

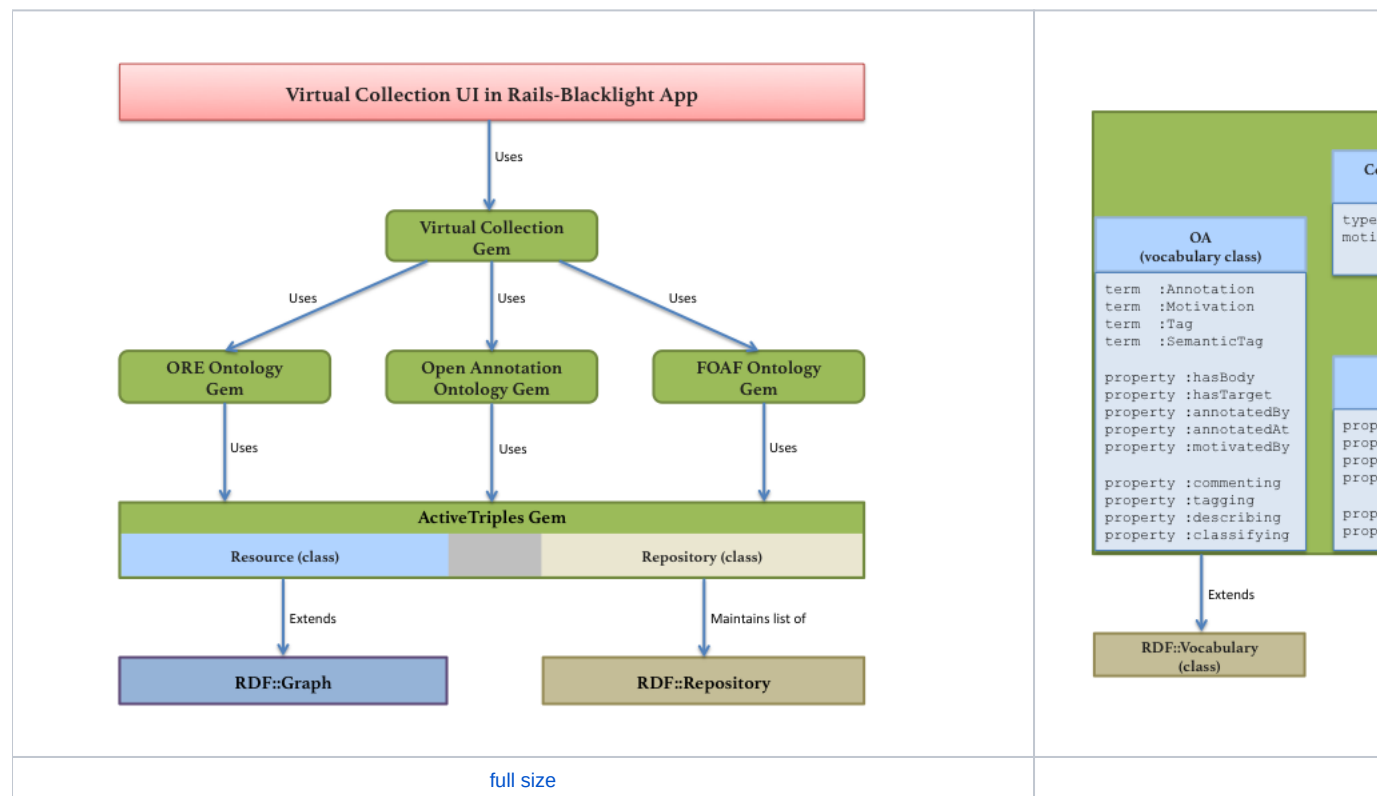
Use Case 1.1 - PLANNING - Build a virtual collection

[Planning](#) | [Description](#) | [Requirements](#) | [UI Mockup](#) | [Triples Examples](#) | [Queries Examples](#)

Table of Contents

- [Architectural Diagram of Virtual Collection GEM and Ontology GEMs](#)
 - [Designs](#)
- [Planned Stack](#)
- [Development Status](#)
 - [Sprints - Planned Development Approach](#)
 - [Sprint 1: Add a single work from an individual work page to a single supported virtual collection named My Virtual Collection.](#)
 - [Sprint 2: Use ActiveTriples instead ActiveFedora](#)
- [Ontology Questions](#)
- [UI Questions](#)
- [Status of Planning Process](#)

Architectural Diagram of Virtual Collection GEM and Ontology GEMs



Designs

- [ORE Ontology GEM Implementation Design](#)

Planned Stack

- Infrastructure – SHARED — code that can be shared by all universities
 - **RDF Models for Triples:** RDF Model classes and RDF Vocabularies (beyond those defined in `ruby-rdf/lib/rdf/vocab`) are defined in a local Ruby on Rails app.
 - **Developing with Ruby:** 2.1.2
 - **Developing with Rails:** 4.1.1
 - **Gems:** Active-Triples
 - The RDF Model classes extend the Resource class defined in the Active-Triples gem.
 - NOTE: Active-Triples brings in code from several other projects. I'm not sure of the full list, but `ruby-rdf/rdf` is certainly used by Active-Triples.
 - **SOLR Index**

- Triples will be set up to go to a SOLR index using code currently in or soon to be developed code in Active-Triples gem. I know there is some SOLR support, but I don't know how extensive it is. For example, we will want to put some information in our SOLR index that isn't going into the triplestore and may need to expand the code to accomplish this.
 - UI will query the SOLR index to retrieve data to display on the screen
 - Triplestore
 - I am currently developing over a simple triplestore in sqlite3. We will be moving the final version to a fuller featured triplestore long term. The triplestore is configured as part of the ActiveTriple gem and should not be difficult to move to a different triplestore system. At this point, it is envisioned that this will be a standalone triple store that is separate from the main library catalog triplestore that will be created through the conversion process.
- Infrastructure – Potentially Shared – code that may be able to be shared by all universities
 - Conversion of university id (e.g. Cornell netid) into a VIVO URI. Each university may need to write this utility for their IDs with a conversion to the URI of choice for that university.
- User Interface — NOT SHARED — all that follows is Cornell specific
 - Authentication Code
 - For Cornell, we will connect into our Shibboleth login system for authentication.
 - UI based on mockups in the wiki will be integrating the infrastructure code into the Cornell catalog system which is implemented over backlight.

Development Status

Ok, best laid plans of mice and men. Implementation is not following the [Sprints - Planned Development Approach](#). Here are where things are currently...

Status	Description	Comments
Infrastructure - Shared		
RDF Models & Vocabularies		
	Write models and vocabularies that allow for generation of triples for data required for Use Case 1.1.	- Models complete and passing rspec tests. Pushed as a single GEM - <code>ld4l-virtual_collection</code>. - Splitting single GEM into one GEM per ontology used. See architectural diagram above. OpenAnnotation & FOAF GEMS are complete and tested. Working on ORE GEM. - Models use the ActiveTriples gem
	Document models in Triples Examples wiki page to facilitate conversations with ontology group.	outstanding issues - owner, contributors - Need to look into PROV ontology (per suggestion from Paolo) - lists - There are still discussions about whether to represent lists as ORE or Collections. I plan to wait until I am further along and know more about the implications for implementation for each model.
	Write rspec tests for models.	- The test for the first model is close to being complete, but not sure what issues will arise in the final tests - The others should be quick to write since they are going to be very similar to the tests for the first model
Triplestore		
	Set up a temporary simple triplestore to hold triples generated by this use case.	Using SQLite3
	Identify and set up a full-featured triplestore for long-term use.	DEPENDENCY: Waiting and see what Rebecca decides to use for her work.
	Write code that persists model properties into a triplestore.	The persistence to the triplestore is handled by the ActiveTriples::Resource class. When RDF Models are complete, this will also be complete.
SOLR Index		
	Explore ActiveTriple support for SOLR	I have located some code in ActiveTriple gem's rspec tests that manipulates SOLR. I need to explore this further.

	Document expected queries in Queries Examples wiki page to facilitate conversations with ontology group.	The queries are currently defined as SPARQL queries, so they need to be rewritten as SOLR queries.
	Write code that adds model properties' values to the SOLR index.	
	Write code that adds bibliographic reference data to the SOLR index.	
Infrastructure - Potentially Shared		
netid to VIVO URI script		
	Write code that converts a Cornell netid into a VIVO URI.	- I have information from Jim Blake about how to get the information from VIVO. It involves getting the information through a curl command and then parsing the returned result. - Need to write code to do the conversion. (Was hoping for a webservice written by VIVO that did this.)
User Interface – NOT SHARED		
Authentication		
	Hook into Cornell shibboleth authentication system.	I've spoken with CULIT developers about how Cornell does authentication and looked over code that is similar to what will need to be done. No development work has been done in this are yet.
User Interface		
	Clone blacklight-cornell and check into GitHub	The clone runs locally on my machine and in my dev environment. Questions: Should the code go into LD4L GitHub or Cornell GitHub. It is a direct clone of the CULIT Discovery and Access system.
	Modify cloned blacklight-cornell to implement the Mock UI	- I've explored the code to see where some of the changes need to be made, but I haven't made any changes yet. I want to get the cloned code in a source code repository before beginning to make modifications. - For details, see also the Use Case 1.1 - Requirements. They are UI centric.
External Collaborations		
ActiveTriples development		
	Make a connection with the ActiveTriples development team.	I have been corresponding with Tom Johnson (the developer of ActiveTriples).
	Work with ActiveTriples development team to meet the goal of being able to use ActiveTriples in place of Fedora within a Hydra stack.	I am exploring the code and engaging in conversations with Tom Johnson to understand his vision and current needs for development. Once I grok where things are, I can start to see if there are shortcomings that limit our goal of ActiveTriples being easy to use in place of Fedora with in a Hydra stack.
	Improve documentation by creating a reference guide for ActiveTriples	There is very minimal doc at this point and I am having to read class and rspec code to figure out how things work. A reference guide will be an extremely valuable tool for me and other developers.

Sprints - Planned Development Approach

Sprint 1: Add a single work from an individual work page to a single supported virtual collection named My Virtual Collection.

- Limitation: Use current login scheme (will require multiple logins by user) **? Is this workable at all ?**
- UI: All pages get link to My Virtual Collections in the banner
- UI: Add new My Virtual Collections page
 - LHS: List only My Virtual Collections
 - LHS: Cannot create new virtual collections. (Only a single collection named My Virtual Collection will be allowed for this sprint and it will be created automatically the first time the user adds a work.)
 - LHS: List the one and only virtual collection named My Virtual Collection
 - Name: My Virtual Collection
 - Description: A collection of resources that interest me.

- iii. Access: Private
- d. LHS: Cannot delete My Virtual Collection
- e. LHS: (maybe) See information (aka metadata) for My Virtual Collection
- f. RHS: Show virtual collection My Virtual Collection
 - i. Show title My Virtual Collection
 - ii. If item added, show add message.
 - iii. Show count of items and range of items displayed.
 - iv. Limitation: Support only 20 per page.
 - v. Limitation: No sorting.
 - vi. Handle case where collection has 0 items.
 - vii. Handle case when the number of items is less than 20 per page setting.
 - viii. Handle case when number of items is more than 20 per page setting.
 - ix. Limitation: Cannot Move items
 - x. Support deleting an item from My Virtual Collection
- 4. UI: Work page gets new button to add the work to the one and only virtual collection named My Virtual Collection
 - a. When added, the page forwards to My Virtual Collections page and shows the newly added item.

Sprint 2: Use ActiveTriples instead ActiveFedora

TBA

Ontology Questions

NOTE: Many of these questions have been answered. I haven't had time to update the answers here. Most of the Ontology Questions are on the [Triples Examples](#) page including answers and current state of the discussion.

User

- How to represent a user?
 - Possibility <foaf:person>
- How to represent ownship/authorship of a user?
 - <co:list> <dc:creator> <foaf:person>
- How to represent user role to list type relationships
 - <foaf:person> owns <co:list>
 - <foaf:person> canEdit <co:list> – facilitates collaboration
 - <foaf:person> canView <co:list>
- How to represent user to group to list type relationships
 - <foaf:person> isIn <group>
 - <group> canEdit <co:list>
 - <group> canView <co:list>
- Are groups and roles in the triplestore?
 - Or are they maintained separately and in the triple store on as a uri to the group?
 - Issue: Any privacy issues with user data being repeated in our triplestore?
 - Issue: What information do we have access to and permission to put in a public triplestore?

Virtual Collection

- How to represent metadata about the virtual collection?
 - basic metadata
 - title – use <rdf:label>?
 - description
 - controlled vocabulary metadata
 - list of selected vocabularies (ordered list) – use <co:List>?
 - what metadata about each controlled vocabulary will be stored in the triplestore?
 - name – use <rdf:label>?
 - description
 - id:label for each term in the vocabulary – use <co:List>?
 - Are these stored in the main triplestore or as a test file in the file system that is accessible via URL?
 - NOTE: Questioning Authority has them as files in the rails app structure at /config/authorities, which is not accessible via URL and has a non-triple format.
 - access controls - How to record metadata about? How to limit access to triples related to virtual collections based on access level?
 - private (only the owner & collaborators can view)
 - shared (semi-private, only those with the access URL can view)
 - public (everyone can view)
 - collaborative (multi-editors)

UI Questions

NOTE: Some of these questions have been answered. I haven't had time to update these, so the questions may be stale. Hopefully, I will update soon.

- How does user stayed logged in to facilitate multiple searches to build a single Virtual Collection?
 - Currently, every time a user clicks My Account, a prompt for logging in is displayed, which implies users are logged out as soon as they leave the My Account page. ([Ask Adam the history of this.](#))
 - Can we add the notion of a public and private computer?

- Can we keep an IP address (or other identifier) for a computer that allows us to know this user is accessing the system from a previously authorized private computer?
 - What default length of time forces a log out? Is this settable by the user?
 - How to find a Virtual Collection to watch?
 - How to search only within the user's owned Virtual Collections?
 - How to search only within the user's watched Virtual Collections?
 - How are user public Virtual Collections exposed to the public?
 - How are works from external library catalogs searchable and available for inclusion in a Virtual Collection? (Potential Demonstration B)
 - The UI Mockup implies that items in a Virtual Collection can be dynamically sorted (show in title A-Z). Are dynamic sorts sufficient (title A-Z/Z-A, author A-Z/Z-A, year ASC/DES)? Potential Demonstration C states that the user can apply an order to items.
-

Status of Planning Process

- UI Mockup is fairly well developed for Potential Demonstrations A and D. See UI Questions above for areas that may not be represented in the mockup.
 - Requirements are fairly well developed for Potential Demonstrations A, C, and D.
 - See implementation status above in section [Development Status](#).
-