

Software compatibility check framework for grid computing elements

Description of content and intended audience - the outcome you expect to achieve.

The target audience of this work are developers of software to be executed on the grid, system administrators and devops operators in charge of putting into production new software packages on the grid.

The goals of this work are basically:

- share our recent experience of our group related to the deployment of new software to be used on the grid and determining the compatibility between software and computing elements.
- receive feedback and suggestions about possible improvements for the proposed framework. There are several aspects of our proposed architecture that would benefit from some feedback from grid operators.

Relevant URL (if any)

<http://bioinformaticslaboratory.nl/twiki/bin/view/EBioScience/CvmfsPublic>

Preferred Day if any (Demos - Mon, Tue, Wed)

Wednesday

Printable summary: this is the only section of the abstract that will be published in the Book of Abstracts.

The proposed framework for software compatibility checking on grid computing elements (CEs) based on gLite middleware derives from the experience at the Academic Medical Center of the University of Amsterdam (AMC, NL) using the Virtual Organisation (VO) VLEMED on Dutch gLite resources of the European Grid Infrastructure.

The jobs submitted by VO VLEMED are based on a heterogeneous set of software for data analysis, which management on the CEs involves several challenges: library incompatibilities, lack of integration in package management systems, binary architecture and version conflicts, etc.

To improve the success rate of submitted jobs, we designed and implemented a framework to check software compatibility on grid CEs. This framework relies on predefined test jobs and their expected results.

CernVMFS, a read-only network file system optimised for software distribution, combined with the Environment Modules tools, overcomes many of the problems associated to the software distribution on the grid described above.

The grid job submission is based on Ganga, a Python-based front-end for job management and submission. Each software to be tested requires the definition of a Ganga grid job, including input data and scripts to validate the results from the CEs.

Finally, Nagios has been used to publish the compatibility results obtained by the framework. The cron daemon and several shell scripts provide automation features.

Primary author: FONT, Juan Luis (EGLEU)

Co-authors: Dr LEE, Hurng-Chun (Radboud University); JAGHOORI, Mohammad Mahdi (Academic Medical Center, UvA); OLABARRIAGA, Silvia (Academic Medical Center of the University of Amsterdam)

Presenter: FONT, Juan Luis (EGLEU)