PROJECT Surfliner

SUPERSKUNK

Building a sidecar API service for a Hyrax application

OVERVIEW

At Project Surfliner, we are building a staff-facing Hyrax-based application (called **Comet**), which communicates with discovery platforms over a separate API service (called **Superskunk**). As Superskunk is a separate Rails application from the main repository system, we have depended on a number of technologies to keep metadata handling consistent between the two services.

HYRAX ACLs

Hyrax already has a very good model for managing access controls. In order to use this in Superskunk, however, we needed to extract it into a separate gem. The **Hyrax::Acl** gem in Samvera Labs is a drop-in replacement for Hyrax's existing ACL functionality for use in external applications (maybe one day, Hyrax itself?).

M3/HOUNDSTOOTH

In order for Superskunk to surface the metadata created in Comet, we needed to share a schema between them. We decided to use the **Houndstooth M3** model developed by the Samvera community for this, and built a Ruby gem for working with M3 in both applications.

We decided on M3 because it:

- Supports multiple conceptual "classes" of object.
- Supports mappings for various consumers/discovery platforms.
- Was developed by the Samvera community :) .
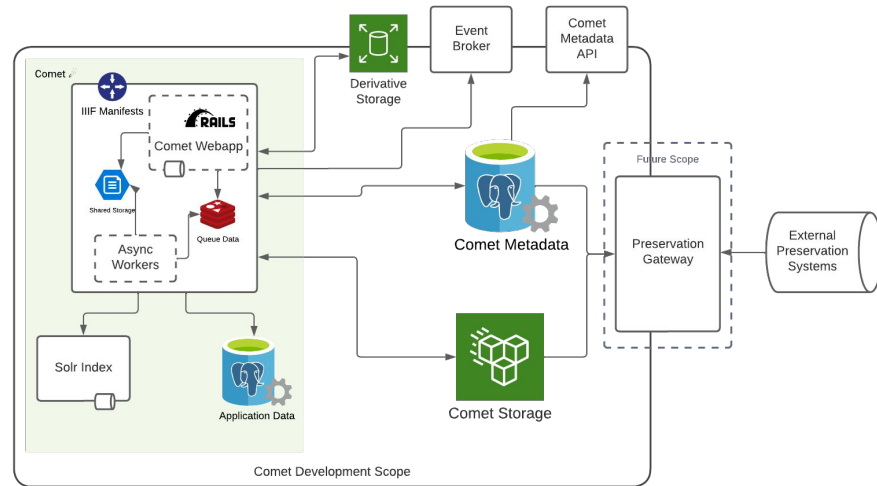
VALKYRIE + SEQUEL

Our work on sharing metadata between Comet and a separate API application wouldn't have been possible without **Valkyrie**. We use a PostgreSQL database as our metadata store, and the **Valkyrie::Sequel** gem for metadata access in both Comet and Superskunk. Combined with M3 and Hyrax ACLs, this gives us everything we need to meaningfully respond to API requests from outside Hyrax.

Credits

Vivian Chu & Matt Critchlow (UC San Diego)

Margaret Kibi & tamsin johnson (UC Santa Barbara)

The Library UC SAN DIEGO

UC SANTA BARBARA Library

samvera