

Mapping Tool Release Process

Overview

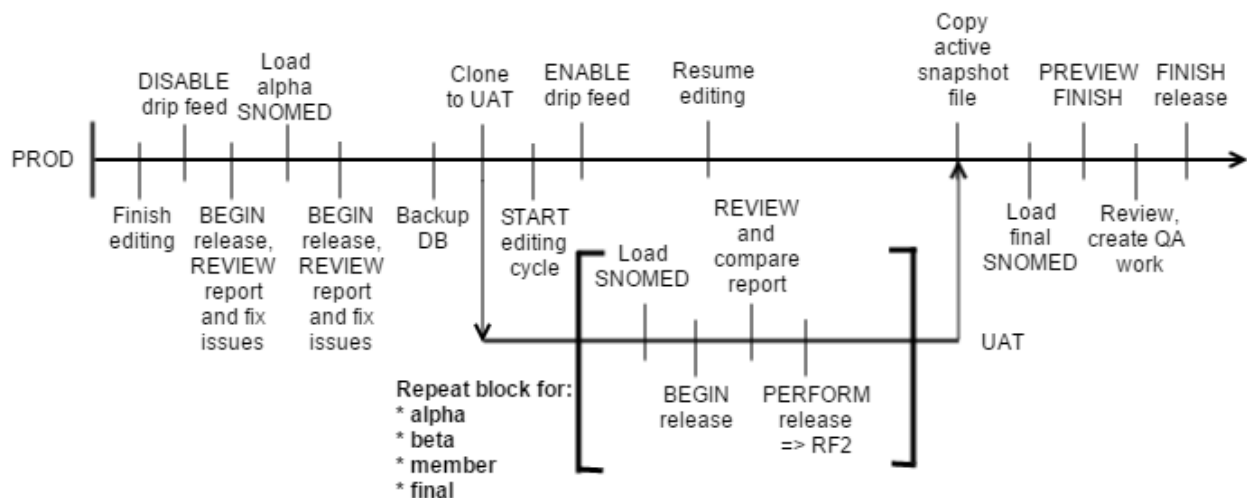
Documents the release process details. There is an admin page already that indicates how to run the mojo. See [Release Processing Tool](#) for more information.

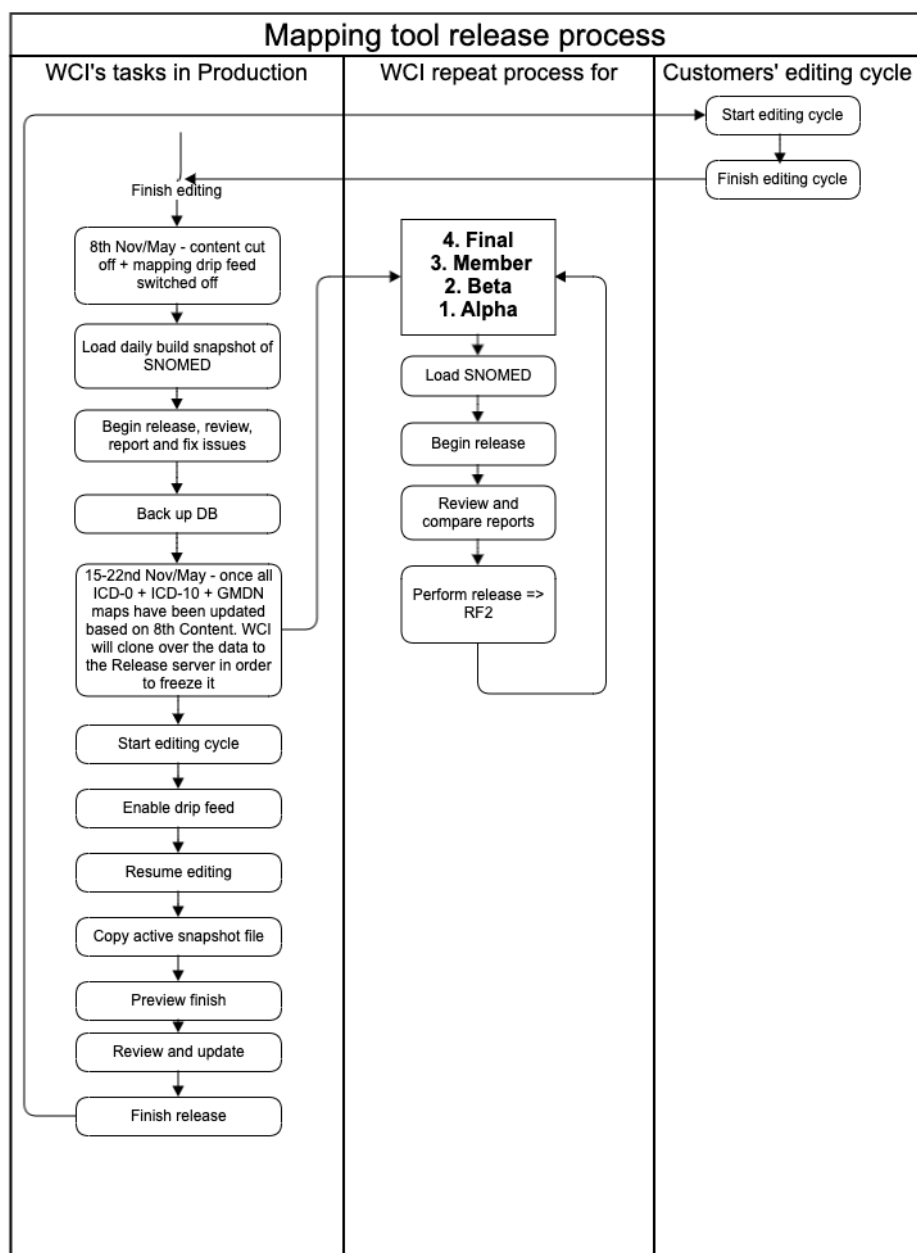
Assumptions

- Source terminology is SNOMED CT
- Release is in RF2 delta format
- QA of overall release is done by an external system (though project-specific validation rules are processed here)
- That editing was done in conjunction with a "drip feed" system that brought emerging content into the system as it was being developed, so it could be mapped.
- That the mapping files are part of the source terminology release

Timeline

The mapping tool supports a release process that takes into account end-of-editing cycle activities, some quality assurance, production of data files, and upgrading terminology versions when the time comes. The release processing is specifically tailored to the needs of IHTSDO and maintenance of the RF2 complex and extended map refsets that are part of the SNOMED CT distribution. (Note: in the below figure, all references to UAT are now handled on the RELEASE server)





The release process requires the ability for mappers to resume editing before the release cycle fully completes. Thus, this release strategy calls for completing the actual release processing on the RELEASE server and feeding back to PROD the final state of the map. Many of the steps can now be handled via the user interface, as described in this section [Running the Release Process with the User Interface](#). A more full accounting of the original manual steps can be found in the section below entitled [Mapping Tool Release Process](#).

Running the Release Process with the User Interface

[Mapping Release Process Demo Tutorial](#)





Release Actions

Pre-release steps - executed on production mapping server



- Compute Workflow** Recomputes workflow. Provided to check for available work prior to running release.
- Check Completion** Checks that all in-scope records are ready for publication. If 'Check Completion' runs clean, ready for WCI to clone to release server. Details available in Reports accordion, 'Project Dashboard' report.
- Start Editing Cycle** Sets the editing cycle start date for the project to today's date. Current value: 11/24/17 4:18 AM

Release processing steps - executed on release server after clone

Current source = SNOMEDCT, latest
Current destination = ICD10, 2016

- Reload SNOMEDCT** Reloads selected SNOMEDCT version and sets projects to it. 
Select version:
- Reload Refset Members** Reloads refset members from last release. 
- Begin** Creates 'Release QA', located on Reports accordion. Requires effective time. 
- Process** Creates release files. Requires effective time and module id. 

Finalize release steps - executed on prod mapping after release file is transferred by WCI

- Preview Finish** Previews results of finish using file generated by Process Release. Requires effective time. Expects release file to be present on server after WCI transfer. Creates 'Release Finalization QA' available on Reports accordion. 
- Finish** Updates map records and refsets from file generated by 'Process' step. Requires effective time. Expects release file to be present on server. 
Cannot be undone.

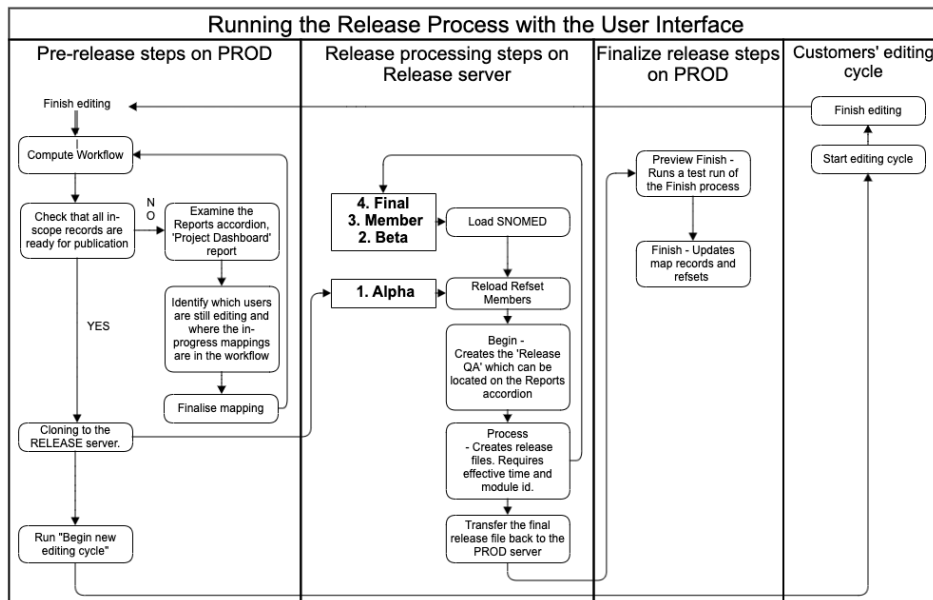
Parameters

Effective Time:

Module Id

Releases

Release file: 



Pre-release steps

These steps will be run on PROD.

- **Compute Workflow** - Recomputes the workflow. Provided to check for available work prior to running the release. New work will show up in the 'Available Work' widget on the main dashboard.
- **Check Completion** - Checks that all in-scope records are ready for publication. If this check runs clean, the release process can commence with WCI cloning to the RELEASE server. If this check does not run clean, details can be examined in the Reports accordion, 'Project Dashboard' report. This report will indicate which users are still editing and where the in-progress mappings are in the workflow. When the mappings are finalized, this check should be run again to confirm that work is complete. The night after work is confirmed to be complete, the nightly backup will run and the next day, WCI can use that night's backup to do the clone to the RELEASE server.
- **Start Editing Cycle** - After WCI has completed the clone to the RELEASE server, the new editing cycle should be started on PROD. Clicking this button, will update the editing cycle start date to today's date.

Release processing steps

These steps will be run on the RELEASE server after the clone is completed by WCI. For the alpha, beta, member, and final versions of the international release, ALL of these steps will need to be run for each iteration (with the exception of Reload SNOMEDCT, which does not need to be run for alpha). For the derivative releases, each of these steps will likely only be run once.

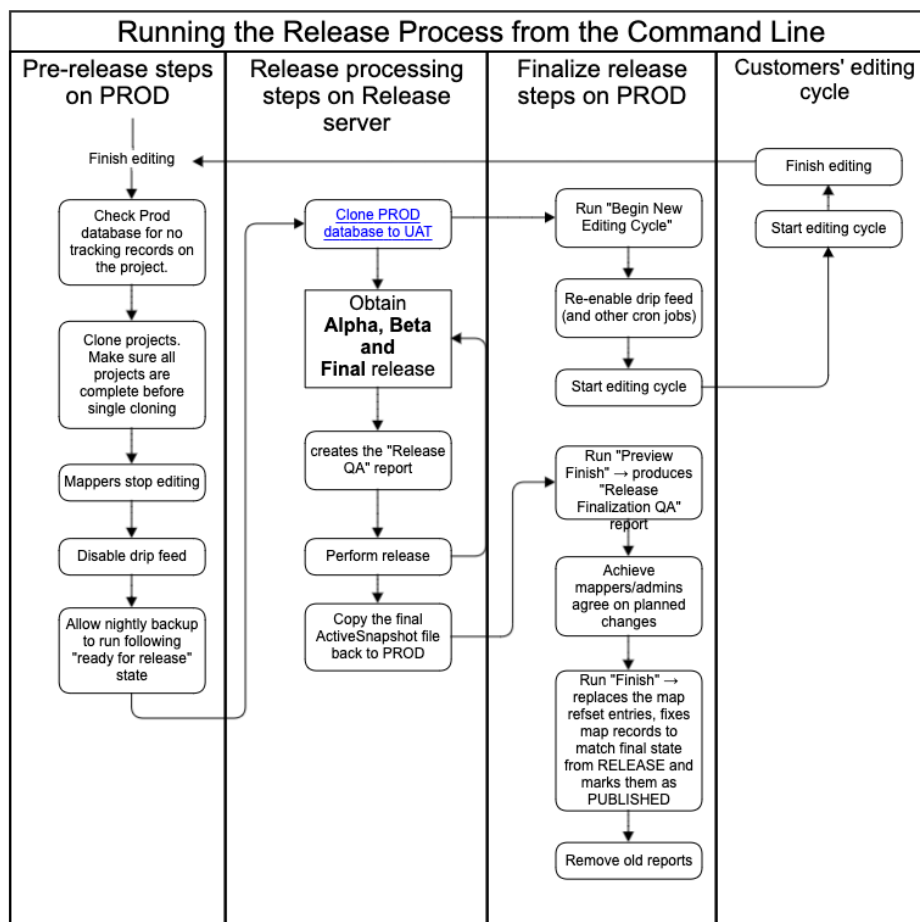
- **Reload SNOMEDCT** - For beta and later, select the SNOMED version that the release upon which the release will be based. After the version is selected, another picklist will allow selection of alpha, beta, member, or final releases of that SNOMED version. Options presented are those available on the AWS S3 instance. This process generally takes a few hours to complete. This step must be followed by running 'Reload Refset Members'.
- **Reload Refset Members** - Loads the refset members from the prior release of this project. This step is critical and enables the release process to compare what is currently in the db with what was included in the prior release. This step **MUST** be rerun every time that 'Reload SNOMEDCT' is rerun.
- **Begin** - Creates the 'Release QA' which can be located on the Reports accordion. Requires effective time to be entered in the Parameters section on the right-hand column of the UI. If it is established that mappers should make corrections based on this report, mappers should make these modifications on the RELEASE server. When the editing is complete, the 'Check Completion' step should be rerun and then the 'Begin' step can be rerun.
- **Process** - Creates release files. Requires effective time and module id. Files are available for download in the 'Releases' picklist on the right-hand column of the UI. When the administrator is satisfied with the release files, WCI will transfer the release file back to the PROD server so it can be integrated back into the current state of the project.

Finalize release steps

These steps will be executed on PROD after the release files have been copied back to the PROD server from the RELEASE server.

- **Preview Finish** - Runs a test run of the Finish process using the file generated by the 'Process' step. Requires the effective time to be entered in the Parameters section on the right-hand column of the UI. Expects release file to be present on server after WCI transfer. Creates 'Release Finalization QA' which is available on the 'Reports' accordion.
- **Finish** - Updates map records and refsets from the file generated by the 'Process' step. Requires the effective time. Expects release file to be present on server. This step **cannot be undone**. Make sure to analyze results of 'Preview Finish' to establish correctness before running this step. When running this step on the international release, it can take a few hours. In this case, it should be **run when others are not editing**, possibly overnight with backups and drip feed turned off so as not to interfere.

Running the Release Process from the Command Line



The following are the original details to produce a release with the command line interface and manually run mojos. It still contains relevant information for running the steps that are not available through the user interface such as cloning the PROD db to the RELEASE server and the return of the produced release files back to PROD.

1. PROD: Check Prod database for no tracking records on the project.
 - a. Currently, we are doing a single clone for all four projects (ICD10, ICDO, GMDN, and ICPC), so make sure all projects are complete before cloning.
2. PROD: Mappers stop editing.
3. PROD: Disable drip feed (and other cron jobs)
 - a. `crontab -e` (on the prod system)
4. PROD: Allow nightly backup to run following "ready for release" state
5. RELEASE: [Cloning PROD Database to UAT](#)
6. PROD: Run "Begin New Editing Cycle"
 - a. Done via the "Start New Editing Cycle" button in the "Release Handling" accordion of the "Project Details" page
 - b. This merely sets the editing cycle start date in the map project
 - c. Mappers can commence editing for the following editing cycle
7. PROD: Re-enable drip feed (and other cron jobs)
 - a. `crontab -e` (on the prod system)
8. RELEASE: Begin release (creates the "Release QA" report)
 - a. Done via the "Begin" button in the "Release Handling" accordion of the "Project Details" page
 - b. This produces a "Release QA" report which can be reviewed and fed into the QA workflow.
 - c. Ensure all in-scope concepts are mapped and there are no validation errors.
 - d. Double-check mapped, but out-of-scope concepts to ensure they should be out of scope.
 - e. Mappers fix any blocker errors on the RELEASE server
9. RELEASE: Obtain the alpha SNOMED release - `/home/ihstsd0/data/SNOMEDCT/$version/...snapshot Refset/Terminology` folders...
 - a. The Snapshot Refset/Terminology directories should be downloaded to `~/data/SNOMEDCT/$version` directory (replace any prior version data files)
 - b. Remove the old SNOMED version from the system (using the [Terminology Loader and Remover](#))
 - c. Add new SNOMED version to the system (using the [Terminology Loader and Remover](#))
10. RELEASE: For each: alpha, beta, final SNOMED Release
 - a. RELEASE: Obtain and load the (alpha, beta, final) SNOMED release
 - i. e.g. use `~/data/SNOMED/$version`, replace any older version data files, or archive them)

- b. RELEASE: Begin release (creates the "Release QA" report)
 - i. (see #2)
 - c. Perform release (into /home/ihtsdo/data/doc/release/\$release/\$destinationTerminology)
 - i. Done via the "Process" button in the "Release Handling" accordion of the "Project Details" page
 - ii. This generates the data files in ~/data/doc/release/\$version/\$destinationTerminology
- 11. PROD: After final RELEASE run for production SNOMED release, copy the ActiveSnapshot file back to PROD (into /home/ihtsdo/data/doc/release/\$release/\$destinationTerminology)
- 12. PROD: Run "Preview Finish" produces "Release Finalization QA" report
 - a. Done via the "Preview Finish" button in the "Release Handling" accordion of the "Project Details" page
 - b. This generates a "Release Finalization QA" report that summarizes changes between the map records in the release on PROD and what was generated on RELEASE
 - c. This process expects the active snapshot RF2 file in ~/data/doc/release/\$version/\$destinationTerminology
 - d. **SPECIAL NOTE:** If changes made in RELEASE caused concepts to be removed from scope, or maps to be created to empty targets (for simple map projects like GMDN), these changes will not be AUTOMATICALLY sync'd to PROD, you will have to manually remove the concepts from scope on prod otherwise they will persist into the future.
- 13. PROD: mappers/admins agree on planned changes
- 14. PROD: Run "Finish" replaces the map refset entries, fixes map records to match final state from RELEASE and marks them as PUBLISHED, and removes out of scope records.
 - a. Done via the "Finish" button in the "Release Handling" accordion of the "Project Details" page
 - b. This does 3 things
 - i. applies any changes made in RELEASE to the PROD copies of the map records (to avoid duplicate editing)
 - 1. Any records that will be changed are reported in the "Release Finalization QA" report.
 - ii. removes out of scope concepts/map records
 - iii. Marks "ready for publication" records as "published"
 - c. This process expects the active snapshot RF2 file in ~/data/doc/release/\$version/\$destinationTerminology
- 15. PROD: Remove old reports
 - When starting a new editing cycle, it may be useful to remove reports more than a year old to keep the overall size of accumulated reports reasonable.

```
service tomcat7 stop
# set date one year ago, e.g. 20141215
set refsetId=447562003
set date = 20141215
cd ~/code/admin/remover
mvn install -PReports -Drun.config=/home/ihtsdo/config/config.properties -Drefset.id=$refsetId -Dend.date=$date >&! mvn.log
service tomcat7 start
```

For each of the currently supported IHTSDO mapping projects, there is a project-specific page for enacting the specific steps needed for that project.

NOTE: paths shown above start ~/data/doc/..., however in principle everything is relative to the directory specified in "map.principle.source.document.dir" in config.properties (in practice, this is set to "/home/ihtsdo/data/doc" in the PROD deployment. We already had a property pointing to a local file system directory so it was reused. TODO: in the future, this property should probably be renamed.

Creating Custom Release Processing

If the process above is not needed or desired, alternative release processing can be built as a custom development task. Things to consider are:

- The release format
- Whether the release is part of the source/target terminologies of the map
- Whether the mapping versioning itself is coordinated with the source/target terminology versioning
- Whether map records were built completely from scratch or were generated from terminology data
- Whether a "delta" release is needed - e.g. a release that compares to the previous release and publishes only the changes.

These things are very project specific and are not the same for all projects.

Custom release processing can be implemented via an outside package, or in the mapping-custom project (see [Custom Release Process](#)).

References/Links

- n/a