**Ganga** an Optimiser and Front-End for Grid Job Submission

Enabling Grids for E-science

---

**Scope**

Ganga is an easy-to-use frontend for job definition and management, implemented in Python. It is being developed to meet the needs of the ATLAS and LHCb experiments for a Grid user interface, but offers possibilities for extension and customisation that make it interesting for a wide range of user communities. ATLAS and LHCb will investigate various aspects of particle production and decay in high-energy proton-proton interactions at the Large Hadron Collider (LHC), due to start operation at the European Laboratory for Particle Physics (CERN), Geneva, towards the end of 2007. Both experiments will require processing of data volumes of the order of petabytes per year, and will rely on computing resources distributed across multiple locations. The experiments’ data-processing applications, from event modelling to physics analysis, are based on the Gaudi/Athena C++ framework. Ganga (Gaudi/Athena and Grid Alliance), deals with configuring these applications, allows simple switching between testing on a local batch system and large-scale processing on the Grid, and helps keep track of results.

**Job representation**

A job in Ganga is constructed from a set of building blocks. All jobs must specify the software to be run (application) and the processing system (backend) to be used. Many jobs will specify an input dataset to be read and/or an output dataset to be produced. Optionally, a job may also define functions (splitters and mergers) for dividing a job into subjobs that can be processed in parallel, and for combining the resultant outputs. Ganga provides a framework for handling different types of application, backend, dataset, splitter and merger, implemented as plug-in classes. Each of these has its own schema, which places in evidence the configurable properties.

**Other projects using Ganga**

In addition to its use in ATLAS and LHCb, Ganga is finding application in a growing number of other projects, both as a standalone tool and as submission engine and monitoring system for the DIANE job-distribution framework. The Ganga user groups include:

- high-energy physics experiments: Compass, Harp
- teams developing detector simulations: Garfield, Geant4
- biomedical projects: med-auston, projects researching Avian flu
- projects involving image processing: Cambridge Ontology, Xmipp
- chemistry project: WIEN2k
- telecommunications industry: International Telecommunication Union (ITU)

---

**User view**

Ganga classes and functions can be used interactively at a Python prompt, can be referenced in scripts, or can be used indirectly via a Graphical User Interface (GUI). The GUI consists of a main panel, which displays job information, and three dockable windows - windows that can be hidden, integrated in the main panel, or floated separately. The dockable windows provide for job building, for execution of arbitrary Ganga/Python commands, and for the logging of system messages.

Ganga tutorials for ATLAS and LHCb have been held at locations including Bologna, Cambridge, CERN, Edinburgh, Lyon, Milan, Mumbai and Munich, and online guides to getting started with Ganga are also available. Tutorials and guides have been key factors in stimulating Ganga take-up, with more than 500 Ganga users recorded.

---

**Grid searches for avian flu cure**

Use of the Grid to search for drugs against avian flu was widely reported. About one eighth of the Grid jobs were submitted using Ganga/DIANE.

---

**The Grid improves efficiency of Geant4 production cycles**

Regression tests for Geant4 major releases require a few years of CPU, concentrated in a short time. These tests are run on the Grid using Ganga/DIANE.