Use Case 1.1 - Queries Examples

Planning | Description | Requirements | UI Mockup | Triples Examples | **Queries Examples**

This page needs to be re-written with a focus on SOLR queries instead of SPARQL queries.

Table of Contents

Common Questions

- Can queries be done across triplestores? The original plan was to reference an external URI for bibliographic references (e.g., `http://da-rdf.library.cornell.edu/individual/b3652730`) and not copy any information about the bibliographic reference into our local triplestore. The requirement to sort by title, author, and publication date with you to require that this information be copied into the local triplestore, or that the SPARQL queries are capable of querying across two distinct triplestores.
    - Drive app data from SOLR instead of triplestore.
    - Can search across multiple SOLR indexes?
        - Some say yes, some say no, looks like no.
        - Can make individual queries and meld results.
    - Can add data to my SOLR from multiple triplestores/apps?
        - YES, but questions about each app keeping multiple SOLR data in sync
        - For example, will library catalog system update my SOLR index when the name of a bib ref gets updated?
            - Need some process added to blacklight-cornell catalog that will update my SOLR index
            - Need some process to add my triple data into the cornell catalog SOLR index
            - Both of these are post Use Case and production implementation issues
    - Process:
        - adding an item
            - copy title, author, pubdate from blacklight-cornell catalog system into my SOLR index
            - updates to virtual collection items go into triplestore and into SOLR index
        - queries go to my SOLR index
    - Why separate SOLR indices? Efficiency of queries, data management (buffer against changes), ease of use by multiple apps
- How does SPARQL handle sorting?
    - look at order_by
- How does SPARQL track paginated results? How does SPARQL track tokens for retrieving the next set of search results?
    - look at limit and offset
- Missing Triples for an 'object' -- From the SPARQL doc, it appears that if any one of the 'objects' (e.g., title, description, size, visibility) listed in the WHERE of the query is not defined for a virtual collection, then that virtual collection will not be included in the results. I have not tested this. Is there a way around this limitation? I'd like the value for the missing object to be _BLANK_ or nil.
    - Example from SPARQL doc

      Data:

```
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .

_:a  foaf:name   "Johnny Lee Outlaw" .
_:a  foaf:mbox   <mailto:jlow@example.com> .
_:b  foaf:name   "Peter Goodguy" .
_:b  foaf:mbox   <mailto:peter@example.org> .
_:c  foaf:mbox   <mailto:carol@example.org> .
```

Query:

```
PREFIX foaf:   <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox
WHERE
  { ?x foaf:name ?name .
    ?x foaf:mbox ?mbox }
```

Query Result:

name	mbox
"Johnny Lee Outlaw"	<mailto:jlow@example.com>
"Peter Goodguy"	<mailto:peter@example.org>

Question:
- Why isn't mbox=<mailto:carol@example.org> part of the result with name=_BLANK_?

Get Metadata for one RDF Resource

Get Metadata for a Virtual Collection

Data Referenced in Query

- VirtualCollectionRDF.rdf_subject == _VIRTUAL_COLLECTION_URI_

Data Returned

- VirtualCollectionRDF.title
- VirtualCollectionRDF.description
- VirtualCollectionRDF.size
- VirtualCollectionRDF.visibility

Example Query

```
# _CURRENT_USER_URI_ = "http://localhost:3000/individual/p234"
# _VIRTUAL_COLLECTION_URI_ = "http://localhost:3000/individual/vc123"
SELECT ?title ?description ?size ?visibility
WHERE
{
   <http://localhost:3000/individual/vc123> <http://purl.org/dc/terms
/creator> <http://localhost:3000/individual/p234> .
   <http://localhost:3000/individual/vc123> <http://purl.org/dc/elements/1.
1/title>       ?title .
   <http://localhost:3000/individual/vc123> <http://purl.org/dc/elements/1.
1/description> ?description .
   <http://localhost:3000/individual/vc123> <http://purl.org/co
/size>                     ?size .
   <http://localhost:3000/individual/vc123> <http://TBD
/visibility>                       ?visibility .
}
```

Notes

- This should be fairly straightforward query that matches a single Virtual Collection.
- The inclusion of the <virtual collection> <owner> <person> triple is a sanity check that the current user has permissions to view this item.
- The implementation of visibility (i.e., PUBLIC, PRIVATE, SHARED) has not yet been fully designed.

Questions

- See also Common Questions section at the top of this document, especially the question about missing triples for an 'object'.
- size is not currently defined in VirtualCollectionRDF because I'm modeling the ORE ontology. It is defined in the Collections ontology. Is there an equivalent in the ORE ontology?

Get Metadata for a Virtual Collection Item

Data Referenced in Query

- VirtualCollectionItemRDF.rdf_subject == _VIRTUAL_COLLECTION_ITEM_URI_

Data Returned

- VirtualCollectionItemRDF.rdf_subject
- VirtualCollectionItemRDF.proxyFor
- VirtualCollectionItemRDF.visibility
- BibliographicReferenceRDF.rdf_subject
- BibliographicReferenceRDF.title
- BibliographicReferenceRDF.description
- BibliographicReferenceRDF.author
- BibliographicReferenceRDF.pubdate

Example Query

```
# _CURRENT_USER_URI_ = "http://localhost:3000/individual/p234"
# _VIRTUAL_COLLECTION_ITEM_URI_ = "http://localhost:3000/individual/vci456"
SELECT ?vc_subject ?title ?description ?author ?pubdate ?visibility
WHERE
{
   <http://localhost:3000/individual/vci456> <http://www.openarchives.org
/ore/terms/proxyIn>    ?vc_subject
   ?vc_subject  <http://purl.org/dc/terms/creator> <http://localhost:3000
/individual/p234> .
   <http://localhost:3000/individual/vci456> <http://www.openarchives.org
/ore/terms/proxyFor>   ?biburi
   ?vc_subject                               <http://www.openarchives.org
/ore/terms/aggregates> ?biburi
   <http://localhost:3000/individual/vci456> <http://TBD
/visibility>                            ?visibility .
   ?biburi                                   <http://purl.org/dc/elements
/1.1/title>            ?title .
   ?biburi                                   <http://purl.org/dc/elements
/1.1/description>      ?description .
   ?biburi                                   <http://purl.org/dc/elements
/1.1/creator>          ?author .
   ?biburi                                   <http://purl.org/dc/elements
/1.1/date>             ?pubdate .
}
```

Notes

- The inclusion of the <virtual collection> <owner> <person> triple is a sanity check that the current user has permissions to view this item.
- The inclusion of the <virtual collection> <aggregates> <biburi> triple is a sanity check that this item is actually in the selected virtual colle
- The implementation of visibility (i.e., PUBLIC, PRIVATE, SHARED) has not yet been fully designed.

Questions

- See also Common Questions section at the top of this document, especially the question about missing triples for an 'object'.

Get List of Virtual Collections with Pagination

Owned by Me and Sorted by Collection Name

Data Referenced in Query

- VirtualCollectionRDF.owner == _CURRENT_USER_URI_
- number of virtual collections to return in the results
- token identifying starting point for next set of results

Data Returned

- VirtualCollectionRDF.rdf_subject
- VirtualCollectionRDF.title
- VirtualCollectionRDF.description
- VirtualCollectionRDF.size
- VirtualCollectionRDF.visibility

Example Query

Example is INCOMPLETE -- Does not address sorting and pagination issues

```
# _CURRENT_USER_URI_ = "http://localhost:3000/individual/p234"
SELECT ?vc_subject ?title ?description ?size ?visibility
WHERE
{
   ?vc_subject <http://purl.org/dc/terms/creator>
<http://localhost:3000/individual/p234> .
   ?vc_subject <http://purl.org/dc/elements/1.1/title>       ?title .
   ?vc_subject <http://purl.org/dc/elements/1.1/description> ?description .
   ?vc_subject <http://purl.org/co/size>                     ?size .
   ?vc_subject <http://TBD/visibility>                       ?visibility .
}
```

Notes

- The concept of owner will not be represented by dc:creator. See Triples Examples page for more details.
- The implementation of visibility (i.e., PUBLIC, PRIVATE, SHARED) has not yet been fully designed.
- Sorting: look at order_by
- Pagination: look at limit and offset
- Better option: pull data from SOLR
  - look at how ActiveFedora triggers updates to SOLR
  - VIVO also has triggers to update SOLR that may be useful

Questions

- See also Common Questions section at the top of this document, especially for sorting and pagination, and missing triples for an 'object'.
- size is not currently defined in VirtualCollectionRDF because I'm modeling the ORE ontology. It is defined in the Collections ontology. Is there an equivalent in the ORE ontology?

Watched by Me and Sorted by Collection Name

NOTE: 'Watched by' implementation is TBD.

Public and Sorted by Collection Name

NOTE: 'Visibility' implementation is TBD.

Data Referenced in Query

- VirtualCollectionRDF.visibility == "PUBLIC"   (NOTE: ' Visibility' implementation is TBD.)

Data Returned

- VirtualCollectionRDF.rdf_subject
- VirtualCollectionRDF.title
- VirtualCollectionRDF.description
- VirtualCollectionRDF.size   (NOTE: Defined in Collections ontology. Is there an equivalent in the ORE ontology?)

Example Query

Example is INCOMPLETE -- Does not address sorting and pagination issues

Example is INCOMPLETE -- Visibility design is under development.

```
SELECT ?vc_subject ?owner ?title ?description ?size
WHERE
{
   ?vc_subject <http://TBD/visibility>                       "PUBLIC" .
   ?vc_subject <http://purl.org/dc/terms/creator>            ?owner .
   ?vc_subject <http://purl.org/dc/elements/1.1/title>       ?title .
   ?vc_subject <http://purl.org/dc/elements/1.1/description> ?description .
   ?vc_subject <http://purl.org/co/size>                     ?size .
}
```

Notes

- The implementation of visibility (i.e., PUBLIC, PRIVATE, SHARED) has not yet been fully designed.

Questions

- See also Common Questions section at the top of this document, especially for sorting and pagination, and missing triples for an 'object'.
- Same as for "Owned by Me and Sorted by Collection Name"

Get List of Items in a Virtual Collection with Pagination

Sorted by List Order

Data Referenced in Query

- VirtualCollectionRDF.owner == _CURRENT_USER_URI_
- VirtualCollectionRDF.rdf_subject == _SELECTED_VIRTUAL_COLLECTION_URI_
- VirtualCollectionItemRDF.proxyFor == _SELECTED_VIRTUAL_COLLECTION_URI_
- VirtualCollectionItemRDF.next - assume it plays a role in the query since this predicate specifies the list order
- number of items to return in the results
- token identifying starting point for next set of results

Data Returned

- VirtualCollectionItemRDF.rdf_subject
- VirtualCollectionItemRDF.proxyFor
- VirtualCollectionItemRDF.visibility
- BibliographicReferenceRDF.rdf_subject
- BibliographicReferenceRDF.title
- BibliographicReferenceRDF.description
- BibliographicReferenceRDF.author
- BibliographicReferenceRDF.pubdate

Example Query

Example is INCOMPLETE -- Does not address sorting and pagination issues

```
# _CURRENT_USER_URI_ = "http://localhost:3000/individual/p234"
# _SELECTED_VIRTUAL_COLLECTION_URI_ = "http://localhost:3000/individual
/vc123"
SELECT ?vci_subject ?title ?description ?author ?pubdate
WHERE
{
   ?vci_subject <http://www.openarchives.org/ore/terms/proxyIn>
<http://localhost:3000/individual/vc123>
   <http://localhost:3000/individual/vc123> <http://purl.org/dc/terms
/creator> <http://localhost:3000/individual/p234> .
   ?vci_subject <http://www.openarchives.org/ore/terms/proxyFor> ?biburi
   <http://localhost:3000/individual/vc123> <http://www.openarchives.org
/ore/terms/aggregates> ?biburi
   ?vci_subject <http://TBD/visibility>                          ?
visibility .
   ?biburi      <http://purl.org/dc/elements/1.1/title>          ?title .
   ?biburi      <http://purl.org/dc/elements/1.1/description>    ?
description .
   ?biburi      <http://purl.org/dc/elements/1.1/creator>        ?author .
   ?biburi      <http://purl.org/dc/elements/1.1/date>           ?pubdate .
}
```

Notes

- The concept of owner will not be represented by dc:creator. See Triples Examples page for more details.
- The inclusion of the <virtual collection> <owner> <person> triple is a sanity check that the current user has permissions to view this item.
- The inclusion of the <virtual collection> <aggregates> <biburi> triple is a sanity check that this item is actually in the selected virtual collection.
- The implementation of visibility (i.e., PUBLIC, PRIVATE, SHARED) has not yet been fully designed.

Questions

- See also Common Questions section at the top of this document, especially for sorting and pagination, and missing triples for an 'object'.

Sorted by Bibliographic Resource Title

Data Referenced in Query

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order except...
  - REMOVE REF TO: VirtualCollectionItemRDF.next
  - ADD REF TO: BibliographicReferenceRDF.title

Data Returned

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order

Example Query

- When Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order is fully specified, the example query for this sort order will be created. Several issues for Sorted by List Order apply to all the sort queries.

Notes

Questions

- Same as for Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order.

Sorted by Bibliographic Author

Data Referenced in Query

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order except...
  - REMOVE REF TO: VirtualCollectionItemRDF.next
  - ADD REF TO: BibliographicReferenceRDF.description

Data Returned

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order

Example Query

- When Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order is fully specified, the example query for this sort order will be created. Several issues for Sorted by List Order apply to all the sort queries.

Notes

Questions

- Same as for Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order.

Sorted by Bibliographic Publication Date

Data Referenced in Query

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order except...
  - REMOVE REF TO: VirtualCollectionItemRDF.next
  - ADD REF TO: BibliographicReferenceRDF.pubdate

Data Returned

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order

Example Query

- When Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order is fully specified, the example query for this sort order will be created. Several issues for Sorted by List Order apply to all the sort queries.

Notes

Questions

- Same as for Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order.

Unsorted with no List Order

Data Referenced in Query

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order except...
  - REMOVE REF TO: VirtualCollectionItemRDF.next

Data Returned

- Same as for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order

Example Query

- When Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order is fully specified, the example query for this sort order will be created. Several issues for Sorted by List Order apply to all the sort queries.

Notes

Questions

- Same as for Query for Get List of Items in a Virtual Collection with Pagination - Sorted by List Order.
  - Pagination issues are the same.
  - Sort by issues will be different since no sort order is applied to the virtual collection.