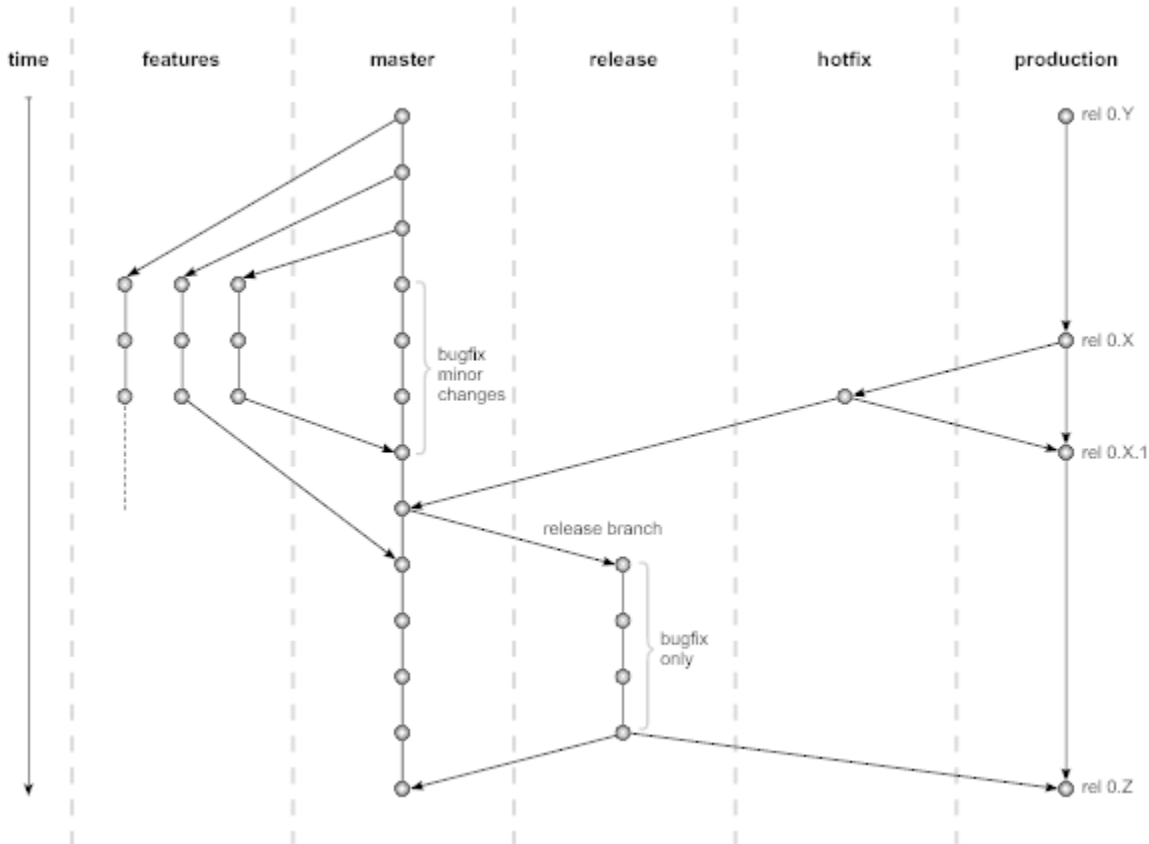


Development in branches

In the KIELER project we use Git. We like Git. We want to get the maximum out of Git. We want branches, many branches.

Branches?

Yes, branches! This shows the KIELER branching scheme. If you want to know whatever the different classes of branches do, check [Source Code Management](#).



Why and how to use branches

Now you should know which use the different kinds of branches have, if not, go back to [Source Code Management](#).

Ok, now nearly everything should be clear. We want to implement everything in features. The master branch should always be release-ready. For developers this means:

1. Break down development into small features. Implement large Features incrementally in smaller Features
2. Make a feature branch for each feature you develop
3. Regularly merge changes from the master branch into your feature branch
4. Keep track of your development tasks in Jira, it helps others to know what you are working on

When is my feature considered as done?

So you took heed to the branching guidelines. There is a little more, you do not know yet when your feature is ready to be merged back into the master branch:

1. All planned functionality must be implemented and working
2. Push your branch to the mainline so that Bamboo can build it
3. The code has to be well commented; especially Javadoc comments have to be complete
4. The code must not contain any warnings, including FindBugs and Checkstyle warnings. If single warnings cannot be avoided, leave a comment in the source
5. The code should be well covered by unit tests
6. The code should have been reviewed

But my feature is soooooo tiny, do I always have to open a new branch?

Well, at least you should. If your feature is really small it is ok to push it directly to master. Remember, do not break the build!