SPARQL Queries

- Overview
- Running SPARQL queries
- Using SPARQL for reporting
- Using SPARQL to clean data
- DESCRIBE queries
- ASK Queries
- Additional SPARQL Resources

Overview

SPARQL is a query language for RDF-based systems such as Vitro and VIVO. Using SPARQL one can extract any information from VIVO or Vitro, producing reports, or providing data for other software such as visualizations or user interfaces.

Running SPARQL queries

To run a SPARQL query, navigate to Site Admin > Sparql query. You will see a page similar to the one below. Note that the text is in various colors. VIVO uses the YASQE editor for SPARQL. You can read more about its features at their web site. See http://yasqe.yasgui.org/ At the top of the Query window are SPARQL prefix declarations for VIVO. These provides abbreviations for various URLs you use in your SPARQL queries. For more on SPARQL, see Learning SPARQL. http://learningsparql.com Below the prefix declarations is the actual SPARQL query.

The query in the figure above asks for a sorted list of the entities in VIVO that have type Relationship. RS_TEXT has been selected as an output format. This will display in a browser window. Note that you can select CSV, or TSV. These will download to your computer. You may also select RS_XML and RS_JSON – these will also display in your browser window.

Running the above query in OpenVIVO generates over 10,000 rows of output. The beginning of the output in JSON format is shown below:

First few lines of JSON output for preceding query


```
{
  "head": {
    "vars": [ "s" , "p" , "o" ]
  } ,
  "results": {
    "bindings": [
      {
        "s": { "type": "uri" , "value": "http://openvivo.org/a/doi10.4225/03/58ca600d726bd-authorship1" }
      } ,
      {
        "s": { "type": "uri" , "value": "http://openvivo.org/a/doi10.4225/03/58ca600d726bd-authorship2" }
      } ,
      {
        "s": { "type": "uri" , "value": "http://openvivo.org/a/doi10.6084/m9.figshare.2002020-authorship1" }
      } ,
      {
        "s": { "type": "uri" , "value": "http://openvivo.org/a/doi10.6084/m9.figshare.2002200-authorship1" }
      } ,
      {
        "s": { "type": "uri" , "value": "http://openvivo.org/a/doi10.6084/m9.figshare.2002200-authorship2" }
      } ,
```


Using SPARQL for reporting

SPRQL can be used to extract data from VIVO for reporting. Using the TSV (tab separated values) output format, the results of a SPARQL query can be downloaded from VIVO and uploaded to a spreadsheet, reporting or presentation tool.

The query below makes a contact list for all people in particular academic unit.

```
#
# Find all the people with a position in the CTSI or any CTSI sub-unit,
# and list them alphabetically with phone, email, gatorlink, eracommons if any
#
SELECT ?person (MIN(DISTINCT ?xname) AS ?name)
    (MIN(DISTINCT ?xphone) AS ?phone)
    (MIN(DISTINCT ?xemail) AS ?email)
    (MIN(DISTINCT ?xgatorlink) AS ?gatorlink)
    (MIN(DISTINCT ?xeracommons) AS ?eracommons)
WHERE {
  {?pos vivo:relates <http://vivo.ufl.edu/individual/n8763427> . ?pos a vivo:Position .}
  UNION
  {<http://vivo.ufl.edu/individual/n8763427> obo:BFO_0000051 ?sub .
   ?pos vivo:relates ?sub . ?pos a vivo:Position .}
  ?pos vivo:dateTimeInterval ?dt .
  OPTIONAL {?dt vivo:end ?end . }
  FILTER (!BOUND(?end))  # current positions do not have end dates
  ?pos vivo:relates ?person . ?person a foaf:Person .
  ?person rdfs:label ?xname .
  ?person a ufVivo:UFCurrentEntity .
  ?person obo:ARG_2000028 ?vcard .
  OPTIONAL { ?vcard vcard:hasEmail ?email_thing . ?email_thing vcard:email  ?xemail .}
  OPTIONAL { ?vcard vcard:hasTelephone ?tel_thing . ?tel_thing vcard:telephone ?xphone .}
  OPTIONAL { ?person ufVivo:gatorlink ?xgatorlink .}
  OPTIONAL { ?person vivo:eRACommonsId ?xeracommons .}
}
GROUP BY ?person
ORDER BY ?name
```

For additional examples, some much simpler than the example above, see Mike Conlon's web site http://mconlon17.github.io/sparql

Using SPARQL to clean data

Using SPARQL queries, one can find triples meeting a criteria for improvement. You might query for people without labels, for example.

CONSTRUCT statements can be used to make triples that you would like to remove from your VIVO. Run the query to make the triples, download them, then use System Admin > Add/Remove RDF data to Remove the triples you have constructed. In a similar manner, you can construct improved triples and add them to you VIVO.

DESCRIBE queries

Vitro SPARQL supports DESCRIBE queries, but DESCRIBE is not well-defined by W3C standards, allowing implementation specific variations. In Vitro, a request to DESCRIBE a URL will return all the triples with that URL as the subject. So, for example:

A sample DESCRIBE query

```
DESCRIBE <http://openvivo.org/a/doi10.4225/03/58ca600d726bd>
```

returns

```
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.obolibrary.org/obo/ARG_2000028>
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd-vcard> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#relatedBy>
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd-authorship2> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.obolibrary.org/obo/BFO_0000031> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.obolibrary.org/obo/RO_0002353>
<http://openvivo.org/a/eventVIVO2017> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.org/ontology/bibo/abstract> "<div>This is the
pre-print version of a paper accepted in Open Repository Conference in Brisbane, Australia, June 2017.<b><br><
/b></div><div><br></div><b>Abstract\u00A0</b><div><b><br></b><div>Research Graph is an open collaborative
project that builds the capability for connecting researchers, publications, research grants and research
datasets (data in research). \u00A0VIVO is an open source, semantic web platform and a set of ontologies for
representing scholarship. \u00A0To provide interoperability between Research Graph data and VIVO systems we
modelled the Research Graph metamodel using the VIVO Integrated Semantic Framework. To evaluate the mapping, we
used the model to connect figshare RDF records to data collections in Research Data Australia using Research
Graph API. In addition, we are working toward loading Research Graph data into a VIVO instance. \u00A0VIVO
provides a search capability, and pages for first class entities in the Research Graph model -- researcher,
dataset, grant, and publication. \u00A0The result provides a visualisation solution for co-authors, co-funding,
timeline, and a capability map for finding expertise related to concepts of interest. \u00A0The resulting
linked open data will be made freely available and can be used in other tools for additional discovery.<br><
/div></div>" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.org/ontology/bibo/doi> "10.4225/03
/58ca600d726bd" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#datePublished>
<http://openvivo.org/a/date2017-03-16> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#relatedBy>
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd-authorship1> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.org/ontology/bibo/freetextKeyword> "Research
Discovery" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://www.w3.org/2002/07/owl#Thing> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#dateCreated>
<http://openvivo.org/a/date2017-03-16> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.org/ontology/bibo/Document> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.org/ontology/bibo/freetextKeyword> "Linked
Open Data" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vitro.mannlib.cornell.edu/ns/vitro/0.
7#mostSpecificType> <http://vivoweb.org/ontology/core#ConferencePaper> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.obolibrary.org/obo/BFO_0000001> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/2000/01/rdf-schema#label> "Creating an
open linked data model for Research Graph using VIVO Ontology" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://purl.org/ontology/bibo/freetextKeyword> "Open
research outputs" .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.obolibrary.org/obo/IAO_0000030> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.obolibrary.org/obo/BFO_0000002> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#dateTimeValue>
<http://openvivo.org/a/date2017-03-16> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://vivoweb.org/ontology/core#dateModified>
<http://openvivo.org/a/date2017-03-16> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://vivoweb.org/ontology/core#ConferencePaper> .
<http://openvivo.org/a/doi10.4225/03/58ca600d726bd> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://purl.org/ontology/bibo/Article> .
```

ASK Queries

Vitro SPARQL supports ASK queries which return either true (there are triples that satisfy the pattern), or false (there are no triples that satisfy the pattern). The query below will return true in most VIVOs and false in a new VIVO.

ASK Query

```
ASK { ?s a vivo:Relationship . }
```

Additional SPARQL Resources

- YASQUE, "YASQUE Home Page," web site. Last Accessed June 17, 2017. http://yasqe.yasgui.org/
- Conlon, M. "Sample SPARQL: SPARQL scripts for getting information from your VIVO," web site. Last Accessed June 17, 2017. http://mconlon17.github.io/sparql/
- DuCharme, B. "Learning SPARQL," Wiley Publishing, 2011. 235 pages. http://www.learningsparql.com/
- Apache Jena, "SPARQL Tutorial," web site. Last accessed June 17, 2017. https://jena.apache.org/tutorials/sparql.html