

GDS: Grid Data Service

GDS is a 3 year project started in september 2003, funded by the French [ACI Masses de Données](#).

The full description of the project is [here](#) (in french).

Table of contents

| | |
|---------------------|---|
| 1 Who are we?..... | 2 |
| 2 Motivation..... | 2 |
| 3 Goals..... | 2 |
| 4 Mailing list..... | 3 |

1. Who are we?

The GDS project gathers three partners:

- [PARIS](#) Research Team, ([IRISA](#), Rennes), contact: [Gabriel Antoniu](#) (GDS project coordinator, e-mail Gabriel.Antoniou@irisa.fr)
- [REGAL](#) Research Team ([LIP6](#), Paris), GDS contact: [Pierre Sens](#) (e-mail Pierre.Sens@lip6.fr)
- [GRAAL](#) Research Team ([LIP](#), Lyon), GDS contact: [Eddy Caron](#) (e-mail Eddy.Caron@ens-lyon.fr)

2. Motivation

One of the main contributions of grid computing environments developed so far consists in decoupling deployment from computation. This aspect is particularly emphasized within the ASP (Application Service Provider) environments, where the deployment is seen as a service provided by the computing infrastructure: The programmer does not have to code it into its application, since the infrastructure is responsible for localizing the necessary physical resources and for interacting with them following a global scheduling policy.

In contrast, we must note that such a service does not really exist for handling large data sets on the grid. Whereas complex computing infrastructures are available and are able to transparently schedule computations on distributed architectures, data storage and data transfer must still to be managed explicitly by the programmer. In the best case, some advanced file transfer facilities are provided, such as GASS/GridFTP. When applications use huge amounts of data distributed on a large number of machines, this results in very complex explicit data management, which limits the efficient usage of the computational grids. This is precisely the point addressed by this project.

3. Goals

The goal of this project is to propose an approach where the grid computation is decoupled from data management, by building a data sharing service adapted to the constraints of scientific grid computations. This service aims at providing mainly two properties.

Persistence. The data sets used by the grid computing applications may be very large. Their transfer from one site to another may be costly (in terms of both bandwidth and latency). To limit the cost of these data movements, we will explore strategies based on

- reusing data already available within the infrastructure;
- enabling prefetching algorithms to anticipate future accesses;
- using data localization information to enhance computation scheduling strategies.

Transparency. We aim at providing a transparent access to data: the programmer will not have to handle data localization and transfer: this is the responsibility of the data sharing service. The service will also transparently use adequate replication strategies and consistency protocols to ensure data availability and consistency in a large-scale, dynamic architecture, where the computing and storage resources may join and leave, or fail.

Performance. The target applications are numerical simulations inherited or derived from motivating applications of high-performance parallel computing on clusters.

Scalability. The algorithms proposed within the context of parallel computing have often been studied on small-scale configurations. Our target architecture today is of the order of tens of thousands of computing nodes. On such architectures, a viable approach for data management has been proposed by peer-to-peer systems.

Mutable data. In our target applications data are generally shared and can be modified by multiple sites. A large number of strategies have been proposed for handling data replication and data consistency, in the context of Distributed Shared Memory (DSM) systems. Again, these strategies and protocols have been designed with the assumption of a static, small-scale, homogeneous architecture. In this project we assume a dynamic, large-scale, heterogeneous architecture. One of the main challenges of this project is to explore adequate consistency models and protocols by taking into account these new hypotheses.

To summarize, our approach will take its inspiration from mainly two sources: the **DSM** systems, which propose consistency models and protocols for **mutable data**, on **static, small-scaled configurations** and the **peer-to-peer systems**, which have proven adequate for the management of **immutable data** on **dynamic, large-scale configurations**. Our data sharing service will thus address the problem of managing **mutable data** on **dynamic, large-scale configurations**.

The main goal of this project is to specify, design, implement and evaluate a data sharing service for mutable data and integrate it into the [DIET](#) ASP environment developed by the [GRAAL](#) team of [LIP](#). This service will be built using the generic [JuxMem](#) platform for peer-to-peer data management (currently under development within the [PARIS](#) team of [IRISA](#), Rennes). The platform will serve to implement and compare multiple replication and data consistency strategies defined together by the PARIS team (IRISA) and by the [REGAL](#) team of [LIP6](#).

4. Mailing list

[Access to the archive of the GDS mailing list](#)