Using git to develop AMT

May 10, 2010

1 Rationale

Git is a version control system developed by Linus Thorvalds to maintain the Linux kernel. It is a distributed version control system scalable to extremely large projects, because there is not central repository. In Git, every developer can have his/her own repository.

Git has been choosen for the AM toolbox, because *branching* in Git is very easy. This allows each developer to work without disturbing the others, because each sub-project is developed in its own branch.

Even though each developer has his own repository, there *is* a main one, namely the one of sourceforge. This means that we don't using Git in the intended way, using multiple repositories, but instead use a little like SVN with properly working branches!

2 Structure of the AMTOOLBOX Sourceforge repository

The sourceforge repository has several branches. The idea is that you only work on specific things in specific branches.

- The main branch is called *master*. Files in this branch are the ones that gets uploaded by a file release. Don't do *any* new development work on this branch, use it only to fix bugs. Development are done in other branches, which are then merged into master when the development is finished.
- All development work takes place on separate branches. Create a branch for your work, push it to sorceforge for backup and for other people's comments, and then merge it into master when you are done.

Git works by having both *local* and *remote* branches. You must connect your local branch to a remote branch in order to track it. For the sake of less confusion, I suggest to use the same names for the local and remote branch.

3 Git on Windows

To use Git on windows install `msysgit` and `TortoiseGit`

4 Getting the code

To get the reposity, you need to clone the Sourceforge one:

```
git clone ssh://soender@amtoolbox.git.sourceforge.net/gitroot/amtoolbox/amtoolbox amtoo
```

On Windows, right-click in where you want the directory, choose "Git clone ..." and then enter the URL above.

This will create a repository `amtoolbox`, which is related remotely to the Sourceforge repository. After this command, change to the `amtoolbox` directory for all further operations.

After this operation, you will only have obtained the `master` branch. To see the other remote branches, type

```
git branch -r
```

To see you local ones, type just

```
git branch
```

On Windows, both these task can be done in the right-click menu TortoiseGit -> Checkout/Branch.

To get the code from any of the remote branches, type

```
git branch --track devel origin/devel
```

This will connect the local branch to the remote branch, both named devel.

On Windows, again use the "TortoiseGit -> Checkout/Branch" menu, but remember to check the "Create New Branch" box if you are getting a remote branch for the first time. Choose the same name (i.e. `devel`) as the remote branch.

To switch to this branch use

```
git checkout devel
```

The checkout command is always used to switch between branches.

5 Working with the code

To commit, use `git commit`, this is similar to SVN or CVS, only difference is that you only commit to your local tree, and not to Sourceforge. You must *add* your changes before they can be commited, then command `git commit -a` is very usefull for this.

To upload your changes, use `git push`, and to get new updates, use `git pull`.

6 Getting more help

More help on `git` can be found online, e.g. git book