JRA1 Contribution to CHEP07

Report of Contributions

https://indico.cern.ch/e/14232

Contribution ID: **0** Type: **not specified**

Building a robust distributed system: some lessons from R-GMA

Monday 3 September 2007 08:00 (20 minutes)

[Oral]

R-GMA, as deployed by LCG, is a large distributed system. We are currently addressing some design issues to make it highly reliable, and fault tolerant.

In validating the new design, there were two classes of problems to consider: one related to the flow of data and the other to the loss of control messages. R-GMA streams data from one place to another; there is a need to consider the behaviour when data is being inserted more rapidly into the system than taken out and more generally how to deal with bottlenecks. In the original R-GMA design the system tried hard to deliver all control messages; those messages that were not delivered quickly were queued for retry later. In the case of badly configured firewalls, network problems or very slow machines this led to long queues of messages, some of which were superseded by later messages that were also queued. In the new design no individual control message is critical; the system just needs to know if each message was received successfully.

The system should also avoid single points of failure. However this can require complex code resulting in a system that is actually less reliable.

We describe how we have dealt with bottlenecks in the flow of data, loss of control messages and the elimination of single points of failure to produce a robust R-GMA design. The work presented, though in the context of R-GMA, is applicable to any large distributed system.

Session Classification: Information Contributions

Contribution ID: **1** Type: **not specified**

Job Submission and Management Through Web Services: the Experience with the CREAM Service

Monday 3 September 2007 08:00 (20 minutes)

[Oral presentation]

Modern GRID middlewares are built around components providing basic functionality, such as data storage, authentication and security, job management,
resource monitoring and reservation. In this paper we describe the Computing Resource Execution and Management (CREAM) service. CREAM provides a Web service-based job execution and management capability for Grid systems; in particular, it is being used within the gLite middleware. CREAM exposes a Web service interface allowing conformant clients to submit and manage computational jobs to a Local Resource Management System. We developed a special component, called ICE (Interface to CREAM Environment) to integrate CREAM in gLite. ICE transfer job submissions and cancellations from the Workload Management System, allowing users to manage CREAM jobs from the gLite User Interface. This paper describes some recent studies aimed at measuring the performance and the reliability of CREAM and ICE, also in comparison with other job submission systems. We discuss recent work towards enhancing CREAM with a BES and JSDL compliant interface.

Presenter: Mr ZANGRANDO, Luigi (INFN Padova)

Session Classification: Resource Access, Accounting, and Brokering Contributions

Contribution ID: **2** Type: **not specified**

Tools for the management of stored data and transfer of data: DPM and FTS

Monday 3 September 2007 08:00 (20 minutes)

[presentation]

As a part of the EGEE project the data management group at CERN has developed and support a number of tools for various aspects of data management:

A file catalog (LFC), a key store for encryption keys (Hydra), a grid file access library (GFAL) which transparently uses various byte access protocols to access data in various storage systems, a set of utilities (lcg_utils) for higher level operations on data are all supported. However, in this presentation we will focus on giving an overview of two components in particular:

A disc pool manager (DPM) which provides a service to coordinate the storage of files across discs. The DPM features POSIX ACLs on files and pools, file lifetime with garbage collection, optional replication of data within the DPM and authorization based on VOMS grid certificates. The DPM offers an SRM interface, versions 1.1 and 2.2, along with its own control interface. Access to data is supported via gsiftp, rfio and an optional xrootd module is also available.

A file transfer service (FTS) allows the replication of data from one data store to another. The FTS features individually configurable, unidirectional management channels. The channels allow allocation of parameters such as number of concurrent transfers, number of parallel streams or TCP buffer size. SRM (version 1.1 or 2.2) is used to send requests to the storage systems. Third party gridftp or SRMCopy initiated transfers are supported.

Presenter: SMITH, David (CERN)

Session Classification: Data Management Contributions

Contribution ID: **3** Type: **not specified**

File Transfer Service

Monday 3 September 2007 08:20 (20 minutes)

[paper]

A key feature of WLCG's multi-tier model is a robust and reliable file transfer service that efficiently moves bulk data sets between the various tiers, corresponding to the different stages of production and user analysis. We describe in detail the WLCG transfer service, both the tier-0 data export and the inter-tier data transfers, discussing the transition and lessons learned in moving from a reliable software product to a full production service based on that software. The focus is upon the deployment and operational experience of the service gained during the 2006 and 2007 experiment production activities and dress rehearsals. We discuss the software and operational features that have been deployed to meet the reliability and performance needs of the service, and integration of the service with the WLCG and experiment operations.

Presenter: MCCANCE, Gavin (CERN)

Session Classification: Data Management Contributions

Contribution ID: **4** Type: **not specified**

DPM Status and Next Steps

Monday 3 September 2007 08:40 (20 minutes)

[paper/poster]

The DPM (Disk Pool Manager) provides a lightweight and scalable managed disk storage system. In this paper, we describe the new features of the DPM.
It is integrated in the grid middleware and is compatible with both VOMS and grid proxies. Besides the primary/secondary groups (or roles), the DPM supports ACLs adding more flexibility in setting file permissions.
Tools to interact with the DPM at different levels have been extended so that site managers can more dynamically configure and manage their DPM in a consistent way. In addition to rfio and gsiftp, users can now use the xrootd and https protocols to access the DPM.
A new version of Storage Resource Manager (SRM) interface, v2.2 has been implemented. One of the novelties is the reserve space concept, useful to guarantee space for a specific user or a group during a given period of time.
DPM has been deployed in roughly 80 Tier-2 sites and in several medical institutes. Unlike physics data, medical data is very sensitive. The DPM will offer the possibility to encrypt data throughout the process in a very secure way by implementing a key-distributed system.
Performance has been improved by the use of bulk queries. Stressing tests have shown a good robustness of the DPM against concurrent accesses.

Presenter: ABADIE, Lana (CERN)

Session Classification: Data Management Contributions

Contribution ID: **5** Type: **not specified**

Recent Developments in LFC

Monday 3 September 2007 09:00 (20 minutes)

[paper/poster]

The LFC (LCG File Catalogue) allows retrieving and registering the location of physical replicas in the grid infrastructure given a LFN (Logical File Name) or a GUID (Grid Unique Identifier). Authentication is based on GSI (Grid Security Infrastructure) and authorization uses also VOMS.
The catalogue has been installed in more than 100 sites. It is essential to provide consistent and user-friendly tools to manage the catalogue. The LFC is based on a hierarchical namespace with a POSIX interface. The LFC API is similar to UNIX commands and includes functions to start/end a session or a transaction. The support of secondary groups and ACLs (Access Control Lists) allow a flexible management of file permissions.
An automated recovery strategy based on a retry mechanism offers a better reliability. Accessed very often by many tools, the catalogue needs to guarantee a fast response time. Performance issues have been studied and tuned by implementing bulk queries. Several other tests are being conducted such as the impact of the size of the communication buffer between the client and a local/remote LFC server on the response time.

Presenter: ABADIE, Lana (CERN)

Session Classification: Data Management Contributions

Contribution ID: **6** Type: **not specified**

GFAL and LCG-Util

Monday 3 September 2007 09:20 (20 minutes)

[paper/poster]

GFAL, or Grid File Access Library, is a C library developed by LCG to give a uniform POSIX interface to local and remote Storage Elements on the Grid. LCG-Util is a set of tools to copy/replicate/delete files and register them in a Grid File Catalog.
In order to match experiment requirements, these two components had to evolve. Thus, the new Storage Resource Manager interface, SRM v2.2, is now supported.
In addition to that, important requirements were to have a python interface to GFAL/LCG-Util, and to fully support Logical File Name (LFN) at GFAL level.
Data privacy is a very important issue for some people. Therefore, we have to integrate Hydra client into GFAL. It allows to manage encrypted files and the cryptographic keys. Others important topics of development are to optimize the number of requests to BDII, and also to provide Perl API to GFAL and LCG-Util.

Presenter: MOLLON, Remi (CERN)

Session Classification: Data Management Contributions

Contribution ID: **7** Type: **not specified**

Medical Data Management Status and Plans

Monday 3 September 2007 09:40 (20 minutes)

[paper/poster]

The goal of the Medical Data Management (MDM) task is to provide secure (encrypted and under access control) access to medical images, which are stored at hospitals in DICOM servers or are replicated to standard grid Storage Elements (SE) elsewhere.

In gLite 3.0 there are three major components to satisfy the requirements: The dCache/DICOM SE is a special SE, which encrypts every requested image with a file specific key. It does not provide a storage area on its own, but interfaces a hospital's DICOM server to the grid. The gLite I/O server with a Fireman catalog service provides the access control by wrapping an SE, which holds medical images. And finally Hydra client library does the en/decryption of the files, using the file specific keys stored in the Hydra keystore.

In gLite R3.1 we are planning to simplify the software stack by relying on richer functionality of the underlying components: as storage elements (for example DPM) provide ACLs on individual files, we can remove the wrapping gLite I/O layer from a storage element and access it directly from the client side.
Refactoring of the dCache/DICOM SE is also necessary to unify the server side en/decryption and access control functionality in a single component.
Finally the Hydra keystore is being split into distributed services for reliability and to reduce the impact of a compromised key server.

Presenter: FROHNER, Akos (CERN)

Session Classification: Data Management Contributions

Contribution ID: **8** Type: **not specified**

The gLite Workload Management System

Monday 3 September 2007 08:20 (20 minutes)

The gLite Workload Management System (WMS) is a collection of components providing a service responsible for the distribution and management of tasks across resources available on a Grid. The main purpose is to accept a request of execution of a job from a client, find appropriate resources to satisfy it and follow it until completion. Different aspects of job management are accomplished by different WMS components such as the WMProxy (a Web Service managing users authentication/authorization and operation requests) and the Workload Manager (which performs the matchmaking on the job's requirements and determines where it has to be actually executed). Different kinds of job can be descibed providing needed information through a flexible high-level language called JDL. The most interesting and innovating job types are the Directed Acyclic Graphs (a set of jobs where the input/output/execution of one of more jobs may depend on one or more other jobs), the Parametrics (which allow the submission of a large number of jobs by simply specifying a parametrized description), and the Collections (which represent a possibly huge number of jobs specified within a single description) Several new functionalities (such as the use of Service Discovery for obtaining new service endpoints to be contacted, the automatic sandbox files archiving/compression and sharing, the bulk-matchmaking support), intense testing and a constant bug fixing activity dramatically increased job submission rate and service stability. Future developments of the gLite WMS will be focused on reducing external software dependency, improving its portability, robustness and usability.

Session Classification: Resource Access, Accounting, and Brokering Contributions

Contribution ID: **9** Type: **not specified**

Experimental Evaluation of Job Provenance in ATLAS environment

Monday 3 September 2007 08:00 (20 minutes)

[poster]

Grid middleware stacks, including gLite, matured into the state of being able to process upto millions of jobs per day. Logging and Bookkeeping, the gLite job-tracking service keeps pace with this rate, however it is not designed to provide a long-term archive of executed jobs.

ATLAS—representative of large user community— addresses this issue with its own job catalogue (prodDB). Development of such a customized service took considerable effort which is not easily affordable by smaller communities and is not easily reused.

On the contrary, Job Provenance (JP) is a generic gLite service designed for long-term archive of information on executed jobs. Its design priorities are:
(i) scalability – store data on billions of jobs;
(ii) extensibility – virtually any data format can be uploaded and handled by plugins;
(iii) uniform data view – all data are logically transformed into RDF-like data
model, using appropriate namespaces to avoid ambiguities;
(iv) configurability – highly customizable components maintaining pre-cooked
queries provide efficient query interface.

We present first results of experimental JP deployment for the ATLAS production infrastructure. JP installation was fed with a part of ATLAS production jobs (thousands of jobs per day). We provide a functional comparison of JP and ATLAS prodDB, discuss reliability, performance and scalability issues, and focus on the application level functionality as opposed to pure Grid middleware functions.

The main outcome of this work is a demonstration that JP can complement large-scale application-specific job catalogue services, as well as serve similar purpose where these are not available.

Session Classification: Logging & Bookkeeping and Job Provenance Contributions