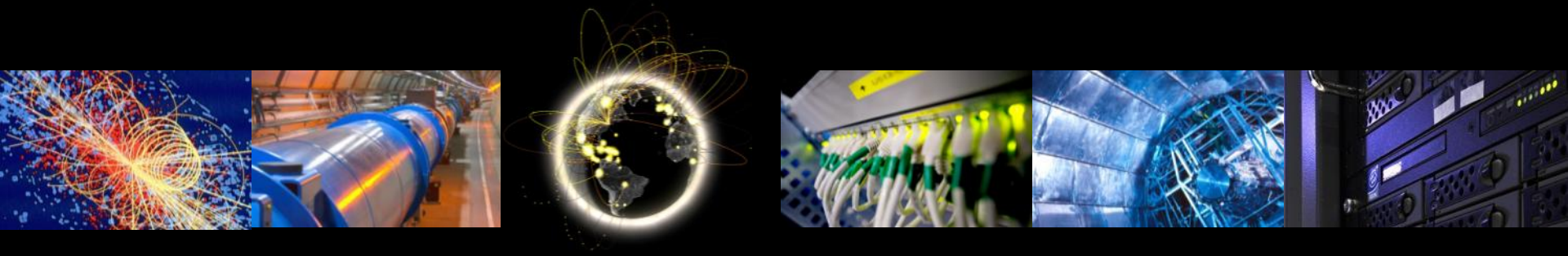
Summary of Data Management Jamboree

Ian Bird

WLCG Workshop

Imperial College

7th July 2010



WLCG Worldwide LHC Computing Grid

Summary – 1

- **General points**
  - Today use of available resources, particularly network bandwidth and disk storage is probably not optimal, and reasonable step can be taken to make the existing resources more effective.
  - Using the network to access data when it is not available locally is a reasonable mechanism to improve the efficiency of data access (e.g. 95% of files may be available at a site, the other 5% can be retrieved or accessed remotely, presumably with local caches too).
- **Storage**
  - It was clear that traditional HEP use of tape (write once, read very often) is no longer supportable. It clearly cannot support the almost random access patterns typical in all but organised production scenarios, and there is a distinct mismatch in performance between tape and the consumers.
  - The (tape) archives should be separate from the caching layers, and treated as a true archive where data is written and rarely read. The exception is when entire datasets may be brought online by a production manager, or when data has been lost from disk and needs to be recovered.
  - This should simplify the interfaces to both the archive layer and the caches. The interface to the archive can be based on a simple subset of SRM, to enable explicit move of data between archive and cache (essentially including bulk operations).
  - This will also allow the use of industrial solutions for managing the archives, and puts the use of the large tape systems into the operational mode for which they were designed (as opposed to the way in which HEP traditionally uses them).
  - This also removes the direct connection between cache and archive, avoiding that general users can trigger random tape access.

Summary – 2

- **Data Access Layer**
  - Improving the efficiency of user access to data, particularly for analysis, was one of the major drivers behind this workshop.
  - It should no longer be assumed that a job will find all the files that it needs at a site. Missing data should be accessed remotely or fetched and cached locally. Which is the better mechanism should be investigated.
  - It cannot be assumed that a catalogue is completely up to date, and hence it is important that missing data be fetched on demand.
  - Effective caching mechanisms can reduce or optimise existing usage of disk space. Several caching mechanisms and ideas were discussed. Particularly interesting are proxy caches where a proxy at a site manages the interaction between requestors and remote access.
    - It is important that the caching tools and caching algorithms be treated separately. A variety of algorithms may be required in different circumstances and should be supported by the tools.
  - A combination of organised pre-placement of data and caching may be one option.
  - Alternatively it could be that no pre-placement is needed and that data popularity will define what gets into the caches.
  - The desired model of data access by the application is that of the file system. I.e. the job should be able to open, read, etc. and not have to concern itself with the underlying mechanisms. Complexities such as SRM should not be visible to the user.

Summary – 3

- **Data Transfer**
  - There are several data transfer use cases that must be supported:
  - A reliable way to move data from a job to an archive, asynchronously. A job can finish and give its output file to a service and be sure that the data will be delivered somewhere and catalogued. (The tools perhaps should separate the two operations – i.e. catalogue updates should not be in the transfer tool).
  - A mechanism to support organised data placement.
  - A mechanism to support data transfer into caches.
  - Support for remote access to data not at a site, or to trigger a transfer of data to a local cache.
- Several specific requirements on improving FTS were made, which should also apply to any other transfer mechanism:
  - Well defined recovery from failure.
  - Ability to do partial transfers, or to transfer partial files, particularly to recover from failure
  - Ability to manipulate chunks of data (many files, datasets) as a single entity

Summary – 4

- **Namespaces, Catalogues, Authorization, etc.**
  - The need is for dynamic catalogues that reflect the changing contents of storage, and must be synchronized with the storage systems (asynchronously)
  - The experiment computing models must recognise that this information may not be 100% accurate. Again back to the need to recover from this by remotely accessing (or bringing from remote sites) the missing files.
  - Technologies for achieving the catalogue synchronisation could be based on something like LFC together with a Message Bus, or perhaps the "catalogue" could be actually a dynamic information system (e.g. Message Bus, Distributed Hash Table, Bloom Filter, etc.).
  - There is a need for a global namespace, based on both LFNs and GUIDs depending on usage.
  - ACLs should be implemented once (e.g. in Catalogue) but back doors into storage systems must not be allowed.
  - The Alien FC was proposed as a general solution for a central catalogue, with many other features.

Summary – 5

- **Global Home Directory facility**
  - Such a facility seems to be required. The Alien FC has this as an integral part. Other solutions could be based on commercial or opensource technology (drop boxes, Amazon S3, etc.).

- Demonstrators:
  - To be fleshed out and plans presented at WLCG workshop next week

Demonstrators proposed

- Brian Bockelman: xrootd-enable filesystems (HDFS, Posix, dcache + others) at some volunteer sites. Global re-director at 1 location, allow the system to cache as needed for Tier 3 use.
- Massimo Lamanna: very similar proposal with same use cases for ATLAS. Also include job brokering. Potential to collaborate with 1)?
- Graeme Stewart: Panda Dynamic Data Placement.
- (name?): LHCb/Dirac – very similar ideas to 3).
- Gerd Behrman: ARC caching technology: propose to improve the front-end to be able to work without the ARC Control Tower, also to decouple the caching tool from the CE.
- Jean-Philippe Baud: Catalogue synchronisation with storage using the Active MQ message broker. a) add files, catalogue them and propagate to other catalogues; b) remove entries when files are lost if a disk fails; c) remove a dataset from a central catalogue and propagate to other catalogues.
- Simon Metson: DAS for CMS. Aim to have a demo in the summer.
- Oscar Koeroo: Demonstrate that Cassandra (from Apache) can provide a complete cataloguing and messaging system.
- Pablo Saiz: Based on Alien FC – comparison of functionality, and demonstration of use in another experiment.
- Jeff Templon: Demonstrate the Coral Content Delivery Network – essentially as-is. Proposed metrics for success.
- Peter Elmer: wants to show workflow management mapping to the available hardware (relevant to use of multi-core hardware).
- Dirk Duellmann/Rene Brun: prototype proxy-cache based on xrootd. Can be used now to test several things.
- Jean-Philippe Baud+Gerd Behrman + Andrei Maslennikov + DESY: use of NFS4.1 as access protocol.
- Jens Jensen + (other name?): simple ideas to immediately speed up use of SRM and to quickly improve the lcg-cp utility