

These minutes were hastily scribbled by Adam. Please send him corrections and/or comments.

**** 2006.09.14 Meeting**

See agenda at <http://cd-docdb.fnal.gov:8080/cgi-bin/DisplayMeeting?conferenceid=227>

***** DO Db servers**

Moving DB Servers -- the reason to move is due to Oracle 10 client on solaris. And that we seem to be pegging d0ora2 sometimes.

SW mentioned that having the chooser on a high availability machine and DB servers on backends, would be good for CDF since they have no redundancy at all for the DB servers.

Want a little document with our plans. Plan - Wait for machines, test things on two, write out a plan for what we want where.

***** CDF autodeest and new robot**

Dehong sitting in for Krzysztof...
(E-mail from Krzysztof with proposal)--
after talking with Dehong, we suggest that the table:
FILECATALOG.CDF2_FILE_FAMILIES be extended to include
the following enstore pnfs tag info:

```
library  
file_family_wrapper  
file_family_width
```

with the initial values:

```
library = CDF-9940B  
file_family_wrapper = cpio_odc  
file_family_width = 1
```

(with proper to be determined types)

We hope that this, with proper modifications of the sam autodeestination server, would allow for gradual migration of file families to the new robot, and would also allow for an ability to set the file family width and the library wrapper on a per family basis.

How does autodeest interact with this?

So when parameters change, the get destination method should look at this table and make a new leaf with the new parameters, even if the old leaf was not full.

Randolph: Problem is that to make this possible, some of the code that needs to be changed is common to all experiments.

Steve - Current system was never intended to change the current leaf. Always assumed that creation of new leaf would pick up information of old leaf would have old parameters. New leaf would only be created when old leaf was full.

What's new is checking the parameters to see if we need a new leaf (for every file store). -- this breaks the common code.

Right now, you can't create a new leaf unless the old leaf is full,

and that common code figures that out by looking at the DB. -- Hard to do this by hand.

R: Needs a callback to experiment code to do "force full" hook that would force a new leaf and get the new parameters. Checking if things have changed requires lots of DB lookups (but fast because tables are small).

SW: Upon check, update the storage config table from file.family table - then the rest of the code will work.

What makes this work is the hook that only slightly breaks the common code.

SW: Randolph should code this up so that more than one person knows the autodesc code.

SW: More concerns: If CDF wants additional changes that breaks the common code further, then the code will need to be split and a request for time will be required.

What if we said no---

They could update the storage locations with the new information; BUT there is no way for them to make a new leaf without changing another table. Would have to carefully alter two tables -- very prone to error.

What is in there now is that a file family and data-tier pair defines what PNFS leaf you go do. --- Is this functionality still required? This was hard to do, so we don't want to take it back out.

R: Questions for K - Is this your final answer? [Where's my million dollars?] -- e.g. once we go down this path, we're not going to change it, so make sure this is what CDF really wants.

R: Perhaps put in all enstore parameters (6, not 3) into the file.family table so that they're there and if any change do the stuff (e.g. change permissions - get a new leaf).

SW: Also asked Randolph to fix problem that prevents db4 to db5 migration (not anything major).

So conclusion is to do the work to implement what we believe CDF wants, but wait for Krzysztof to verify.

*** Null movers

Problem - Dehong uses null movers, when the path name is constructed with the PNFS leaf it must modify it for null movers. So the idea is to add a column to the file family table to identify null movers which will modify the pnfs path accordingly. [This only goes into the CDF tables and the CDF specific code]. Randolph claims this is a trivial change, ONCE the above changes are also done to autodesc. To do this on its own would be difficult.

*** SRM Update (Andrew)

SRM installed in Italy - Andrew has code for SRM upload for Italy. (but Italian guy Andrew was working with disappeared -- we need him to help with CDF SAM upload to make it SRM aware). Also getting a

node from Joe for transatlantic transfers.

Tests with Steve and MIS - tests succeed (varying datasets, number of files, number of projects). Now combining SRM tests with MIS and the test harness. Incorporated MIS into test harness. Still perhaps memory jumps -- dumping the station info caused memory to jump by 500 MB. Without MIS, SRM tests worked fine. [using dCache model - no data movement for tests].

Dehong will help with changing sam-upload (but still need Italians to try it out and for operations issues).

*** MIS

Using for SRM tests and it works. Problem found by listing orphaned files in the MIS information that was related to the prefetch file limit. We've learned stuff from MIS!

Next step - getting it into production.

*** Robert's name service

Last D0 downtime had to restart nameservice, and it wouldn't start because of corrupted persistence file. So we have to kill all bindings. Updating this "log" file is not transaction safe (if disk is full, all dies). And the nameservice randomly crashes [but not seen by CDF?].

Robert is writing his own nameservice with a SQLite persistence back end. It is also possible to make a nameservice that could almost automatically fail over (could put multiple host-names in a Corba IOR). The problem is to distribute the persistence so that the new nameservice could start up with the old bindings. SQLite is not good for this, but perhaps put the persistence in Oracle.

Two questions: Is performance good enough (using python right now)? Is there a transparent way to migrate? (Problem here is that the Orbacus name service has a orbacus only bit that is not part of the standard. We would have to redistribute IOR strings to the SAM universe or somehow fake the orbacus part).

Now working out if python is good enough or would have to put into C++.

*** Pick events from DBB

Try

[http://dbb.fnal.gov:8520/answers?exp=d0&what=pick_event_files&data=\(\(206915,39041776\)\)](http://dbb.fnal.gov:8520/answers?exp=d0&what=pick_event_files&data=((206915,39041776)))

Put in run and event numbers, get back file name, file ID, run, event.

The SQL to do this is (see IT 2195)...

```
1 select file_id,
2     file_name,
3     run_number,
4     event_number
5 from data_files
6 join (select file_id, run_number, event_number
7     from runs r
```

```
8      join events e using(run_id)
9      join event_catalog ec using(event_id)
10     join data_files df using(file_id)
11     where (run_number,event_number) in
12      ((204158,57981582),
13       (206915,39041776),
14       (206915,39089774),
15       (206915,39099745),
16       (204158,57986220)) using(file_id)
17* order by file_id, run_number, event_number
```

I should talk to D0 about eliminating the use of the event catalog or make it smaller. For pick events, could use the first and last event numbers we store in data_files (but with streaming, this could result in multiple files having to be looked through).

Resource Selection in OSG

Parag Mhashilkar

Fermi National Accelerator Laboratory

Comments from Adam are in text boxes like this.

This is a comment from Adam too

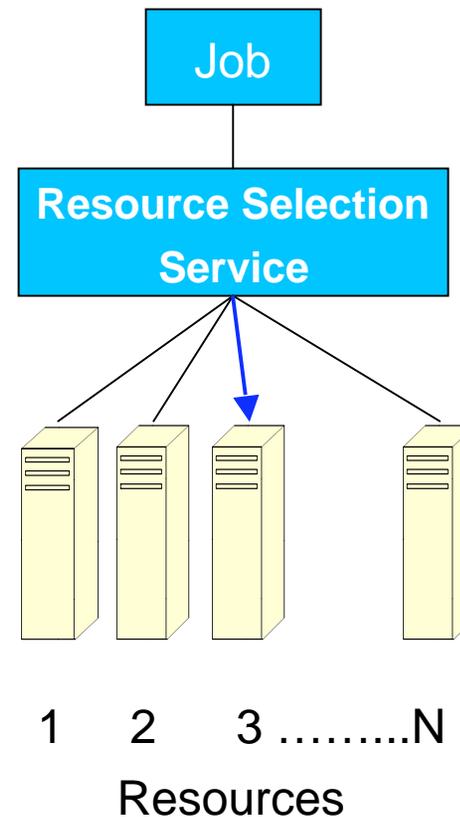


Resource Selection in OSG

- **Overview**
 - Resource Selection Service
 - Resource Selection Service Architecture
 - Work In Progress

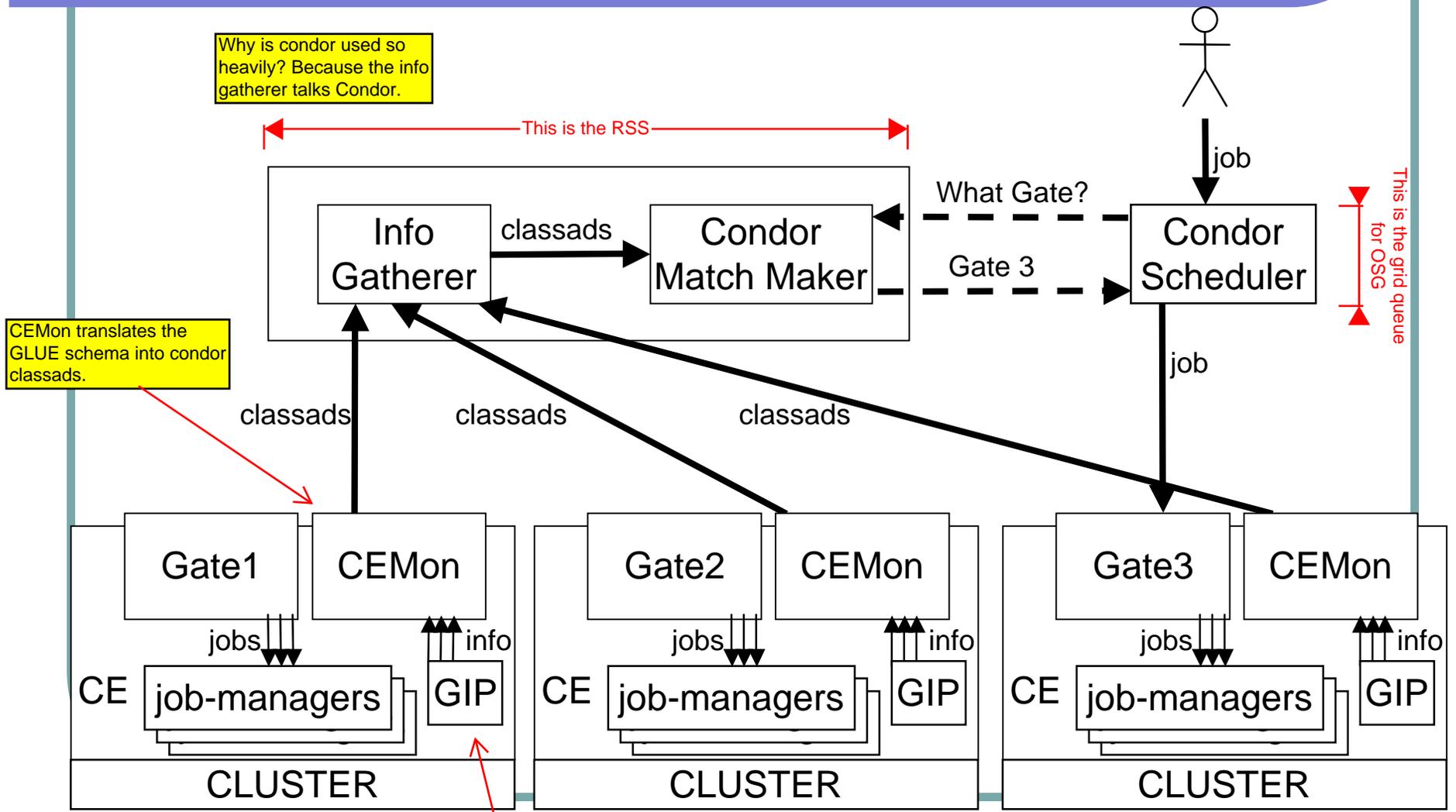
Resource Selection Service

- Gather the information about the job and resources and make decision where the job should run.
- Dereference abstract attributes to bind to the job during match-making or execution time.
- Use Case for DZero
 - Example, site can be certified to run p20 Monte Carlo. Site needs to advertise this information as a part of the GlueSchema. Resource selector matches the job based on the information available.
- The Resource Selector is a component of the OSG Job Management Infrastructure.



What does LCG do? LCG resource broker is a distributed resource broker (and is also a queue for the grid jobs). For OSG, the grid queue is a different program (part of condor). But we haven't used tagging for LCG, mostly because of politics [remote submitters want to control grid jobs to go to their resources]. Want to avoid these politics for OSG.

Resource Selection Service Architecture



Why is condor used so heavily? Because the info gatherer talks Condor.

CEMon translates the GLUE schema into condor classads.

GIP is the grid information provider which gives information in the GLUE schema.

Architecture ...

- Generic Information Provider (GIP) describes resources in LDIF format using GLUE Schema.
- CEMon provides flexible plug-in mechanism to translate classads.
- Information Gatherer (IG)
 - Subscribes to several CEMons to gather the information about the CEs and advertises it to several condor pools.
 - It acts as an adapter between CEMons and Condor matchmaker.
- Support for callouts to external match-making functions. These functions can make match-making more extensible.

Work In Progress

- Resource broker has been deployed on integration test bed at Fermilab. We are working with OU to do the integration testing for ReSS.
- Testing/Studying the scalability of IG and ReSS infrastructure. Focus is on testing the match making infrastructure submitting $O(1)$ job/sec for 1 hour.
- Plans to do p20 'primary processing' with Samgrid via OSG. This will involve –
 - Number of events: 300,000,000
 - CPU hours: 5.8 Million (4 month job with 2,000 CPU's)
- In future -
 - Support for tagging certified sites and making resource selection more manageable.
 - Support new VOs

IG = info gatherer

Parag is now doing stress testing. Note that the RSS requires a new version of the OSG release code. Instead of waiting for a new release, they will instruct sysadmins to install the needed newer components.