Information and Monitoring

WP3

Steve Fisher, RAL
s.m.fisher@rl.ac.uk

DataGrid is a project funded by the European Commission
under contract IST-2000-25182

3rd EU Review – 19-20/02/2004

Outline

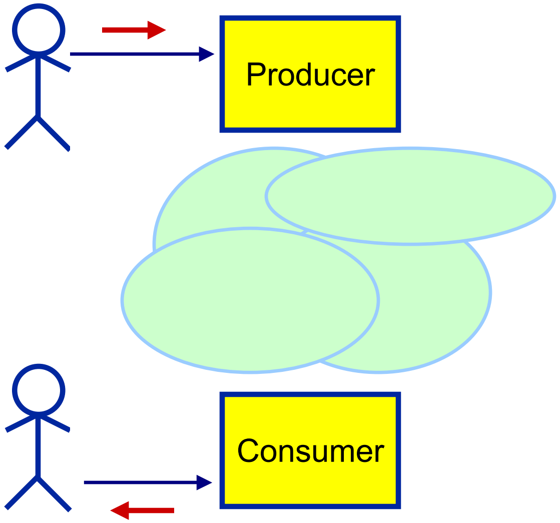
- Objectives of WP3
- Achievements
- Lessons Learned
- Future Work
- Exploitation Plans

Objectives of WP3 (TA)

- To provide a system or systems able to meet all the information and monitoring needs within a Grid from resource discovery to application monitoring

Objectives (D3.2) – part 1

- Base on the Grid Monitoring Architecture (GMA) from the GGF
  - Very simple model
- For Relational Grid Monitoring Architecture (R-GMA): hide Registry mechanism from the user
  - Producer registers on behalf of user
  - Mediator (in Consumer) transparently selects the correct Producer(s) to answer a query
- Users just think in terms of Producers and Consumers
- Use relational model (R of R-GMA)
  - Facilitate expression of queries over all the published information



Objectives (D3.2) – part 2

- Following the GMA, the system should offer one-off and streaming queries
- Ensure that all records carry a timestamp - then all information can be used for monitoring
- Highly scalable
- No single point of failure
- Dynamic schema mechanism to make it easy for applications to publish information
- Fine grained authorisation mechanism
- Ability to deal with very high rates of data to monitor the performance of parallel jobs
- Interoperation with other information systems – e.g. MDS

Achievements

- Since December the R-GMA code has become very much more stable – as recorded in D3.6
- We achieved what we set out to do by the end of the project
- Unfortunately much of the experience of the Application Work Packages has been with the earlier versions

Achievements

- We have successfully challenged the conventional wisdom on information and monitoring services on the Grid and produced a system that the user community is keen to use

- The main product is R-GMA, which does treat the whole area of information and monitoring as a single coordinated system

- Tools have been developed to allow R-GMA to interoperate with other systems:
  - GIN/GOUT – for compatibility with MDS
  - Nagios integration for displays and alerts
  - Ranglia (ganglia integration) to allow R-GMA access to ganglia

- We have a new version of GRM, which is integrated with the GridLab Mercury monitor for performance monitoring of parallel applications. This combines the flexibility of R-GMA with the performance of the Mercury monitor

Users

- WP1,2,4 and 5 for MDS like use
- WP8: CMS for BOSS – for job monitoring
- WP8: D0 – similar job monitoring
- WP7 – Network monitoring
- LCG – Trying it for accounting information

BOSS with R-GMA

- Simulation MC jobs
  - Each job creates a stream producer publishing a number of tuples depending on the job phase
- An Archiver collects tuples
  - Archived tuples are checked
- Results
  - Early 2003 fell over at around 10 jobs!
  - October 2003 managed about 400
    - Reported at IEEE NSS Conference, Oregon, USA, 21-24 October 2003
  - Now 2 MC simulations, each running 4000 jobs

Application testbed – performance

URI	data age < 300 seconds (%)	response time < 10 seconds (%)	timedout queries (%)	Service Status OK (%)
http://heplnx30.pp.rl.ac.uk/R-GMA/StreamProducerServletSummary	97	0	2	-
http://gw22.hep.ph.ic.ac.uk:8080/R-GMA/LatestProducerServlet	98	100	0	100
http://tbn08.nikhef.nl:8080/R-GMA/LatestProducerServlet	98	100	0	100
ldap://tbn08.nikhef.nl:2169/Mds-vo-name=local%2Co=grid	-	97	0	99
ldap://gw22.hep.ph.ic.ac.uk:2169/Mds-vo-name=local%2Co=grid	-	98	0	100

Lessons learned

- Release working code early
- Distributed Software System testing is hard
  - We learned the tremendous value of a private WP3 testbed. While some problems only appear with real users, we were able to detect most problems on our own testbed
  - Have recently added code to stress R-GMA on the WP3 testbed. This shows up problems which previously only showed up on the application testbed
- Automate as much as possible
  - Have made use of an open source product, Cruisecontrol, to rebuild and test software whenever people check in
    - Most of the time, most people run most of the tests
    - Cruisecontrol *always* runs *all* of them

Future

- Functionality
  - Improve Virtual Organisation (VO) support so that each VO only sees its own information
  - Need multiple physical registries for performance and for resilience
  - Implement fine-grained authorisation
  - The mediator should support a broader range of queries
- Packaging
  - Web services will be the base grid technology for the next few years, so it is essential that all WP3 software be migrated to Web services
    - Prototypes already written
  - The portability of the system will be improved
    - Need to make it easy to install "anywhere on any platform"

Exploitation – R-GMA in GGF

- The inclusion of GMA concepts in OGSA will be very beneficial to OGSA and to the widespread acceptance of R-GMA
- Have submitted documents to the OGSA working group of the GGF to explain how GMA fits into OGSA
  - We bring an implementation: R-GMA
  - Participated in phone meetings with the OGSA-WG discussing these documents
  - Attended F2F meeting of the OGSA-WG in San Diego last week

Exploitation – R-GMA in EGEE

- R-GMA will be reengineered within EGEE
  - To provide robust web services
- University-based research groups should be able to attract the necessary funding to take the ideas forward
  - Seeking collaborations to make this happen
  - Once the direction is established, those working on EGEE can produce the necessary production quality code

Exploitation – R-GMA in the world

- To increase visibility and to provide a focus for our users a web site (http://www.r-gma.org/) has been constructed
- Once the system is repackaged to make it easy to build and configure on most platforms and with clear documentation we anticipate a good take-up

Summary

- We did what we said we would
- Now the challenge is to make it a real success worldwide