VIVO/Vitro system architecture for linked open data regarding scholarship

HTTP

Ensures that only the VIVO/Vitro application, and not internal services such as Solr, are exposed to the public. Provides security filtering and a means to serve non-VIVO resources. This layer is optional, but recommended.

Presentation

Vitro UI

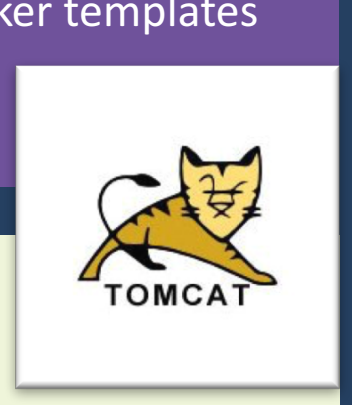
<#FREEMARKER>

VIVO Visualizations

VIVO UI Customizations

<#FREEMARKER>

Vitro provides a default web presentation for all entities. VIVO Freemarker templates override Vitro templates to provide presentation customized for scholarship. D3 is used to create viz that run on all modern devices.



Business Logic

Business logic and presentation services run as servlets in a Tomcat container

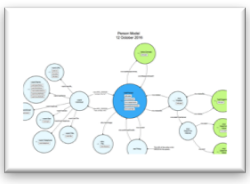
Simple Loader

Harvester

External applications load data through the Vitro APIs

User Access

Ontology Editor



Vitro APIs

SPARQL

Apache JENA

Reasoner

Jfact

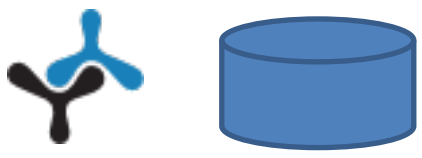
User access can be done with local credentials or external authentication services. An ontology editor supports creation of new ontologies, and management of classes and properties for ontologies loaded to Vitro. VIVO is pre-loaded with ontologies for representing scholarship. The Vitro APIs support SPARQL and LDF.

Persistence

Search Index

Content Triple Store

Configuration Triple Store



Vitro stores triples as named graphs in MySQL. Configuration info is stored as triples in the file system. Solr provides a search index and faceted search capability for Vitro and VIVO.

Michael Conlon and Graham Triggs, Duraspace http://duraspace.org