

LHCb migration from Subversion to Git

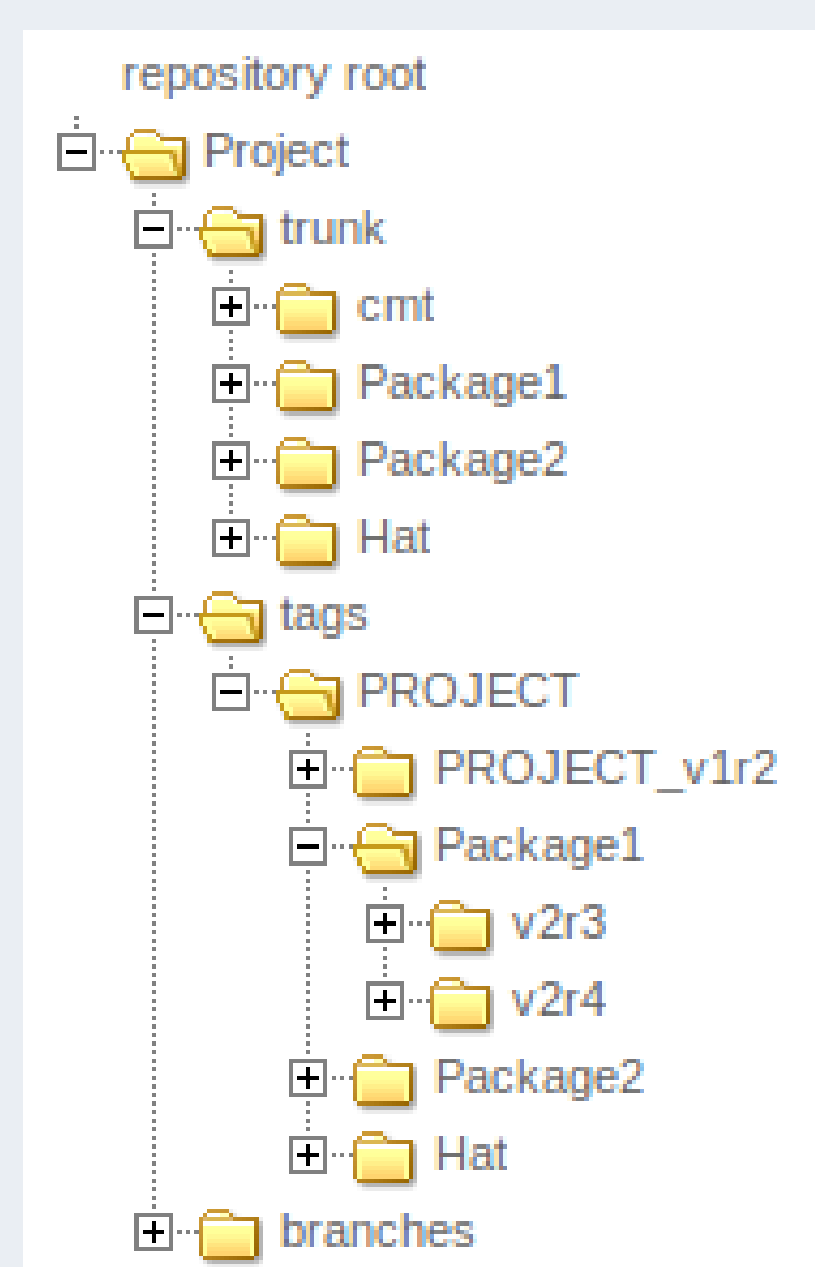
M. Cattaneo, M. Clemencic, J. Closier and B. Couturier on behalf of the LHCb Core Software Team

Abstract

Because of user demand and to support new development workflows based on code review and multiple development streams, LHCb decided to port the source code management **from Subversion to Git**, using the **CERN GitLab hosting service**. Although tools exist for this kind of migration, **LHCb specificities** and development models required careful planning of the migration, development of migration tools, changes to the development model, and redefinition of the release procedures. Moreover we had to support a **hybrid situation** with some software projects hosted in Git and others still in Subversion, or even branches of one project hosted in **different systems**. We present how we addressed the special LHCb issues, the technical details of **migrating large non standard Subversion repositories**, and how we managed to **smoothly migrate** the **software projects** following the schedule of each project manager.

Strong demand by developers for Git support in LHCb, but...

- LHCb Software is organized in interdependent projects
- Common libraries, applications (simulation, reconstruction, analysis, ...)
- Projects subdivided in packages
- Special tool to checkout all packages in a project release
- Custom layout of Subversion repository
- One repository with many projects
- Tags and branches created at package level



Requirements for the migration

- ✓ Smooth/adiabatic transition
- ✓ Integration with LHCb Nightly Builds
- ✓ Support development with partial checkout
- ✓ Definition of development workflows

Development on partial checkouts

Implemented a couple of **Git subcommands** to allow **sparse checkouts of multiple projects** in one Git managed workspace/repository and push local history of changes to the main repositories.

Integration with LHCb Nightly Builds

Extended Nightly Builds checkout scripts to allow checkout from Git and **automatically apply** relevant **GitLab merge requests** to the checked out branch.

Definition of development workflows

Moved from a workflow based on direct commits to SVN, to a workflow based on production and **development branches**, and **merge requests** with **code review**.

Summary

The migration of LHCb Software Projects from Subversion to Git started in March 2015 with the first mirrors of a few projects.

Early 2016, we defined development workflows and prepared the required custom tools.

At present, only a few special cases have not migrated yet. Allowing users to develop with Git **increased productivity** (number of contributions, new projects) and **code quality** (testing and reviews).

Transition

