

# Git Structure

## Question

How does the Git Structure work? When I look at the "Branches" of MediaPortal 1 on Github, I will see something called "Master" sitting on top and a lot of branches below. What is the relationship between the Master and the various branches?

## Background

## Answer

### Master

The master is the base or "stable" branch. It represents our next release

Branches will only be merged with the master once they are fully tested, confirmed to be working, documented and approved to get merged/included in the next release.

### Branches

For every new feature, rework or bugfix we use a separate branch. While this does look like a lot of extra work, this is actually very easy and fast when using GIT. This allows the team to more closely follow release schedules, because only stable code will be allowed to merge with the master.

Also developers are no longer stalled by feature and code freezes. They can just continue their work in their branch.

Generally speaking there are four different kinds of branches:

- **BUG-<Jira Ticket>-<Description>** - Such a branch is about a bugfix. The Jira ID clearly links it to the issue in Jira.
- **FEAT-<Jira Ticket>-<Description>** - This branch is a new feature. The Jira ID clearly links it to the issue in Jira .
- **RELEASE** - Such a branch is used for all maintenance releases for a major release. i.e. Release\_1.2.x stores the stable code of all 1.2.x maintenance releases.
- **EXP-<Description>**- These are experimental branches which are used to try out ideas. Such are unclear if they ever reach a state where they can be integrated into a major release.

### Commits

For every commits done inside dedicated branch, common branch or Master branch, the commit text must include :

- **<Jira Ticket> - <Description>** - The Jira ID must be the 1st text in commit message, for Jira Web App can detect it.