

RFC: Git workflow

Regular development

Develop documents using DM's standard workflow. That is:

1. Create a ticket branch from `master`.
2. Work on the ticket branch.
3. Have the ticket peer-reviewed.
4. Rebase and merge the ticket branch to `master`.

Merging to `master` does not denote acceptance by a CCB. Instead, it adds to the changeset that will be included in the next CCB review.

Submitting a document to the CCB

Establishing a new baselined version of the document begins with submitting it to the CCB. The release manager is responsible for orchestrating this process on behalf of DM.

1. Choose a commit on `master` where the document's content is ready for CCB review.

2. Create a DocuShare tag at the commit. DocuShare tags are formatted as `docushare-vN`, where **N** is the next-available document version number in DocuShare.
3. Upload the PDF built by CI for the `docushare-vN` tag to DocuShare. By definition, this uploaded document should become version **N** in DocuShare.
4. For LDM documents, create a DM RFC. For project documents (LSE), create an LCR.
5. Once the RFC or LCR number is known, create a branch from the `docushare-vN` tag named `tickets/RFC-N` or `tickets/LCR-N`. This release branch initially has no new commits on it.

Addressing CCB feedback

The CCB may request amendments before the document can be baselined.

1. Create a ticket branch from the `tickets/RFC-N` or `tickets/LCR-N` release branch.
2. Commit amendments to that ticket branch and peer review.
3. Rebase and merge the ticket branch onto the `tickets/RFC-N` or `tickets/LCR-N` release branch.
4. Multiple ticket branches can be created and merged to the release branch if the CCB's requests are being addressed by multiple people working on separate parts of the document.

5. The release manager creates a new DocuShare tag (**docushare-vN**) from the head of the release branch, uploads the revised document to DocuShare, and notifies the CCB.

Releasing a new baselined version of the document

When the CCB baselines (accepts) the new document, the project librarian designates the baselined document with a semantic version. The DM release manager is responsible for releasing a document.

1. From the head of the release branch (**tickets/RFC-N** or **tickets/LCR-N**), add a commit that:
 - Updates the semantic version of the document to the one designated by the project librarian.
 - Updates the change history table to reflect the new version.
 - Fixes the document's date to the date of baselining.
 - Removes the "draft" watermark (for example, by removing **draft** from the document class's arguments).
2. Create a new DocuShare tag (**docushare-vN**) and upload the document to DocuShare.
3. Once the project librarian has set that DocuShare version as the preferred version, create a semantic version tag (formatted **v<major>.<minor>**) at the same commit as the corresponding DocuShare tag.

4. Create a new ticket branch from the **head** of the **master** branch. Cherry pick amendment commits from the release branch onto that ticket branch. Also include a commit that updates the document change history table. Merge that ticket branch to **master**.

Notes:

- The release branch is never merged back to **master**. Amendments get back to **master** through cherry picking commits from the release branch.
- Development is allowed to happen on the **master** branch while CCB review is simultaneously happening on a release branch. This means that the DM release manager is responsible for properly addressing conflicts while cherry picking amendments back to **master**.

Hotfixing a baselined document

It may be necessary to release a minor update to a baselined document (to fix typos, for example). This is done by creating a branch based off the prior release branch. Hotfixes cannot be made from **master** because significant edits may have been merged since the document was baselined.

1. Create a ticket branch from the semantic version tag of the prior release.
2. Commit fixes to that ticket branch and have the changes peer reviewed.
3. Create a DocuShare tag (**docushare-vN**) and upload the document to DocuShare.
4. Create an RFC or LCR proposing the changes to the CCB.

5. Once the RFC or LCR number is known, create a new release branch (`tickets/RFC-N` or `tickets/LCR-N`) from the `docushare-vN` tag created in Step 3.
6. The remaining process of releasing this document, upon CCB approval, is the same as in *Releasing a new baselined version of the document*.