



The ARDA Project

A Realisation of Distributed Analysis for LHC

Background

Requirements

The general requirements for distributed analysis were documented by the GAG in the HEPCAL 2 report (October 2003) and have been endorsed by the SC2. The ARDA workshop on 21-22 January 2004 showed general agreement that these constitute a reasonable set of distributed analysis use cases to guide an ARDA project. There is some concern that they are not aggressive enough in emphasising interactive analysis, and the requirements for scale and performance have not been specified. These physics requirements will have to be complemented by similar use cases from other sciences that will be served by EGEE and VDT. However views expressed at the workshop were that HEPCAL 2 already covers many of the current concerns of the other sciences likely to be involved in EGEE.

ARDA RTAG recommendations -

At the workshop there was general agreement on the principal recommendations of the ARDA report (November 2004):

- specification of a set of services to support the distributed analysis requirements identified in HEPCAL 2, starting from the service decomposition developed by the ARDA RTAG;
- implement prototypes of these services, leveraging existing technology and experience where applicable;
- implementation using a web services framework;
- interfacing application middleware (e.g. POOL, LCG-AA, ROOT, GAE, ..) to the prototype;
- interfacing experiments software (frameworks, meta-data handlers, experiment-specific services, ..) to the prototype;
- the service specifications and the prototype to be developed and deployed in close collaboration with experiment data analysis teams, ensuring end-to-end coherence.

The goal is to arrive at an agreed specification of distributed analysis services, common to the four experiments, along with an initial implementation of these services. The project should then continue to improve the implementations to meet the LHC requirements for interactive and batch data analysis and production, taking account of scale, performance and stability goals.

There should be an emphasis on delivery, integration, experience and feedback – short development and deployment cycles.

The ARDA Project

Following the workshop there has been considerable discussion in the context of the LCG Project Execution Board (PEB) and on 12 February 2004 the PEB agreed to set up an ARDA project with the following high level objectives and deliverables.



Objectives

The purpose of this project is to coordinate the interaction of the different projects and activities that are involved in the realisation of a distributed analysis system:

- Grid middleware, generic in the sense that it does not have significant functionality that is of exclusive interest to high energy physics, or any other science.
- HEP common tools and libraries that interact directly with the middleware or that are specific to distributed analysis and would benefit by being part of a distributed analysis project common to the LHC experiments. Examples are POOL, ROOT, PROOF, GANGA, GAE.
- Teams within the LHC experiments that have been building distributed analysis systems using the applications environment of their experiment.
- Users - early adopters of these distributed analysis systems providing feedback to applications and middleware developers to guide development priorities and direction.
- Providers of computing resources (Regional Centres) that will take part in early deployment of the distributed analysis system.

The high-level goals of the project are:

- Arrive at a common agreement on the services provided by the middleware.
- Agree on the priorities for development of the middleware components and functionality, taking account of the requirements of the applications teams and the HEP-common tools.
- Agree on the priorities for development of new functionality in the HEP-common tools.
- Maintain a common plan covering the development of successive versions of the middleware and the HEP-common tools, their integration with the experiment distributed analysis systems, and “end-to-end” validation of the complete facility. The assumption is that there will be one such distributed analysis *prototype* for each experiment.
- Fulfil the requirements defined by HEPCAL2, enhanced by performance and stability targets to be defined for interactive and batch analysis.

As this is a coordination project the deliverables are largely produced by and owned by the collaborating projects and activities. The main deliverables are listed here.

- Specification of the middleware services. This is owned by the middleware activity, but must be endorsed by the other participants. The specification is backed up by an initial implementation, but must be sufficiently complete to allow alternative implementations to be provided by other groups or projects.
- Security model and policies. Ownership is shared by the middleware activity and the Security Group of the GDB, representing the Regional Centres.
- Early *demonstrations* of a *prototype* analysis system, including middleware integrated with example distributed analysis systems of all four experiments. Target timescale: ~ 6 months for at least two experiments.
- Validation suite for the HEPCAL2 use cases, applied to the prototype.
- Targets for scalability, performance and reliability for interactive and batch analysis in 2008.
- Initial distributed analysis *service*: enhanced middleware suitable for a sustainable distributed interactive analysis service for the four experiments, integrated with the common HEP tools and the analysis software of the experiments, and deployed at a number of Regional Centres. Target timescale: ~12 months.
- Validation suites for the general use cases and specific use cases for each experiment.
- Regular ARDA reviews assessing the success in addressing use cases and achieving targets, and the suitability of the middleware and HEP-specific tools.

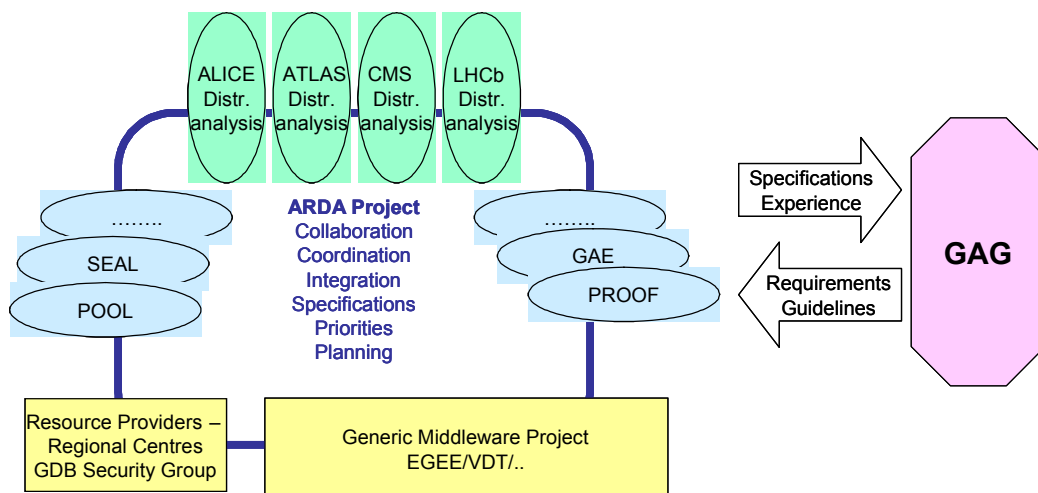


Participants

The main participants in the project are:

- The basic middleware development team. This a joint European-US team, managed as part of the EGEE project, but with a secondary reporting line to the LCG PEB.
- At least one team in each experiment, that undertakes to take part in adapting/developing the analysis system of their experiment to use the basic middleware - in other words to implement instances of the HEPCAL 2 use cases.
- Providers of common HEP tools, who are adapting and extending their products to integrate with the ARDA middleware. This would include at least POOL, SEAL, ROOT, GANGA and PROOF. Other tool providers would be encouraged to participate. The LCG applications area has some resources earmarked for activities in this area.
- Users within experiments that will exploit the evolving distributed analysis systems and provide immediate feedback to the developers.
- An integration team based in PH/SFT and partly funded by EGEE, each member with an experiment label, assisting the experiments to set up their analysis systems and integrate them with the ARDA middleware.
- Service providers from regional centres who will be involved in deploying early versions of the middleware and analysis systems.
- The participation of teams from other sciences, such as those participating in EGEE, is welcomed.

Each participating team must have a clearly defined role, with agreed responsibilities for deliverables. The decision making process must be transparent.



The overall coordination of the project will be the responsibility of a project leader, appointed by the LCG Project Leader with the agreement of the PEB. The project leader takes responsibility for pulling the above activities together, pro-actively looking for problems and finding solutions, managing the feedback loop, seeking collaboration with related activities and perhaps bringing them in to the project, but at the same time protecting the people working on the core of the project to ensure that they are not overwhelmed. The project



leader must ensure appropriate discussion with associated groups working on issues like security, data management, storage management, networking requirements.

The project leader also chairs a regular technical forum, bringing together all of the participants. As far as possible all of the common decisions needed to enable the participants to work together (specifications, targets, priorities, timescales, ..) are made by consensus in the forum. The project has the status of an LCG project "area", with the project leader an ex officio member of the PEB. All substantial decisions are reported for endorsement to the PEB.

The Middleware Development Activity

The guideline is that all services that are not HEP-specific should be defined as *middleware*. The basic middleware will be put together by a team integrating developers from AliEn, EDG, VDT, and other projects such as Dirac, NorduGrid and GAE that are willing to participate. The team will be responsible for producing the specification of the middleware services and the development of initial implementations of each service (developing from scratch or integrating and adapting existing technology). Once the service specifications have stabilised they will be presented to the GAG for endorsement as LCG standards, with which enhanced implementations by the middleware team or other developers should comply. Where appropriate the middleware team should bring these project standards to the attention of the GGF or other standardisation bodies. The team will be organised under a joint EGEE/VDT umbrella, led by Frédéric Hemmer. The intention is to use as far as possible funding from non-HEP sources. There is sufficient EGEE funding for this activity, with the accompanying matching funds already committed by institutes. Others are of course welcome to join, but must accept that working in the team will require a commitment to group decisions, and ensured availability (i.e. participants should be >60% available to work in the team).

The leader of the middleware team is a member of the LCG PEB, with the role of middleware area manager.

Role of the GAG

The GAG would have the following roles in the project:

- Preparation of requirements for distributed analysis, including a set of use cases. These should be reviewed every 12-18 months in the light of experience. The GAG should review the compatibility of the HEP requirements with similar requirements from other sciences involved in EGEE (and other projects?).
- Endorsement of middleware service specifications developed by the project.
- Consideration of feedback from the experiment data analysis groups working within the project.

Experiment Participation

It is clearly essential for the ARDA project to have identified analysis groups in the experiments with whom the middleware people can work to specify the services and validate the implementations. Each of the LHC experiments must identify someone to take responsibility for leading the prototyping of their distributed analysis systems in the context of the project, and will have to identify and commit people to work on this, with timescales and milestones agreed in the project.



A Final Word on Naming

The name *ARDA* was introduced for the *Distributed Analysis* RTAG, but has become synonymous with the “second generation” middleware implementation recommended by the RTAG, based on a factorisation of the services provided by AliEn and other middleware projects. This middleware will be developed by a joint EGEE/VDT team, with broader goals than LHC distributed analysis, covering also the wider needs of a generic, multi-science middleware project. The middleware team should **decide now** on a name for this new generation of EGEE/VDT middleware.

ARDA has also become the term used to refer to the wide discussions on LHC distributed analysis that have been stimulated by the RTAG preparation and report. If we go ahead with a project to coordinate the activities involved in distributed analysis we should keep the name ARDA. If we want it to be an acronym rather than just a name, I suggest *A Realisation of Distributed Analysis*. We should just be clear that the coordination project is just that – coordinating the integration and pilot exploitation of all of the distributed analysis activities including the middleware. The ARDA middleware is managed by its own project leader, just as POOL, PROOF and ROOT have their own management structures.