499928

License: CC0

---

**References**


W3C SPARQL WORKING GROUP, et al.: SPARQL 1.1 overview. W3C

---

**Extended data**


This project contains the following extended data:

- Table 1. “Cheat sheet” for RDF data available in the four sources considered in this tutorial. (*) GO annotations can be retrieved from the UniProt RDF store through UniProt cross-references.
- Supplementary protocols: Retrieving MBGD Homologous Groups
- Aggregation queries

Data are available under the terms of the Creative Commons Zero “No rights reserved” data waiver (CC0 1.0 Public domain dedication).

**Software availability**

SPARQL endpoints (all links include further example queries):

- **OMA** https://sparql.omarowser.org/lode/sparql
- **OrthoDB** https://sparql.orthodb.org/
- **EBI** https://www.ebi.ac.uk/rdf/services/sparql

  - for example orthology queries see “Ensembl” category
- **MBGD** http://mbgd.genome.ad.jp/sparql/

---

Reference Source


Publisher Full Text


PubMed Abstract | Publisher Full Text


PubMed Abstract | Publisher Full Text | Free Full Text


PubMed Abstract | Publisher Full Text | Free Full Text
The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com