BaBar MC Production on the Canadian Grid using a Web Services Approach

Ashok Agarwal, Ron Desmarais, Ian Gable, Sergey Popov, Sydney Schaffer, Cameron Sobie, Randall Sobie, Tristan Sulivan, Daniel Vanderster

University of Victoria

Overview of GridX1 – A GT2 Grid

- Use Canadian resources
  - calliope, mercury, mcgill
- Clusters: standard Globus Toolkit 2 (GT2)
- Resource Mgmt:
  - CondorG-based MS
  - Condor Brokering
  - MyProxy credential repo
- Central monitoring and accounting with web GUI
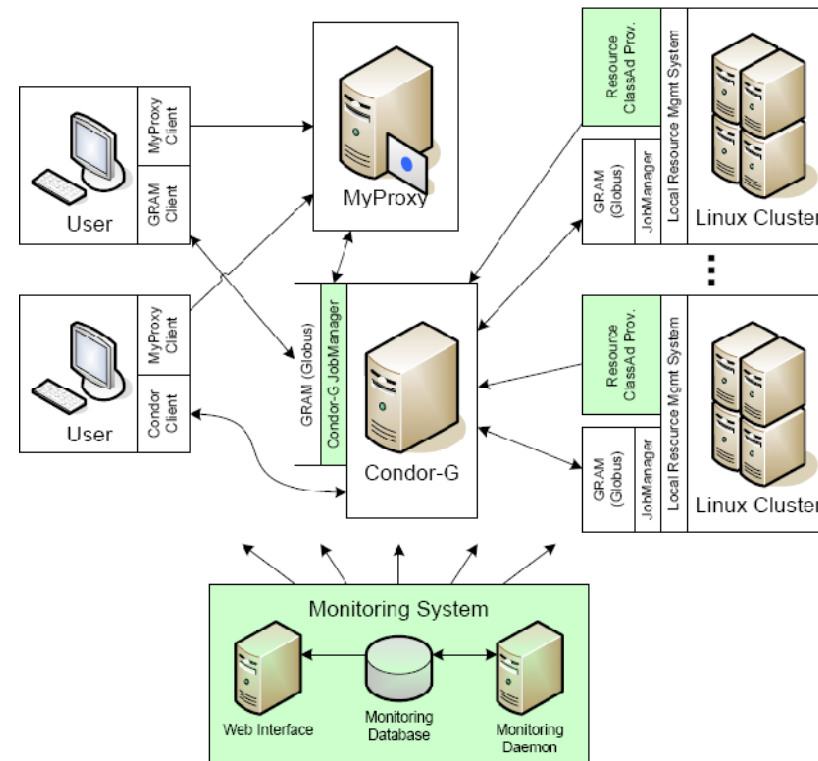
Limitations of GridX1

- Due to numerous service-specific protocols
  - e.g. GRAM, MyProxy, Condor

1. Difficult to extend:
   - Adding a new service involves modifying protocol, or developing a new one
2. Compatibility issues:
   - Lack of protocol standardization
   - Backwards compatibility is not perfect (protocols modified between releases)
3. Firewall problems:
   - Each service uses its own TCP port
   - Many ports must be opened by each institution
     - this may conflict with local policies
4. Security vulnerabilities:
   - The GRAM job service runs as root, which could lead to a compromised resource
   - Access is often limited to trusted hosts, limiting usefulness of the service

Solution: Web Services Resource Framework (WSRF)
Globus Toolkit v.4

Globus Toolkit v.4 Advantages

- WSRF solves the 4 key problems with GridX1

  1. Easy to extend
     - One common protocol (SOAP)
     - Easy to develop new WSRF services

  2. Seamless upgrade support
     - Changes to service interfaces are described in WSDL

  3. Reduced firewall problems
     - Fewer ports (the service container)
     - Non-privileged ports

  4. Good security
     - Service container runs as non-privileged user

WSRF-Based Grid

- Consists of multiple metaschedulers

- Central resource registry to store the resource attributes, RFT and LRMS

- Having multiple metaschedulers and registries gives high scalability and reliability of the grid

Proposed WSRF-based grid
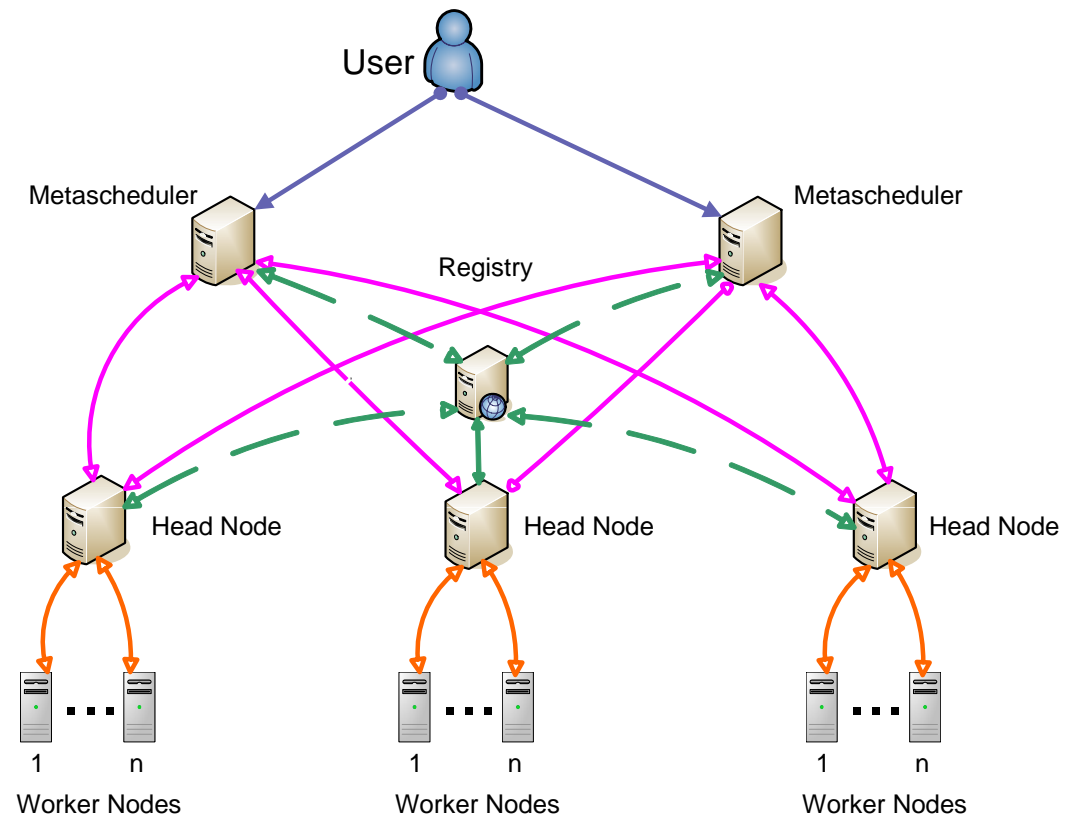
Important Features

- Condor-G used as the metascheduler

- Automatic registering of resource ClassAds to the central registry

- Automatic ClassAds extraction from the registry to the metascheduler for matchmaking

- Incorporation of input/output file staging

- Job submission using WS-GRAM or Condor_Submit

- Web-based monitoring

Metascheduler Service

Publishing Tool
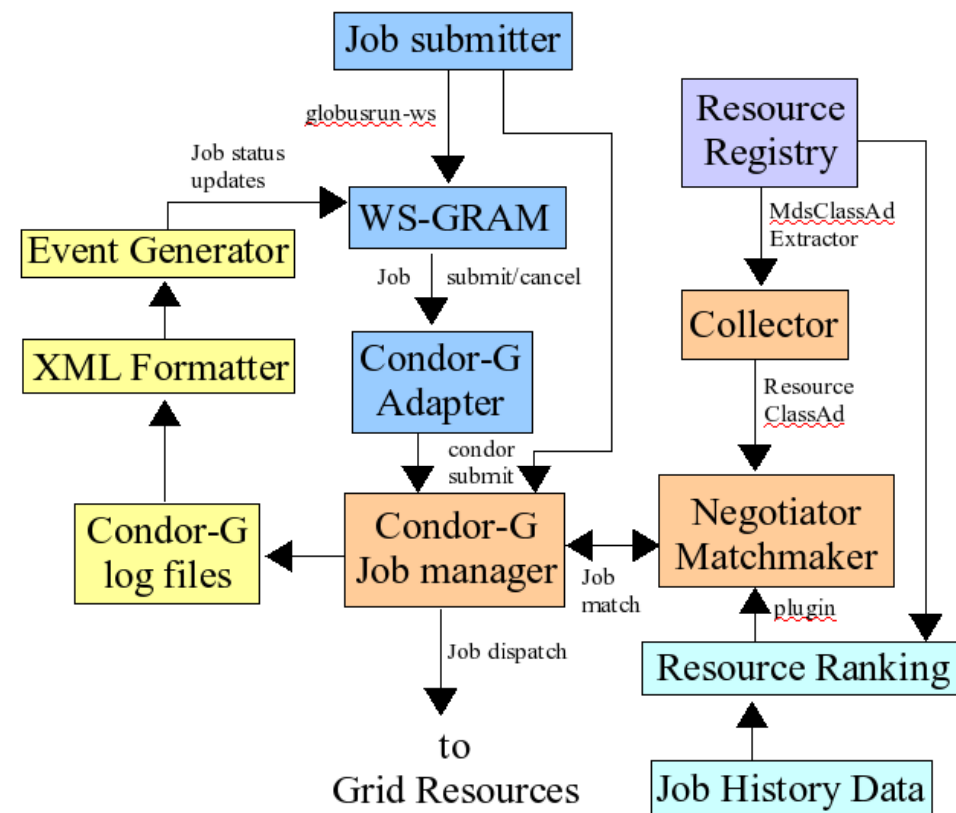
ClassAd Publishing Tool

- An information provider script runs on every Grid resource and generates the resource ClassAd in GLUE 1.2 scheme

- Inserts the resource information in the form of XML-formatted Condor ClassAds into the local WS-MDS

- Publishes the resource information into the WS-MDS of the central registry
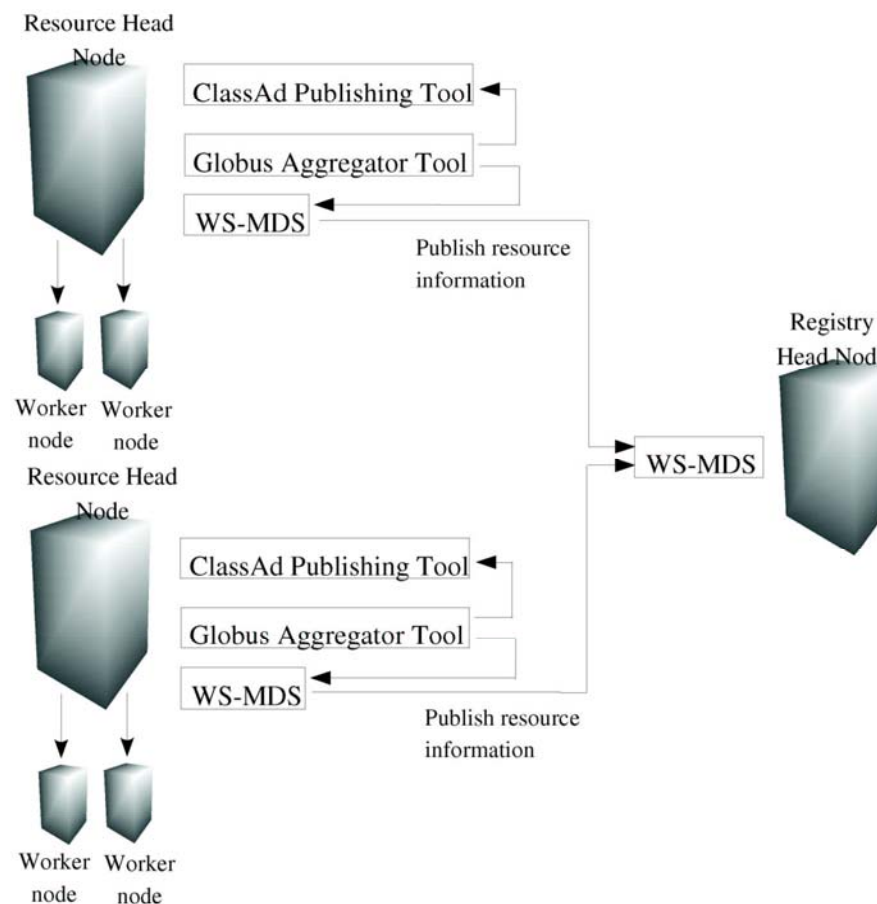
ClassAd Extractor

- Java application software

- Runs periodically on the metascheduler

- Extracts compute resources ClassAds in GLUE 1.2 scheme

- Converts the XML data into the Condor ClassAds for each resource

- Publishes these ClassAds to the Condor collector for jobs matchmaking

MdsClassAdExtractor Tool
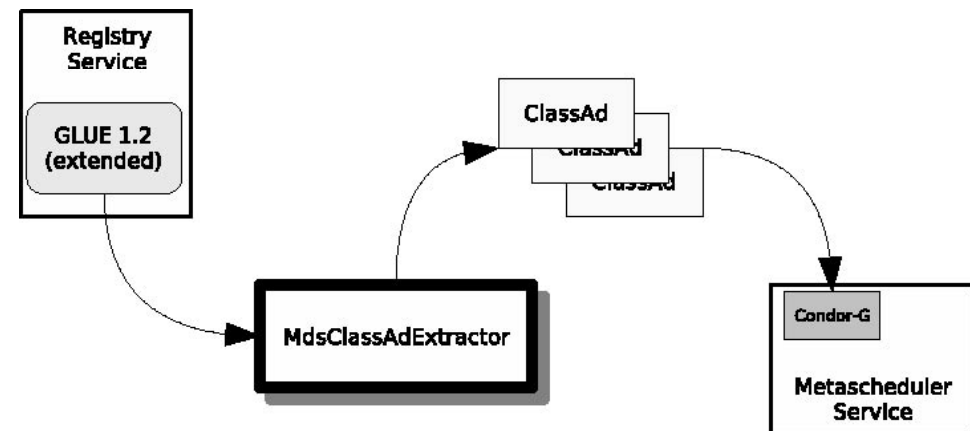
BaBar MC Grid Requirements

- Metascheduler head node
  - Install Condor-G and GT4
  - Install BaBar software
  - Set up metascheduler
  - Set up ClassAd extraction tool

- Grid Resources Head Node
  - Set up Portable Batch System (PBS) to act as the local resource management system
  - Install BaBar software
  - Set up classad.pm to advertise the local resource information
  - Set up ClassAd publishing tool

BaBar MC Production Setup Using Resource Broker
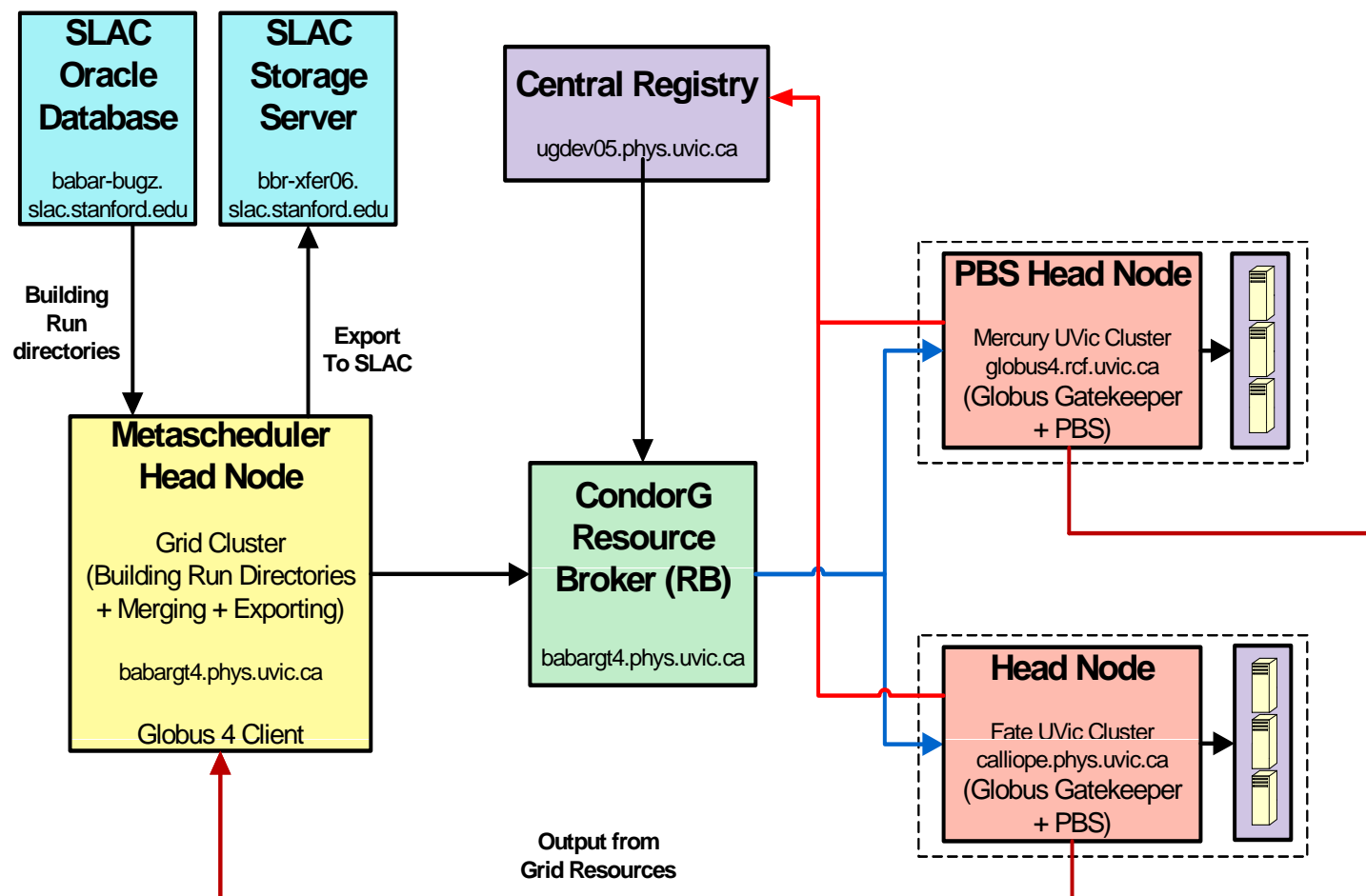
Performance Test

Comparison of CPU time and percent efficiency (% Eff)

Sr. No.	SP8 Validation Run Number	Globus Toolkit 4				Globus Toolkit 2			
		Fate		Mercury		Fate		Mercury	
		CPU Time (HH:MM)	% Eff	CPU Time (HH:MM)	% Eff	CPU Time (HH:MM)	% Eff	CPU Time (HH:MM)	% Eff
1	9941582	4:26	99	4:59	85	4:23	99	4:06	98
2	9941585	4:24	99	5:01	80	4:40	98	4:08	98
3	9941587	4:31	98	4:55	84	4:26	99	4:14	97
4	9941589	4:09	98	4:40	85	4:41	98	4:20	97

% Eff = CPU Time / Wall Time

JSAM: Job Submission Tool
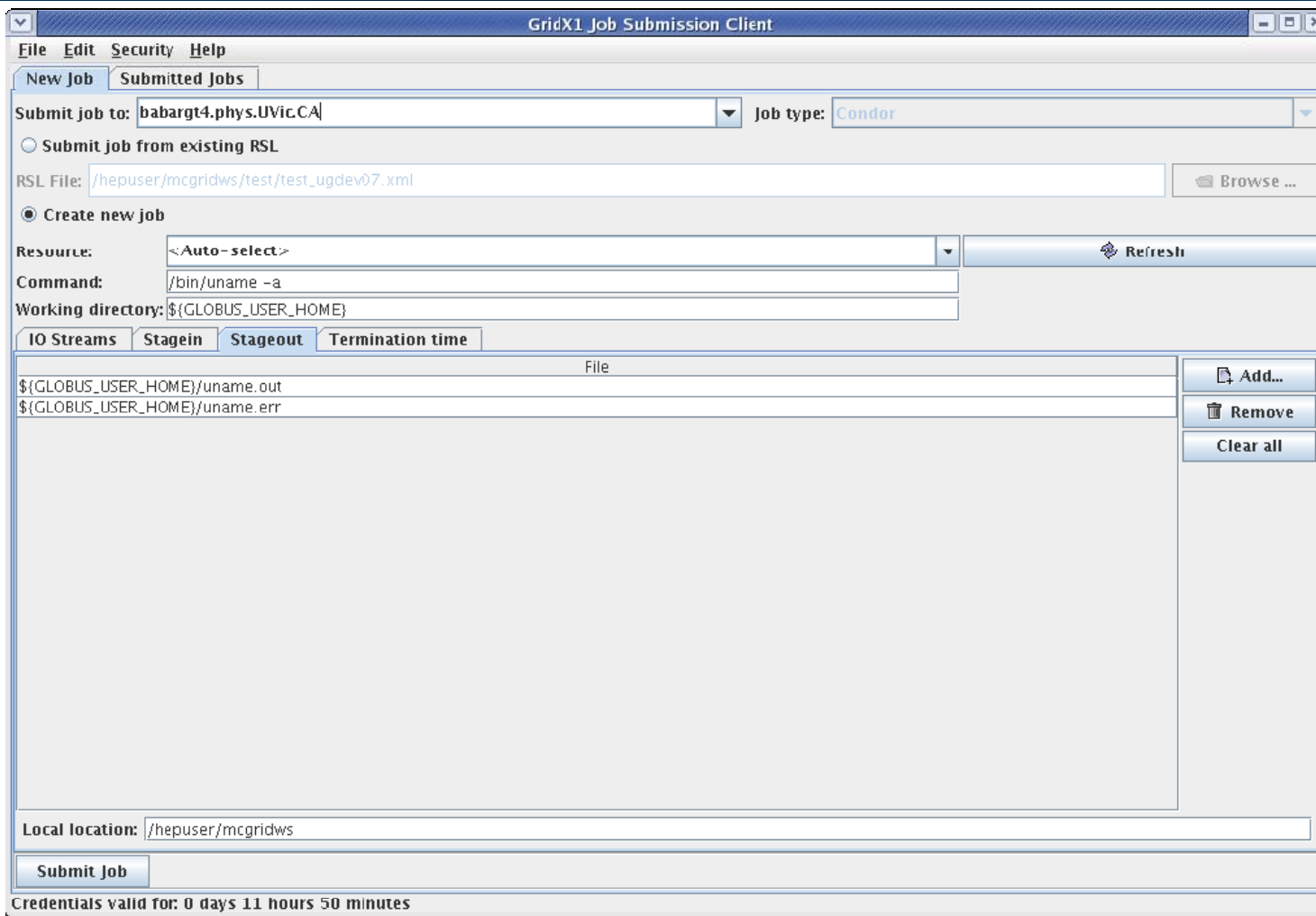
GridX1 Condor-G Monitoring

WSRF-Based Grid Monitoring

Conclusion

- With the WSRF (GT4), we have developed
  - A metascheduling service using Condor-G
  - Resource information provider
  - Automatic ClassAd extraction tool
  - Job submission client tool

- Execution of BaBar jobs is successful on the GT4 grid

- Web-based monitoring is useful for providing the status of grid resources and the jobs

- Monitoring is based on Condor_history. Work is in progress to improve monitoring using condor_quill.

- Production will start soon on this WSRF-based grid