Approval Workflow for SKOS or RDF Statements Maintenance

by Christophe Dupriez@destin.be

October 2010

Context

Keeping up to date a Thesaurus (or an Authority list or an RDF[1] Knowledge Base) requires to welcome the contributions of everybody but also to ensure regular reviews to achieve a coherent edifice built in full transparency.

Aligning thesauri (making equivalences between them) is also requesting a contribution / reviewing mechanism.

We propose to integrate this kind of contribution and reviewing process in ASKOSI[2] to support those needs. We think this support could be done with a rather generic "Workflow Manager for RDF Maintenance", something explored by different research teams (summary in http://www.semanticuniverse.com/blogs-http-patch-and-tracking-rdf-changes.html): we hope to find something readily done or a team investing in a similar direction with whom we could collaborate.

General Data Modelling

Persons are contributing, persons are reviewing, persons are managing: persons must be authenticated and authorized / empowered for specific tasks. **E-mails** may have to be sent periodically to remind persons about the waiting reviewing tasks.

Groups are a usual way to manage the authorisations to persons.

The data is made of **statements** (RDF statements):

1. about SKOS Concepts, Schemes or Collection (or any other RDF **subject**),
2. for a given property or sub-property (label, relation, note, etc.) (or any RDF **predicate**)
3. with a given value (term, "about" of a linked concept or scheme) (or any RDF **object**)
4. for a **language** (labels)

Contributions are:

1. <**addition**>: adding new statements (for instance, add a new altLabel to a Concept for a given language);
2. <**removal**>: removing existing statements (for instance, remove an altLabel to a Concept for a given language to add separately a replacing one);
3. <**correction**>: making very small corrections to existing statements (corrections invisible to the matching of existing value, e.g. a character case change.

[1] RDF: Resource Description Framework: Information is represented as a network of triples: subject, relation, object

[2] ASKOSI: Java tools for SKOS Concepts Definition and Usage management: http://www.askosi.org

Contributed RDF Statements can link SKOS Schemes from different sources (matching relations). Those sources can be internally controlled or imported from an external institution: the workflow system is then a repository for matching relations between sources, especially the ones which are not controlled internally.

Contributions must be reviewed and approved to become effective: a voting process must be supported:

1. A person either votes on all the statements for a given “subject” (a SKOS concept for instance) or votes on each statement separately.
2. A **vote** is an integer number that each allowed person can set (and change).
3. A person is limited on how negative/positive his/her vote can be.
   Rejection power limit can be different than approval power limit.
4. The voter can add a **comment** nearby its vote.
5. The votes eventually result in **approvals** or **rejections** of contributions based on a trigger point. The trigger point could be based on subject type, SKOS Scheme, etc.
6. Applications should be helped by an API[3] to apply dynamically the current approvals/rejections to the SKOS (or RDF) data image they have.
7. “**Publications**” freeze approvals/rejections and transfer permanently their results in the SKOS (or RDF) repository. Publications do not have to be very frequent.

Contributions have a **time-stamp** and, when approved, they are incorporated in chronological order.

OpenMetadataRegistry[4] SKOS **status** for a Concept (an RDF subject) provides the generic maintenance states of this “knowledge object”:

- ?New-Proposed (somebody contributed a new subject)
- ℹNew-Under Review (at least a vote has been recorded but no approval/rejection level has been reached)
- ✓Published (approval trigger has been reached)
- ?Change-Proposed (somebody contributed a statement on the subject)
- !Change-Under Review (at least a vote has been recorded but no approval/rejection level has been reached)
- ?Deprecate-Proposed (somebody propose to delete a subject)
- !Deprecate-Under Review (at least a vote has been recorded but no approval/rejection level of the deletion has been reached)
- ✖Deprecated (deletion approved but the subject is still used somewhere in the system)
- ⛔Not Approved (subject deletion has been approved or the initial proposition for a new subject has been rejected)

Those status would be calculated dynamically following contributions / votes on data statements.

[3] API: Application Program Interface

[4] Open MetadataRegistry: http://metadataregistry.org

API (Application Program Interface)

Like ASKOSI, we imagine this workflow manager as a Shared JAR for instant access by all applications of a J2EE container and as a Web Application for easy integration within existing applications.

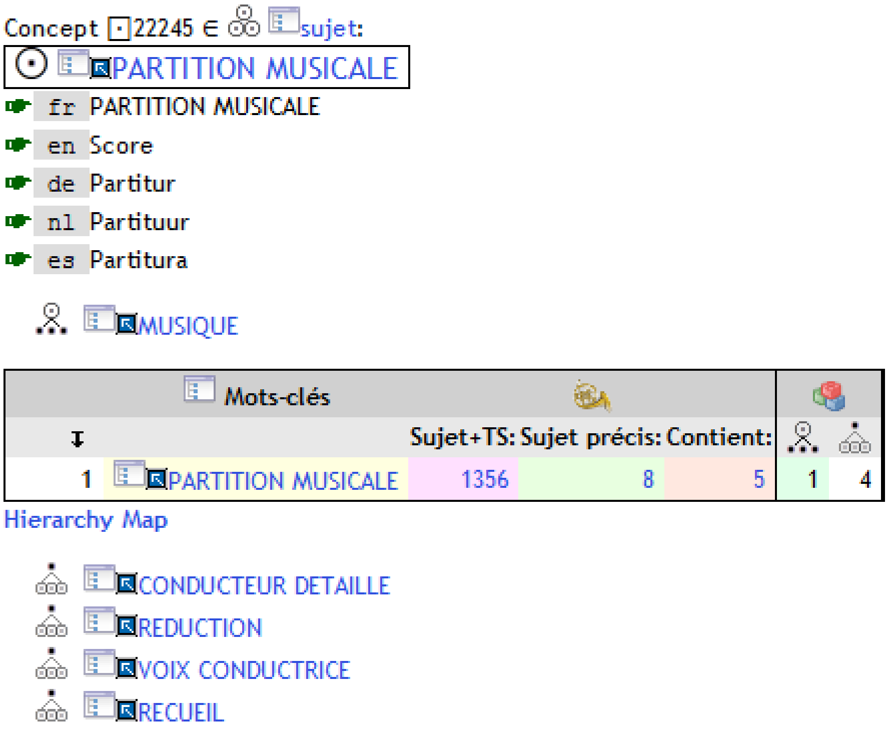
The Shared JAR should provide easy access to approved or pending statements for each Subject. It should also provide the HTML code inserted within a Web Page to add or display a Contributed Statement with the forms, buttons and input fields for voting or commenting it. Any application could also receive the NDSL status of a subject.

The J2EE Web Application should support the reviewing / voting / commenting process by:

- authenticating the users (using container authentication and Single Sign On);
- managing contribution and voting rights of users and groups;
- storage and retrieval of statements, votes and comments using an SQL database;
- listing pending / approved / rejected / archived statements and their linked subjects;
- calculating approvals and rejections;
- recording publications to archive approved and rejected statements.

User interface

ASKOSI currently displays a concept:



Concept 22245 ∈ sujet:

PARTITION MUSICALE

fr PARTITION MUSICALE

en Score

de Partitur

nl Partituur

es Partitura

MUSIQUE

	Mots-clés					
↧		Sujet+TS:	Sujet précis:	Contient:		
1	PARTITION MUSICALE	1356	8	5	1	4

Hierarchy Map

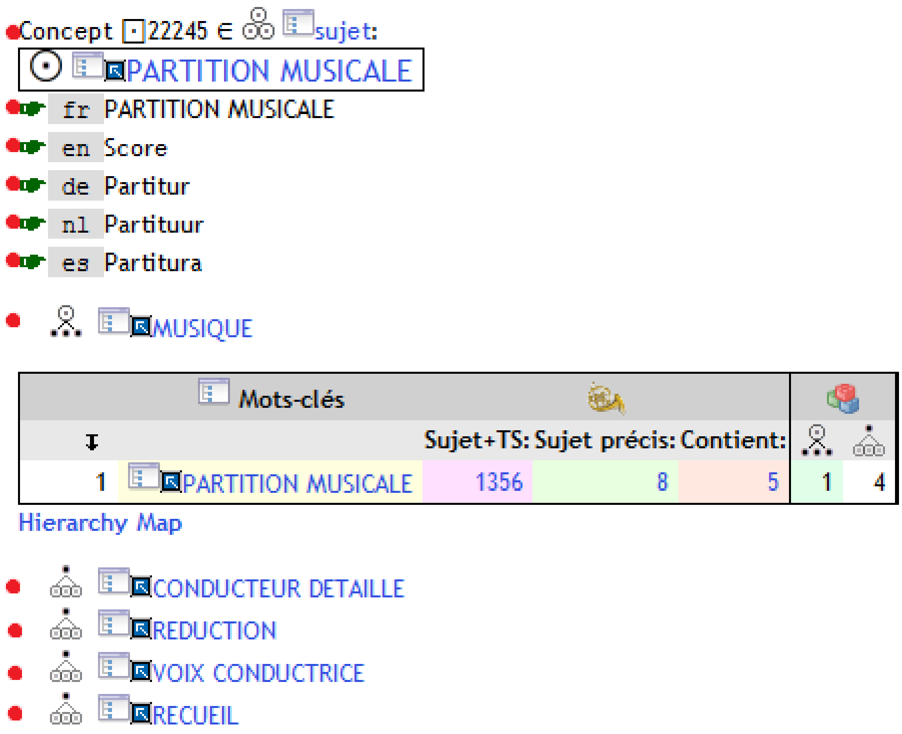
CONDUCTEUR DETAILLE

REDUCTION

VOIX CONDUCTRICE

RECUEIL

Let imagine correcting the character case of the French preferred label. We could log-in as a potential contributor and redisplay the concept using the “red dot” approach:



●Concept 22245 ∈ sujet:

○ PARTITION MUSICALE

- fr PARTITION MUSICALE
- en Score
- de Partitur
- nl Partituur
- es Partitura

● MUSIQUE

	Mots-clés					
↧		Sujet+TS:	Sujet précis:	Contient:		
1	PARTITION MUSICALE	1356	8	5	1	4

Hierarchy Map

- ● CONDUCTEUR DETAILLE
- ● REDUCTION
- ● VOIX CONDUCTRICE
- ● RECUEIL

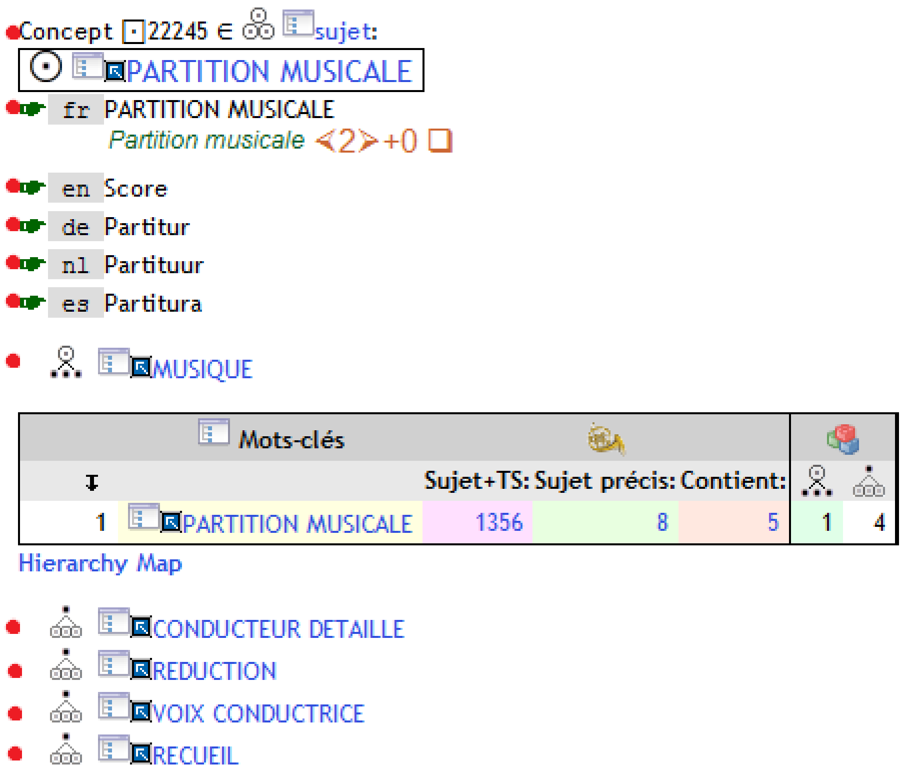
When passing with the mouse over a “red dot”, a contextual menu would propose the following actions:

- modify current occurrence
- add new occurrence
- remove current occurrence
- minor correction (character case change)

For all (except “remove”), a text box would appear to allow entering the new value (for links, this would be an “auto-complete” giving possible values). The new statement is then inserted in the display and buttons are added to allow voting:

- For a pending statement: ⮜2⮞+0 ❏
- For an approved statement: ⮜2⮞+6 ☑
- For a rejected statement: ⮜-1⮞-5 ☒

Arrows allow to modify user's vote. The next number is the total vote up to date. The colour and shape gives the approval/rejection status. An exclamation mark is added if comments are present: they are displayed (with time-stamps and other management information) whenever the mouse passes over it.



●Concept ⊡22245 ∈ sujet:

PARTITION MUSICALE

● fr PARTITION MUSICALE

Partition musicale <2>+0 □

● en Score

● de Partitur

● nl Partituur

● es Partitura

● MUSIQUE

	Mots-clés					
↧		Sujet+TS:	Sujet précis:	Contient:		
1	PARTITION MUSICALE	1356	8	5	1	4

Hierarchy Map

● CONDUCTEUR DETAILLE

● REDUCTION

● VOIX CONDUCTRICE

● RECUEIL

Other initiatives

In random order:

- The Open MetadataRegistry: http://metadataregistry.org/
- Talis.com ChangeSets
- GUO - Graph Update Ontology: http://webr3.org/specs/guo/
- Edutella: https://edutella.dev.java.net/