Dynamic Model

Version Control

# Introduction

The dynamic model team uses [Git](https://git-scm.com/) for version control along with the [SourceTree](https://www.sourcetreeapp.com/) Git desktop client.

Every dynamic model team member has two dedicated directories—a Development directory located on the team’s local shared drive and a Production directory located on the team’s HPCC network drive. Both directories contain Git repositories that can be managed using SourceTree.

The Development directory is used for active development work. Its repository has visibility into all other Development repositories. The Production directory is used for production runs of the dynamic model. Its repository only has visibility into its paired Development repository.

This guide will cover:

[I. Getting Started](#_I._Getting_Started_1)

How to get started working on the dynamic model.

[II. Making a Commit](#_II._Making_a_1)

How to commit changes you make to the dynamic model.

[III. Performing a Merge](#_III._Performing_a_1)

How to merge changes from another version of the dynamic model.

[IV. Managing Branches](#_IV._Managing_Branches_1)

Best practices for managing your Development repository.

[V. Doing a Production Run](#_V._Doing_a_1)

How to do a production run of the dynamic model.

# I. Getting Started

To get started working on the dynamic model:

1. Click **Fetch** in SourceTree to fetch all commits (i.e. dynamic model versions) made by the team. This will update your Development repository.

Your Development repository is the first top tab in SourceTree; it should be selected by default. The second top tab is your Production repository, which is used for production runs, as described in [Doing a Production Run](#_V._Doing_a_1).

1. Identify the commit off of which you would like to base your work. This is the version of the dynamic model you will start from.
2. Create a new branch to track your work by right clicking on the target commit, selecting **Branch…**, specifying a name, then clicking **Create Branch**.

You can give your branch a generic name such as *working* or a descriptive name such as *parameter\_update*. Branches are simply pointers to commits (i.e. versions) and are easily created, deleted, and renamed.

Alternatively, if you already have a branch pointing to your target commit and would like to use that branch, make the branch your active branch by double clicking on it. SourceTree indicates your active branch with an empty circle to the left of the branch name.

You are now ready to work on the dynamic model from your Development directory.

# II. Making a Commit

To commit changes you make to the dynamic model:

1. Flip to the **File Status** bottom tab in SourceTree.
2. Review the changes you’ve made in the **Unstaged files** panel then click **Stage All** to include the changes in your commit.

Use **Stage Selected** to only include changes made to select files.

1. Write a commit message in the text box above the bottom tabs using the following format:

<Short description beginning with a past tense verb>

<Longer description>

For example:

Updated policy parameters.

New parameters make use of latest data and discussions with experts. Parameter generation functions and parameter files have all been updated.

For small commits, the newline and longer description can be omitted.

1. Click **Commit**.

You can confirm that your commit was made successfully by flipping back to the **Log / History** bottom tab. Your active branch should now be pointing to your new commit.

Your team members will be able to see your new commit when they fetch from your repository.

# III. Performing a Merge

To merge changes from another version of the dynamic model into your active branch:

1. Commit any changes to the dynamic model you’ve made, following the steps in [Making a Commit](#_II._Making_a_1).
2. Right click on the commit (i.e. the version) you’d like to merge changes from, select **Merge…**, then click **OK**.

If the selected commit is a direct descendent of the commit your active branch is pointing to, your active branch will simply be relocated to the selected commit. This is known as a fast forward merge and is a common way to update your active branch with more recent commits, such as those made by your team members.

1. If merge conflicts are found, review these by flipping to the **File Status** bottom tab. All files with conflicts will be listed in the **Unstaged files** panel.
2. Resolve any merge conflicts, click **Stage All**, then click **Commit**. If no conflicts were found, this step will be completed automatically by SourceTree.

By default, fast forward merges do not produce new commits.

1. Test the new merged version of the dynamic model for functional compatibility.

Remember that the merge process only checks for structural (e.g. textual) conflicts. The absence of structural conflicts however does not guarantee the absence of functional conflicts.

Your active branch now includes the merged changes.

# IV. Managing Branches

It is best practice to only keep branches in your Development repository that you are actively working with.

You will always have a *master* branch. This branch should be used to indicate commits that you feel are ready for production. Your team members will do the same, thus allowing you to track their production-ready work by observing the progress of their *master* branches.

To move your own *master* branch so that it points to a new commit, first double click on your *master* branch to make it your active branch then merge a target commit following the steps in [Performing a Merge](#_III._Performing_a_1). This will often be a fast forward merge.

Note that you can also work with your *master* branch directly as with any other branch. However, it is recommended that this be reserved for relatively simple commits, such as cosmetic updates, that are unlikely to impact the production readiness of the dynamic model.

All branches other than your *master* branch are considered working branches and should be deleted when no longer actively developed.

To delete a working branch, make another branch (e.g. your *master* branch) your active branch, right click on the branch to be deleted in the left-hand **Branches** listing, then select **Delete**.

If there are commits associated with the branch that are not associated with any other existing branch, you will need to check **Force delete** before clicking **Delete**. Note that this will discard those commits along with the branch.

In general, use as many branches as you need to track the various branches of your development work—no more, no less.

# V. Doing a Production Run

To do a production run of the dynamic model:

1. Flip to your Production repository by selecting the second top tab in SourceTree.
2. Click **Fetch** to fetch all commits from your Development repository.
3. Identify the commit representing the version of the dynamic model you’d like to run. Move the *run* branch to the target commit by right clicking on the commit, selecting **Reset current branch to this commit**, selecting **Hard** from the dropdown menu, then clicking **OK**.

The *run* branch should be the only branch in your Production repository and will be the active branch by default.

1. Log in to the HPCC and run the dynamic model from your Production directory.

Results from the production run will be saved to the Output directory under the DynamicModel parent directory, tagged with a commit identifier.