Baseline Services Group Report

GDB

22nd June 2005

Ian Bird
IT/GD, CERN

Overview

- Introduction & Status
  - Reminder of goals
  - Membership
- Baseline services
  - Overview, implementation, etc.
- Future work
- Summary

Mailing list: project-lcg-baseline-services@cern.ch
Web site: http://cern.ch/lcg/peb/BS.
Agendas: (under PEB):
http://agenda.cern.ch/displayLevel.php?fid=3l132

Goals

- Experiments and regional centres agree on baseline services
  - Support the computing models for the initial period of LHC
  - Thus must be in operation by September 2006.
- The services concerned are those that
  - supplement the basic services
    - (e.g. provision of operating system services, local cluster scheduling, compilers, ..)
  - and which are not already covered by other LCG groups
    - such as the *Tier-0/1 Networking Group* or the *3D Project*.
- Expose experiment plans and ideas
- Timescales
  - For TDR – now
  - For SC3 – testing, verification, not all components
  - For SC4 – must have complete set
- Define services with targets for functionality & scalability/performance metrics.
- Very much driven by the experiments' needs –
  - But try to understand site and other constraints

Not done yet

Group Membership

- ALICE: Latchezar Betev
- ATLAS: Miguel Branco, Alessandro de Salvo
- CMS: Peter Elmer, Stefano Lacaprara
- LHCb: Philippe Charpentier, Andrei Tsaragorodtsev
- ARDA: Julia Andreeva
- Apps Area: Dirk Düllmann
- gLite: Erwin Laure
- Sites: Flavia Donno (It), Anders Waananen (Nordic), Steve Traylen (UK), Razvan Popescu, Ruth Pordes (US)

- Chair: Ian Bird
- Secretary: Markus Schulz
- … and others as needed …

Baseline services

- Nothing really surprising here – but a lot was clarified in terms of requirements, implementations, deployment, security, etc

- Storage management services
  - Based on SRM as the interface
- Basic transfer services
  - gridFTP, srmCopy
- Reliable file transfer service
- Grid catalogue services
- Catalogue and data management tools
- Database services
  - Required at Tier1,2
- Compute Resource Services
- Workload management

- VO management services
  - Clear need for VOMS: roles, groups, subgroups
- POSIX-like I/O service
  - local files, and include links to catalogues
- Grid monitoring tools and services
  - Focussed on job monitoring
- VO agent framework
- Applications software installation service
- Reliable messaging service
- Information system

Preliminary: Priorities

A: High priority, mandatory service
B: Standard solutions required, experiments could select different implementations
C: Common solutions desirable, but not essential

Service	ALICE	ATLAS	CMS	LHCb
Storage Element	A	A	A	A
Basic transfer tools	A	A	A	A
Reliable file transfer service	A	A	A/B	A
Catalogue services	B	B	B	B
Catalogue and data management tools	C	C	C	C
Compute Element	A	A	A	A
Workload Management	B/C	A	A	C
VO agents	A	A	A	A
VOMS	A	A	A	A
Database services	A	A	A	A
Posix-I/O	C	C	C	C
Application software installation	C	C	C	C
Job monitoring tools	C	C	C	C
Reliable messaging service	C	C	C	C
Information system	A	A	A	A

1) Storage Management Services

- A Storage Element should provide following services:
  - Mass Storage system: disk pool or disk-cache front-end to tape system:
    - Disk pools: dCache, LCG-dpm, DRM
    - MSS: various
  - SRM interface
    - Standard grid interface: LCG-functional set
  - gridFTP service
    - Provides remote data transfer
  - POSIX-I/O service
    - Provides local data access: rfio, dCap, etc
    - gLiteIO, GFAL, xio, xrootd, etc.
  - Authentication, authorization, audit, accounting
    - Respecting VOMS extended proxy certs (roles, groups)
- A site might have many SE
- Storage ACLs must be respected by grid or local access

SRM agreement

- All experiments require SRM at all SE
  - But for SC3 CMS stated will not require it
  - Must have for SC4
- The WG has agreed a common "LCG-SRM" set of functions, that the experiments need: (CMS ratification missing)
  - SC3: v1.1
  - SC4: LCG-SRM
- LCG SRM functionality:
  - V1.1 + space management, pin/unpin, etc
  - Not full set of V2.1
  - V3 not required
  - CMS still to confirm agreement with this set
- Coordination group with SRM developers set up in April workshop
  - Slowed down? – must push this together with experiments
- Most apps will use ROOT (via POOL or direct) to access data
  - ROOT will interface to SRM

2) Basic Data Transfer Tools

- MUST be made as reliable as possible
- gridFTP
  - Is the underlying transport protocol
  - Current version is latest in GT2
  - New version in GT4 –
    - far more reliable?
    - Provides needed hooks for monitoring etc
    - Being tested – should move to this asap
- srmCopy
  - Necessary to allow MSS to optimise scheduling etc
  - Uses gridFTP

3) Reliable File Transfer Service

- Service above basic transfer tools, that
  - Improves reliability – accepts and manages a queue of transfer requests
  - Provides scheduling and prioritisation
  - Permits inter-VO scheduling of the service at a site
  - Provides full set of monitoring etc
  - Should not prevent push-down of as much scheduling as possible to SRM
  - Provides mechanism to interact with other services
- gLite FTS proposed as prototype of such a service
- To be validated in SC3
- Other implementation – Globus RFT?
  - Does not talk to SRM, does provide retry of partial files
- File placement – not required (yet?) – could become layer to hide details of fts implementations

File transfer – experiment views

Propose gLite FTS as proto-interface for a file transfer service:

- CMS:
  - Currently PhedEx used to transfer to CMS sites (inc Tier2), satisfies CMS needs for production and data challenge
  - Highest priority is to have lowest layer (gridftp, SRM), and other local infrastructure available and production quality. Remaining errors handled by PhedEx
  - Work on reliable fts should not detract from this, but integrating as service under PhedEx is not a considerable effort
- ATLAS:
  - DQ implements a fts similar to this (gLite) and works across 3 grid flavours
  - Accept current gLite FTS interface (with current FIFO request queue). Willing to test prior to July.
  - Interface – DQ feed requests into FTS queue.
  - If these tests OK, would want to integrate experiment catalog interactions into the FTS

FTS summary – cont.

- LHCb:
  - Have service with similar architecture, but with request stores at every site
  - Would integrate with FTS by writing agents for VO specific actions (eg catalog), need VO agents at all sites
  - Central request store OK for now, having them at Tier 1s would allow scaling
  - Like to use in Sept for data created in challenge, would like resources in May(?) for integration and creation of agents
- ALICE:
  - See fts layer as service that underlies data placement. Have used FTD (with aiod as protocol) for this in DC04.
  - Expect gLite FTS to be tested with other data management service in SC3 – ALICE will participate.
  - Expect implementation to allow for experiment-specific choices of higher level components like file catalogues

4) Database Services

- Reliable database services required at Tier 0, Tier 1, and Tier 2 depending on experiment configuration
- For:
  - Catalogues, Reliable file transfer service, VOMS,
  - Experiment-specific applications
- Based on
  - Oracle at Tier 0, Tier 1
  - MySQL at (Tier 1), Tier 2
- 3D team will coordinate with Tier 1 sites
  - Tier 1 should assist Tier 2
- Must consider service issues
  - Backup, hot stand-by etc., or experiment strategy

5) Grid Catalogue Services

- There are several *different* views of catalogue models
- Experiment dependent information is in experiment catalogues
- All have some form of collection (datasets, …)
  - CMS – define fileblocks as ~TB unit of data management, datasets point to files contained in fileblocks
  - ATLAS - datasets
- May be used for more than just data files
- Hierarchical namespace
- All want access control
  - At directory level in the catalogue
  - Directories in the catalogue for all users
  - Small set of roles (admin, production, etc)
- Access control on storage
  - clear statements that the storage systems must respect a single set of ACLs in identical ways no matter how the access is done (grid, local, Kerberos, …)
- Interfaces
  - Needed catalogue interfaces:
    - POOL
    - WMS (e.g. Data Location Interface /Storage Index – if want to talk to a WLMS)
    - Posix-like I/O service

Summary of catalogue needs

- ALICE:
  - Central (Alien) file catalogue.
  - No requirement for replication
  - will use the Alien FC, but is testing both LFC and Fireman
- LHCb:
  - Central file catalogue; experiment bookkeeping
  - Uses an old version of the AliEn FC, but notes performance issues that are fixed in the new AliEn FC, will evaluate LFC now and will test Fireman.
    - selection on functionality/performance
  - The LHCb model allows the parallel use of different catalogue implementations for direct comparison of performance and scalability.
  - No need for replication or local catalogues until single central model fails
    - But will test read-only replicas of LFC in SC3

Summary of catalogues – 2

- ATLAS:
  - Will use an ATLAS provided catalogue as the central dataset catalogue.
    - Use POOL FC (mySQL) – evaluating LFC, Fireman.
  - Local site catalogues (this is the ONLY basic requirement) – will test solutions and select on performance/functionality (different on different grids)
  - Expect that the local site catalogues will be provided by the local grid infrastructure middleware. In the EGEE sites this will be LFC or Fireman, in the US likely to be the Globus RLS. In NDGF not yet clear
- CMS:
  - Central dataset catalogue (expect to be experiment provided)
    - but LFC or Fireman could also fulfill this need.
  - Local site catalogues – or – mapping LFN→SURL; will test various solutions
  - Local site catalogues will be implemented in the same way as for ATLAS – by the local grid infrastructures.

Catalogues – comments

- No immediate need for distributed catalogues;
- Interest in replication of catalogues (3D project)
- Potential for a reliable asynchronous catalogue update service
  - Simple (re-use FTS?); RRS?
- Sites would like that all catalogues at a site be the same implementation – run a single service for all experiments.
- Summary table of catalogue mapping and other issues: http://lcg.web.cern.ch/LCG/PEB/BS/baseline-cats.html
  - (missing CMS entry)

Catalogue implementations

- AliEn FC:
  - provides the mapping interfaces required by ALICE.
  - Does not interface to POOL as this is not required by ALICE.
  - Implements Storage Interface, and provides metadata interface
- LFC (LCG File Catalogue):
  - provides all the interfaces described here: POOL, implements the DLI, and can be used together with GFAL.
- FireMan (gLite file catalogue):
  - also provides all the interfaces described here: POOL, implements the Storage Index (and soon DLI also), and works with the gLite I/O service.
- Globus RLS:
  - is now integrated with POOL. Does not (???) implement the DLI or Storage Index interfaces. Posix I/O ???
  - Does not have hierarchical namespace
- DLI vs SI → this should converge

6) Catalogue & Data Management tools

- Provide a set of management tools that combine
  - POOL cli tools
  - lcg-utils
  - gLite cli tools
- Should be made into a consistent set of "lcg-utils"

7) Compute Resource Services

- Compute Element is set of services providing access to compute resources:
  - Mechanism to submit work to LRMS
    - Globus gatekeeper, gLite CE, ARC (?)
  - Publication of information – GLUE schema
  - Publication of accounting information
    - Agreed schema (based on GGF schema)
  - Mechanism to query job status
  - Authentication, authorization service
    - Respecting VOMS extended proxy – roles, groups, etc.
- Several experiments need to manage relative priorities between users themselves
  - Interface to LRMS fairshare, or
  - Permit VO to manage its own queue (task queue)
    - gLite/Condor-C CE
- Issue – take up of Condor_C based CE: EGEE, OSG, NDGF?

8) Workload Management

- Workload management services (e.g. RB, etc) will be provided
  - Are required by ATLAS, ALICE, CMS
  - For LHCb is low priority compared to other services
- WLM solutions should use the DLI/SI interfaces to enable catalogue interactions in a transparent way
  - To schedule jobs where the data resides
- Expect WLM solutions will evolve and become a basic service

9) VO Management Services

- VOMS is required
- EGEE, OSG, NDGF do, or plan to, deploy VOMS
- Mechanisms for user mappings may vary from site to site
  - Depend on grid and site policies
- Must all respect same set of roles, groups, etc and agreed granularity

10) POSIX-I/O services

- Require POSIX-like I/O operations on local files
- Normally via POOL/ROOT but also directly
- Implementations must hide complexity of SRM, catalogues from application
  - Provide open on guid/LFN
- Existing implementations:
  - GFAL, gLiteI/O (based on aiod), xio (?)
- Security models differ – more work to be done to understand implications of security and what best implementation is:
  - Appropriate security model
  - Performance, Scalability etc

… and

11) Grid Monitoring tools
- Importance of job monitoring, accounting, etc
- Job monitoring: ability to look at job log files while running, trace jobs when failures, etc.

12) VO agent framework
- Require mechanism to support long-lived agents
- E.g. for asynchronous catalogue updates, etc.
- Better as a framework run by site as a service
- Could also be dedicated/special batch queues – some advocate this; prototype as a fork-only CE.

13) Applications software installation tools
- Existing mechanism is acceptable for now
- Extension of current functionality should be discussed

14) Reliable messaging
- Not really discussed but is a common need

15) Information system
- GLUE schema → version 2 by end of 2005 common between EGEE, OSG, ARC

Summary

- Set of baseline services discussed and agreed
  - Nothing surprising or controversial (?)
- Almost all services exist now in some form
  - Most can be tested now
- Group has not finished its work
  - Have not addressed performance targets
  - Should follow up on some issues: lcg-utils, VO-agent, monitoring, etc
  - Should continue to monitor progress
  - Remain as technical forum
  - ☞ Proposal to next PEB

- Thanks to all who contributed to the very illuminating and useful discussions