Belle II - Git migration

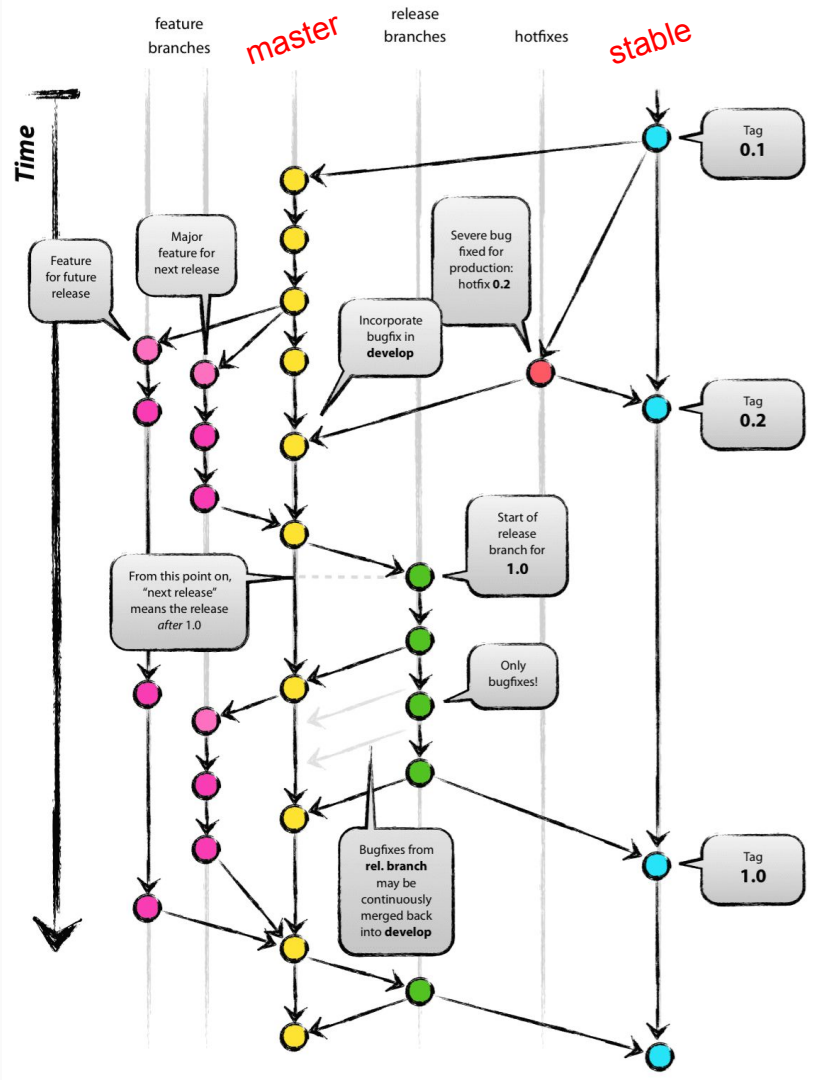
Why git?

- Stash GIT service managed by DESY
- Powerful branching and merging capabilities
  - Resolution of (JIRA) issues directly be map to branches and commits
  - Feature freeze in pre-release
- Allows code publication before integration
  - Librarian reviews / mentoring
  - No more “look at this patch” emails
- Encourages smaller commits
- Conceptually better handling of external resources (with git subtree)
- Developers **love** its powerful tooling
  - (>50% of the commits to the main software are made with git svn)
- Industry / Open Source community standard
  - ROOT, CMS, Python, Genfit2, FastBDT, ... migrate to git
  - Emerging tools based on it

Workflow considerations

- SVN has a centralised workflow with one “branch” -> generally *master*
- **Same style should be allowed** in git for contributors that do not (want to) rely on branching
- Individuals / working groups may adopt a feature branch model
  - Favoured is **Git flow** model
  - Feature branches merged by **pull requests**
  - Approval by responsible **librarians**
- Releases branch from *master*
  - Finish release with pull request to *stable*
  - Approval by **all librarians**

GIT Flow model



Status

Setup of test git repository on the DESY Stash system

https://stash.desy.de/projects/B2

documented in

https://belle2.cc.kek.jp/~twiki/bin/viewauth/Software/GitRepository

Currently three parts of the SVN repository cloned to experiment with:

- git clone ssh://git@stash.desy.de:7999/b2/tools.git
- git clone ssh://git@stash.desy.de:7999/b2/externals.git
- git clone ssh://git@stash.desy.de:7999/b2/software.git

Allowed early access with public key auth for all twiki users.

Experimental Bamboo build service

Bamboo: Build service well integrated with Stash / git

- Big plus : Allow branch builds for feature / release branches
- Managing and sharing build / validation artefacts (e.g. compiled externals, docs, plots…)
- Automatic email notifications
- REST API triggers etc.

Experimental build plans in https://bamboo.desy.de/browse/B2

- Need more and specialised build agents for
  - Scientific Linux 5 / 6, 32-bit systems, compilations with clang / icc

Caveats

Social

- Learning curve
- Workflow changes
- Require librarians to integrate and review code
- Branching and merging confusion
- Commit / publish hurdle
  - Three steps to publish
  - Stage, commit, push
  - Aid with scripts?

Technical

- Splitting the repository
- Unpacking SVN externals
- Rewrite of managing scripts
- Enforcing of code policies
  - Coding styles
  - Commit rights to folders
- Package tags / release
  - Release creation from tagged folders in earlier commits

Social change management

Get all parties involved to agree on the code repository

Collect

- Feedback
- Ideas
- Concerns

as early as possible

Splitting the repositories

- The big svn serves multiple purposes
  - Main software
  - Admin scripts
  - Grid software
  - Belle II notes
  - User analysis
  - Group repositories
- Necessary split into separate git repositories
- Requires in depth understanding of the content
  - **Difficult for group repositories featuring**
    - nested git repositories
    - SVN externals repositories
- Start has been made with main software tools / externals / software

SVN externals - Genfit2

Problem:

- SVN repository links against SVN external repositories

Solution:

- Inline the source from the external repositories
- Done for the Genfit2 external resource

Prospect:

- Use git **subtree** or git **submodule** to track and commit to the linked repository on github/Genfit

Package tags

Problem:

- Package tags mark subfolder <-> GIT tags mark the whole repository

Solution:

- Translated the SVN tags to the commits in the master branch

Remaining problems:

- How to combine releases from package tags

Code policies

Use git hooks on both:

- Server side
- Client side

Possible solution:

- Introduce server hooks with plugin
- Needs activation of by DESY IT
- Local hooks with special setup for the user repository