Editing, Tracking, Workflow

Overview

Documents editing, tracking of changes, and workflows.

Information about Editing and Tracking Change

The mapping tool supports a variety of actions to edit map records, add notes, assign advice, etc. All of the "map objects" (see Domain Model) are audited, meaning all changes to these objects are recorded with each commit. Thus assignment of a new code, movement through the worfklow, addition of advice, and any other changes are recorded. All prior states of map records can be searched and retrieved as needed.

Tracking Changes

There is no log of changes, per se, but each prior state of a map object can be viewed and read. In the "mapped concept view" in the application, the prior history of a map record is shown, with the last state of each map record for a particular concept (within the focus project) being displayed. Currently, there is no interface for seeing all of the incremental changes to a map record, though something like that would be fairly straightforward (but probably too fine-grained to be useful).

If editing causes any undesirable changes, admin tools can be developed to recover any prior state of a map record.

Note, prior states of objects not audited cannot be recovered. This is most important when considering changes in terminology content due to loading new versions, processing deltas (e.g. "drip feed") and tree position recomputation. This is intentional to improve performance and limit the number of archived states. Terminology objects include version information so auditing a change in version would cause an entire copy of the terminology to be recorded in the audit trails - this is the main reason they are not audited.

Origin ids

Map records undergo a complex workflow in which individual specialists edit their own copies of the record and submit them for comparison. When conflicts are detected, new records owned by the corresponding lead are created and at that time, all current states for all the records are available for review by the lead. A mechanism is used to connect the specialists' records to the lead record, called "origin ids". When a record is created from another record (or collection of records), the origin ids of that record and that record's id itself are added as origin ids of the later record. What this means is that for any workflow, records retain information about the prior records that were used ultimately to generate them.   Thus the chain of "provenance" is recorded and can be searched.

At the moment, there are no services to query or use origin ids in sophisticated ways, however the tracking of what records lead to what other records is faithfully preserved and can be reported on. Consider the following chain of events

- A new concept C1 enters the workflow.
- Two specialists each edit this concept leading to two map records M1 and M2.
- Upon comparison, suppose there is a conflict. This results in a lead-assigned map record M3 that has M1 and M2 as origins.
- Later, the lead resolves the conflict and the record gets submitted for publication.
- In this event, the specialists' records are removed (M1 and M2) and the lead's conflict resolution record (M3) gets a workflow status change to PUBLISHED.
- Now, say in a future editing cycle an error is discovered in the lead-resolved record and another lead edits this record on the "fix error" path.
- In this event a 4th record M4 is created and assigned to the editor and gets origin ids M1, M2, and M3.
- Upon being completed, the record is submitted for review where another record M5 is created and assigned to the lead reviewing the error correction. this record will have M1, M2, M3, and M4 as origin ids.
- Any subsequent changes will repeat the prior few steps, adding additional origins.   In this way, the complete history of the editing of this case can be tracked back and all information about workflow and changes can be recovered.

Workflow Concepts

Following are the standard "workflow paths" that a record can follow while it is being edited.

- Non-legacy path
  - For unmapped, in scope concepts where we have no information about starting records
- Legacy/Semi-automated path - UNIMPLEMENTED
  - For unmapped, in scope concepts where we have some information about starting records
- Fix error path
  - For previously mapped and marked as PUBLISHED or READY_FOR_PUBLICATION where an error in target code, advice, or other information has been identified
- QA path
  - For previously mapped and marked as PUBLISHED or READY_FOR_PUBLICATION that meet some QA criteria.
- Consensus management path - UNIMPLEMENTED
  - For non-legacy or legacy path cases that need consensus review because they are particularly complicated.

Workflow Tracking Record

This domain object represents the complete known state of the workflow, including information about:

- Assigned user count
- Assigned users and workflow statuses
- Concept terminologyId, terminology, version, and default preferred name
- Map project id
- Map records
- Workflow path
- Summary workflow status - an aggregation of statuses of the underlying map records.

Workflow Actions

The workflow model is built around actions that affect the various workflows in a variety of ways.

- ASSIGN_FROM_SCRATCH
  - Create and assign a record when no records for it exist yet.
- ASSIGN_FROM_INITIAL_RECORD
  - Assign a record based on an existing record (e.g. for when correcting an error)
- UNASSIGN
  - Return assigned work to the pool
- SAVE_FOR_LATER
  - Indicate that changes are in progress on a record.
- FINISH_EDITING
  - Finalize work on a record.
- PUBLISH
  - Send a finished record to publication
- CANCEL
  - Cancels an editing operation on a record with no other changes (e.g. puts a "fix error path" map record back to a PUBLICATION status for REVISION).
- CREATE_QA_RECORD
  - Used by workflows that need to create records that are not yet assigned (e.g. QA records)

Workflow Paths

The mapping tool supports a number of different workflows that have different actions and state transition diagrams. The currently supported (and planned) workflows include:

- NON_LEGACY_PATH
  - Used when dual independent review is desired and there is no candidate starting point for the map.
- LEGACY_PATH
  - Used when dual independent review is desired and there is a candidate starting point for the map.
- FIX_ERROR_PATH
  - Used for fixing errors among published or ready for publication records.
- REVIEW
  - Used when single review is desired and there may or may not be a starting point for the map.
- QA_PATH
  - Used for addressing reported QA errors
- CONSENSUS_PATH
  - Used for "consensus management" process - similar to non legacy path.

Workflow Status

Some workflow status values are reused by different paths and some are specific to certain workflow paths.

- CONFLICT_DETECTED - when a conflict between two EDITING_DONE records is detected, the EDITING_DONE records are set to this.
- CONFLICT_IN_PROGRESS - when a lead has started editing a conflict but not yet finished
- CONFLICT_NEW - created when a conflict is detected
- CONFLICT_RESOLVED - when a conflict is resolved but has not yet been sent to publication
- CONSENSUS_IN_PROGRESS - unimplemented
- CONSENSUS_NEEDED - unimplemented
- CONSENSUS_NEW - unimplemented
- EDITING_DONE - when a specialist clicks "Finished" on the editing screen
- EDITING_IN_PROGRESS - when a specialist clicks "save for later"
- NEW - when a specialist is assigned a record but has not made any changes yet.
- PUBLISHED - published record
- QA_IN_PROGRESS - when a specialist clicks "save for later" when editing a QA record
- QA_NEEDED - created when QA is put into the workflow
- QA_NEW - when QA work is assigned to a specialist
- QA_RESOLVED - when a specialist is finished editing a QA record (but not yet ready to send to publication)
- READY_FOR_PUBLICATION - when an edited/resolved record is sent back for publiction, the terminal status for editing
- REVIEW_IN_PROGRESS - when a lead has "saved for later" a REVIEW_NEW record.
- REVIEW_NEEDED - created when a specialist is finished working on a record in the REVIEW or FIX_ERROR path
- REVIEW_NEW - when a lead is reviewing a specialists' work that was done on the REVIEW or FIX_ERROR path

- REVIEW_RESOLVED - when a lead is finished work on a review or fix error record.
- REVISION - when a PUBLISHED record is being edited, it gets changed to this status
- READY_FOR_PUBLICATION - when comparison finds two specialists' records are equivalent, or when a lead resolves a conflict.

Workflow Path Documentation

The supported workflow paths are described in detail at the following links:

- Non-legacy Workflow Path
- Review Project Workflow Path
- Fix Error Workflow Path
- QA Workflow Path

The following workflow paths are not yet supported

- Consensus Path
- Legacy Workflow Path

References/Links

- Domain Model