



Contribution ID: 180

Type: Oral presentation to parallel session

## Experiences with moving to open source standards for building and packaging

*Thursday, 17 October 2013 14:38 (22 minutes)*

The LCMAPS family of grid middleware has improved in the last years by moving from a custom build system to open source community standards for building, packaging and distributing. This contribution outlines what improvements were made and the benefits they rendered.

LCMAPS, gLExec and related middleware components were developed under a series of European framework program projects, starting in 2001 (Datagrid) and lasting more than a decade (EGEE-I, -II, and -III, with EMI and IGE ending in 2013).

Many interdependent components were made by these projects; as building and packaging of the ever growing collection of software became more complex, so did the build system that was used to manage it. Eventually ETICS was developed which became the central build system for all of EGEE-III and EMI.

Building outside ETICS proved hard, which raised concerns about long term sustainability and adoption outside the original community, for instance by OSG.

In the past years we upgraded the software we maintained to become independent of any build system, using common patterns in popular open source software as our template. The result is that any package can be downloaded in tarball form, configured to find dependencies on the local system, and build to install in a custom location. But our software is also packaged according to the rigid guidelines of the Fedora and Debian projects, and easily deployed on systems running Debian, Ubuntu, Red Hat (with EPEL) or Fedora.

The required adaptations were not all trivial, but all were worthwhile in retrospect. They helped to uncover several bugs that would have gone unnoticed and offered additional unanticipated benefits.

The adaptations cover the areas of version control, adoption of standards (especially POSIX and ANSI C), backward and forward compatibility, dependency resolution, use of GNU autotools, developing build and release procedures, packaging for EPEL and Debian, automated builds with Koji, and repository generation.

Some of the added benefits are improved portability, fewer bugs, increased serviceability, improved sustainability, a closer match with common open source skill sets, easier configuration, easier installation, easier packaging, shorter release cycles, and inclusion in mainstream distributions.

In this presentation we share our experiences concerning this transition.

**Primary author:** Mr VAN DOK, Dennis (Nikhef (NL))

**Co-authors:** Dr SALLE, Mischa (Nikhef (NL)); KOEROO, Oscar Arthur (Nikhef (NL))

**Presenter:** Mr VAN DOK, Dennis (Nikhef (NL))

**Session Classification:** Software Engineering, Parallelism & Multi-Core

**Track Classification:** Software Engineering, Parallelism & Multi-Core