CMS plans

Peter Elmer
Princeton University
4 March, 2005
LCG BS meeting

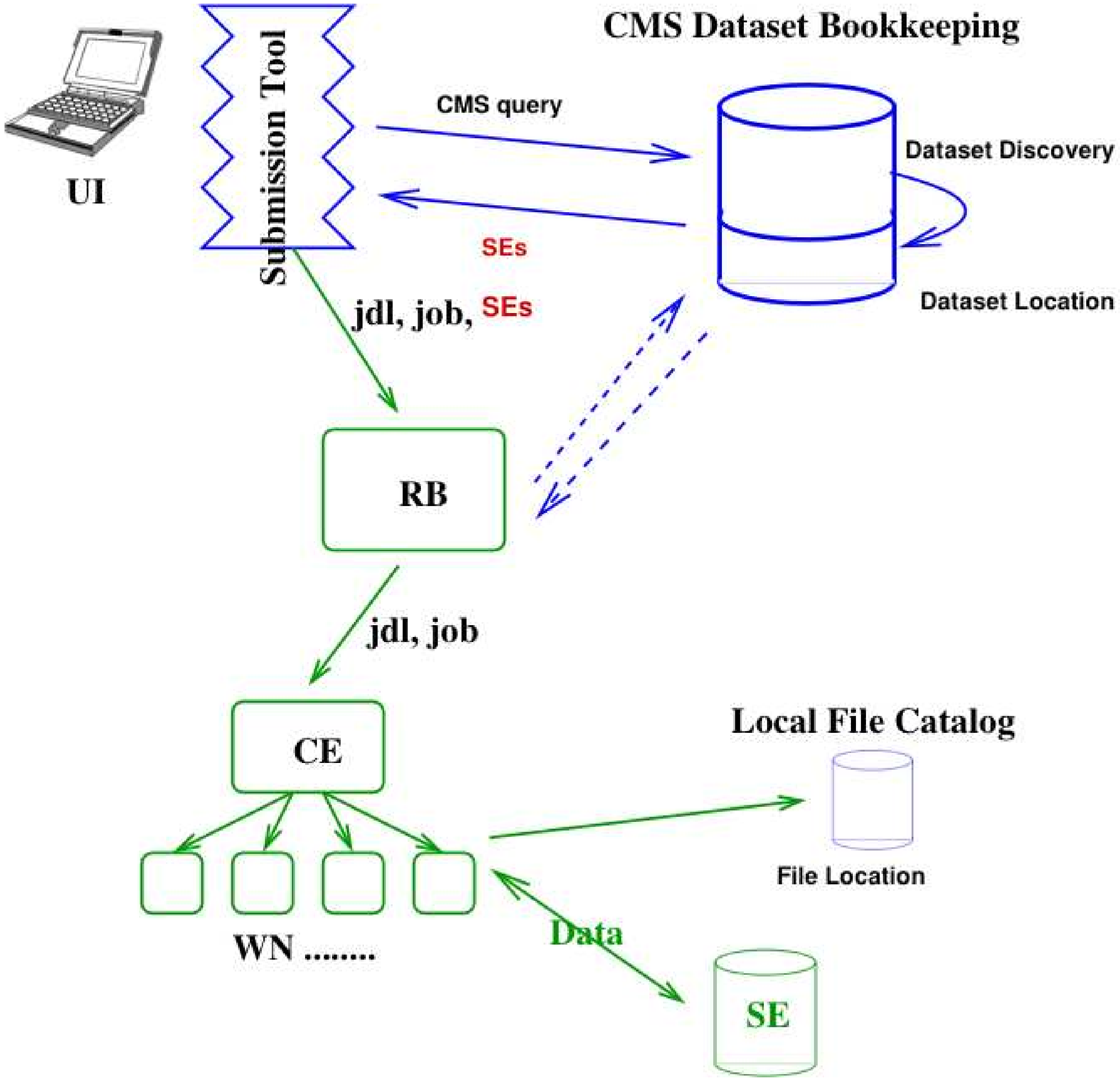
Phedex – data transfers

- Phedex transfers are layered:
  - Actual bit transfers for files (globus-url-copy, srmcp, lcg_rep, …)
  - Point-to-point (= single-hop) reliable file transfer
  - Routed (= multi-hop) reliable file transfer
  - “Workflow” layer: allocator
- At the highest level, transfers are in general dataset-oriented
- Recently introduced file set “chunks” as a scale factor and to partition “active” transfers from those which have been finished

Phedex – data transfers

- As with many things, the key here isn't really new tools, but instead the deployment of fully operational *systems* that can be used 24x7 without “challenge”-style manpower requirements
- In practice this includes operational gridftp servers at all sites, operational storage at all sites and operational tape systems. SRM desirable in at least the largest sites.

CMS Baseline



- Dataset location service
- Dataset bookkeeping service
- Data access/storage
- Local file catalogs

Dataset location service

- CMS has decided that its baseline will not for the moment include a global file catalog
- Data will be managed with a granularity of either datasets or “data blocks” using CMS tools and decisions, “custodial” ownership of data by sites
- The “dataset location service” will map datasets or “blocks” to specific sites
- Contains “mutable” information, but scale is 3-4 orders of magnitude less than global file catalogs

Dataset bookkeeping service

- Expresses CMS-specific notions of relationships between entities like datasets, runs, event collections, “file blocks”and luminosity as well as providing “selectable attributes” on these entities.
- Lowest level component will like be an LFN-catalog
- Main use case is “data discovery” for analysis and detector/reco/calib studies (input to WM, job configuration, data transfer, ...) - “What exists?”
- Mostly non-mutable, all of it is site-independent

Dataset bookkeeping service

- The detailed design for this is currently being discussed, this will be CMS-specific.
- In part this is new functionality, in part an evolution and refactorisation of the existing PubDB/RefDB
- Expect a prototype in a few months, plus an iterative development process after that
- Expect soon to have a more concrete idea of the “distributed database” aspects at which point it will be clearer how we might be able to use (for example) the 3D project (techologies looking for a use-case)

Data access and storage

- By construction this is a site-specific choice for CMS and different solutions will be appropriate depending on the scale of the site.
- A number of solutions already in use by running experiments: xrootd/olbd, dCache, Castor, SRM (+ various MSS implementations). There are active collaborations associated with these projects.
- (Repeating earlier CMS comments): LCG should examine existing solutions and work together with those teams to develop a coherent plan rather than provide its own solutions in isolation.

POOL + local catalogs

- We expect that to provide a "trivial catalog" POOL implementation to allow simplification the LFN to PFN mapping for standard cases where this can all be deferred into a storage system.
- We are looking into different ways to reduce the need for explicit management of GUID to LFN mappings, in particular for bulk data, as part of the new EDM and DM projects in CMS.
- Job can thus be configured in many default cases with no site-specific information. The full flexibility of the catalog is still available for special cases.