BPS Services Planning

Corpus loader

User Stories

1. Corpus service consumer can upload a zipped corpus file that will populate the raw corpus tables
2. User can import corpus into workspace
3. User can view list of documents in a corpus
4. For a given document, user can view name-role-activity instances

Service Stories

1. CRUD on Corpus, Doc, Activity, Role, etc.

Schemas

1. Corpus - name, owner (User, denormalized to force existence). Can Denormalize things like #docs, date range, etc.
2. Document - ID(s), date (orig string, and normalized date).
3. Activity - ID (unique), name, parentID (may be null, if not, this is a sub-genre). Note there is no Document-Activities, since we can get that from Name-Role-Activity-Doc with a select distinct /unique, or a group by to get counts.
4. Name - if an orthographic variant of another name, will include a link to normal form. Gender (male, female, unknown). Type: Person, Clan
5. Name-Role-Activity-Doc
6. Name-FamilyLink. Assertions in doc about familial relations. Relates a Name (patronym or clan name) to a Name-Role-Activity-Doc (i.e., an instance of a mention of a principle, in a document).

Modules

Testing

Workspace

Service Stories

Schemas

1. Workspace - owner (User, denormalized to force existence), public/private. Probably needs some other ACL mechanism.
2. Parameter - name, type?, description, range.
3. NameRefActions - subjectNameRoleActivityDocID, objectNameRoleActivityDocID, actionType (sameAs, notSameAs
4. FilterSpec - name, description, some specification to filter which corpus info is used for a graph. By Clan, by Gender, by Role, By Activity(genre)
5. NameRefGroups - allows for grouping nodes to build a graph.
6. Graph - name, notes, date created, filterSpecID.
7. Graph-Parameters - parameterID, value (float).
8. Person - name, gender, patronym (multiple?), clan, active age begin, active age end, comment
9. PersonCitations - personID, nameRoleActivityDocID.
10. Relation - type, weight, subject, object, graph

Modules

Testing

Graph-Builder

Service Stories

1. User can specify a Workspace context (corpus, filters, parameters) and get a family tree (an output Graph).
2. User can specify a Workspace context (corpus, filters, parameters) and get a social network (an output Graph).

Schemas

Modules

Testing

User Management, including Authentication/Authorization support:

Service Stories

1. User can sign in to system, and set up basic profile information. (done PLS 11-12-09)
2. Admin can create a new user. (done PLS 11-12-09)
3. Application can associate a new workspace to a new user.
4. Admin can set workspace-owner role for user on associated workspace.
5. Workspace-owner can allow other users access to owned workspace.

Schemas

1. User
2. Role
3. User-Roles
4. Permission
5. Role-Perms

Modules

Testing