

MovingComponents

This wiki page is meant to document the action items when a component changes *state*.

Commons Component Lifecycle States

As far as the Commons repository is concerned, a Commons component can be in **one and only one** of the following states:

- proper ([current list](#))
- sandbox ([current list](#))
- dormant ([current list](#))

Pre-requisites

- Any transition where the origin or destination is Commons Proper, requires a VOTE.
- Revival from dormancy requires a VOTE (as documented in the "Revival" section of the [dormant components web page](#)).
- Thus effectively the only move that can occur by lazy consensus is the periodic [sandbox pruning](#) for dormant components.

So, **make sure you have called a VOTE and it has passed.**

You will need commons karma.

Sandbox graduations

Transitions from sandbox to proper are particularly significant, and have a number of additional tasks associated with them. The appropriate wiki page to consult for sandbox graduations is [MovingFromSandboxToProperSVN](#).

SVN move

For a component that changes state from <origin> to <destination>:

1. Remove component from svn:externals from the trunks-<origin>
2. Move component
3. Add component to svn:externals in the trunks-<destination>

The above sequence makes sure that component is always available when listed in the particular svn:externals. The actual SVN commands can be looked up either from the [MovingFromSandboxToProperSVN](#) or [ProposalSandboxPruning](#) wiki pages.

Tagging

Tag the component once it has been moved to its new location in SVN. The recommended tag name is:

<component>_<origin>_mm_yyyy

Example: A component *foo* that has been moved out of sandbox in January 2006 chooses a tag FOO_SANDBOX_01_2006 after the move.

POM update

Update the POM to reflect the correct website URL for the new component state.

Website changes

1. Publish the component website at its new location
2. Make changes to the Commons website (look in commons-build/menus and commons-build/parts) before re-publishing. This bit will change when we move to a Maven2 site.
3. Optionally, remove old website and/or permanently redirect from the old website location to new one

The sequence is important to avoid potential 404s.