



Contribution ID: 346

Type: oral presentation

The GridPP DIRAC project - Implementation of a multi-VO DIRAC service

Thursday 16 April 2015 12:00 (15 minutes)

The DIRAC INTERWARE system was originally developed within the LHCb VO as a common interface to access distributed resources, i.e. grids, clouds and local batch systems. It has been used successfully in this context by the LHCb VO for a number of years. In April 2013 the GridPP consortium in the UK decided to offer a DIRAC service to a number of small VOs. The majority of these had been previously accessing grid resources using the EMI-WMS system and some of them were also using the EMI-LFC for their data management. DIRAC provides integrated workload and data management systems, although these can also be used independently if required.

It was decided that the most maintainable way to deploy the DIRAC server was with one instance shared between all supported VOs rather than a separate one for each VO. While this deployment model was not without precedent, this approach greatly deviates from the model for which DIRAC was originally designed. We set-up a test instance and while using this it was found that a number of core features needed to be improved so that they could be managed within the scope of a VO rather than server-wide. The main features that needed this adaptation were the access controls, the automatic configuration of users & resources and the data management functions.

The DIRAC client toolset is fully-featured and contains analogues to most of the lcg client commands. For the VOs which already have wrappers around the lcg tools, just adapting these would be a possibility however this would still leave every VO with their own solution to maintain. New VOs don't want to spend valuable computing effort to develop custom wrappers for their users based on these tools; a better solution is to try and standardise on a set of common tools. This reduces both the maintenance effort for any single user as bugs get fixed for everyone at once while also minimising the number of different configurations the central experts need to be familiar with. The primary tool we have investigated for this role is Ganga, particularly with its DIRAC backend module.

We report on how we resolved these issues to provide a production grade multi-VO DIRAC service, including where appropriate the details of changes applied to the DIRAC and user interface code bases.

Authors: HUFFMAN, Adam (Imperial College Sci., Tech. & Med. (GB)); Dr RICHARDS, Alexander John (Imperial College Sci., Tech. & Med. (GB)); BAUER, Daniela (Imperial College Sci., Tech. & Med. (GB)); Dr COLLING, David (Imperial College Sci., Tech. & Med. (GB)); RAND, Duncan (Imperial College Sci., Tech. & Med. (GB)); MARTYNIAK, Janusz (Imperial College London); Dr CURRIE, Robert Andrew (Imperial College Sci., Tech. & Med. (GB)); FAYER, Simon (Imperial College Sci., Tech. & Med. (GB))

Presenter: Dr CURRIE, Robert Andrew (Imperial College Sci., Tech. & Med. (GB))

Session Classification: Track 4 Session

Track Classification: Track4: Middleware, software development and tools, experiment frameworks, tools for distributed computing