

## Git Tutorial

**backlog** + **git**  
by nulab

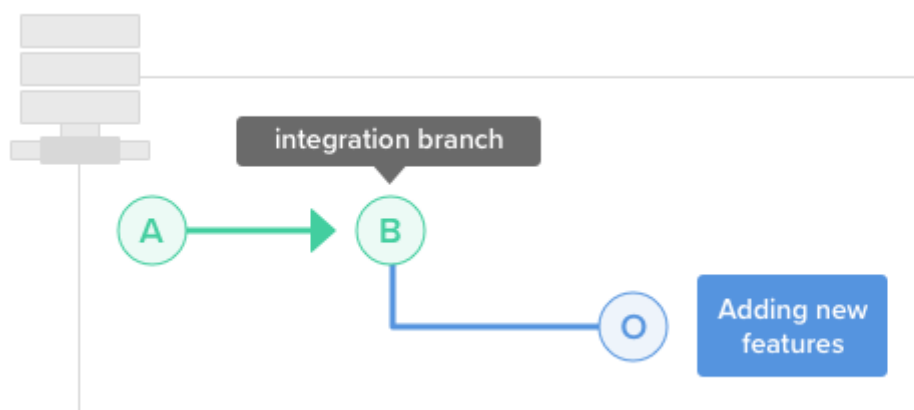
Backlog lets you manage projects with Git integration

Learn more →

### Sample feature branch workflow

You will need to create a branching strategy that works best for your team, of course. But here is a quick example of how to follow a branching strategy workflow involving two types of branches: a develop/integration branch and a feature/topic branch.

Let's say you are working on a new feature. Suddenly, somebody finds a bug in the production and you are now tasked to fix that bug parallel to working on the new feature.

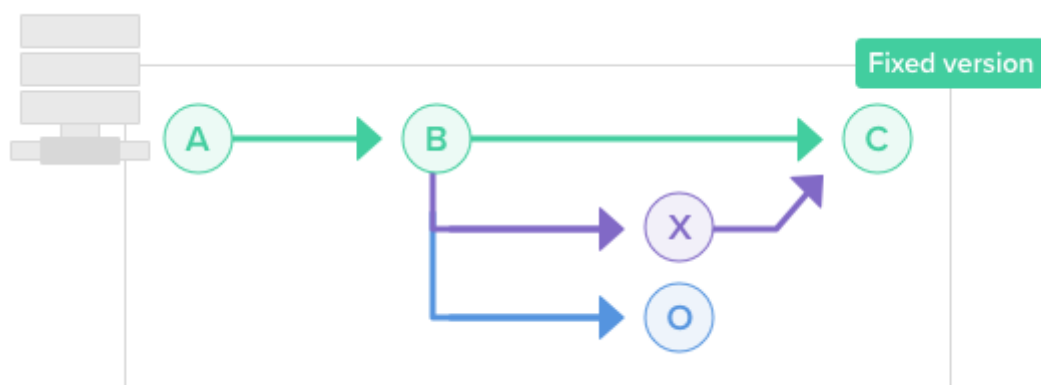


Before starting on the bug fix, you can create a new branch off of the develop/integration branch. This new branch will isolate the bug fix from the new feature that you were working on.

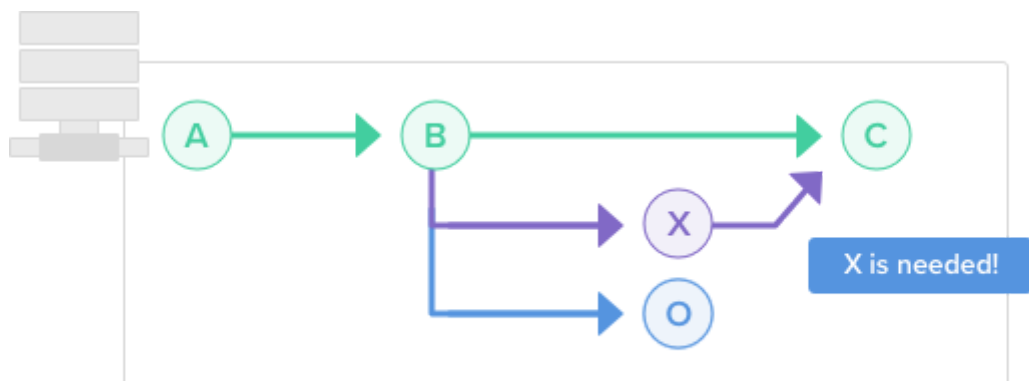
## Git Tutorial



When you are ready to release the bug fix, merge the bug fix feature/topic branch into the develop/integration branch.



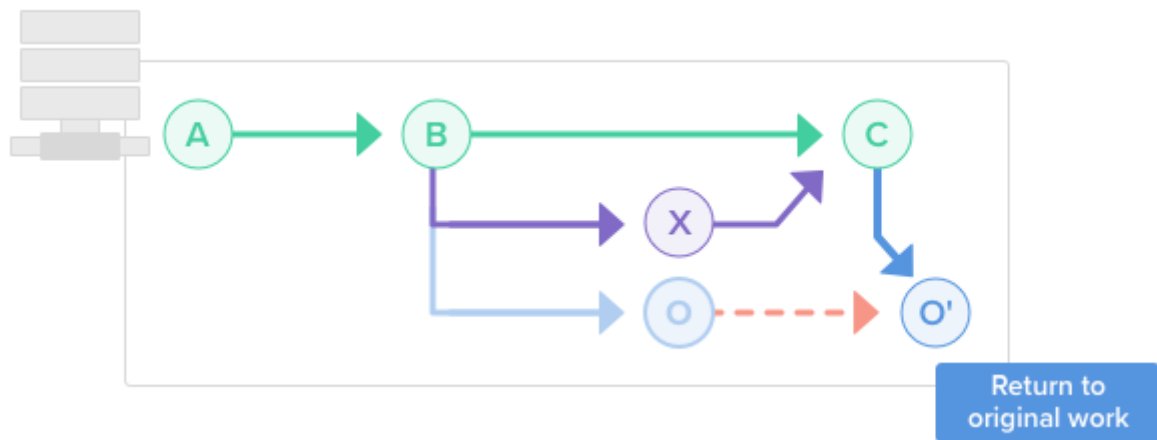
Then you can switch back to your original feature/topic branch and continue working on the new feature.



On your feature/topic branch, you will notice that commit "X", which is the bug fix commit, is needed to continue implementing the new feature. In other words, you will have to synchronize your current branch with the changes on the develop/integration branch.

There are two options to go about doing this. The first is to merge the develop/integration branch that includes commit "X" with the current branch.

## Git Tutorial



Once you commit "X" on our current branch, you can safely proceed to work on your new feature.

[Prev](#)[Next](#)

**Backlog has built-in Git repositories for every project**

Try it free

Follow us on

[English](#)[Terms](#)[Privacy](#)

© 2021 Nulab, Inc. All rights reserved.