Logging and Troubleshooting – An OSG Perspective

Anand Padmanabhan and Shaowen Wang

CyberInfrastructure and Geospatial Information Laboratory (CIGI)
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign

March 3, 2008

OSG End-End Troubleshooting – A Typical Procedure

- Check OSG monitoring services
  - E.g., VORS, site_verify, GIP, CEMon
- Get resource side logs
  - Sometimes request shell access
- Get client side logs
- Associate client and resource side logs
  - E.g., grep through log files
- Replicate and isolate problems

Resource Side

- Logs
  - Globus
    - GRAM, WS-GRAM and GridFTP
  - Batch systems / resource managers
  - CEMon
  - Apache & Tomcat
  - Gratia
  - GUMS/edg-mkgridmap
  - VDT-Install log
  - System logs (occasionally)
- Configuration files
  - osg-attributes.conf, gip-attributes.conf
  - osg-user-vo-map
  - osg-supported-vo, osg-undefefined-accounts
  - gums config, gridmap-file

Client Side

- Condor-G
  - Grid Monitor log
- stderr

How Are We Doing Troubleshooting?

- Well if we
  - Gain complete access to logs
  - Usually can find necessary logs that indicate problem sources
- Challenges
  - Takes time to get necessary logs
    - Manual process
  - Search and parse each log individually
  - Develop troubleshooting tools

NCSA

A Wish List

- Access log information through a SQL-like query interface
- Pull out full log entries as needed
- Link entries from multiple logs across resources and clients
  - Event-threaded
- DN-based access
- Zoom across various logging levels

NCSA

Open Questions

- How to handle client-side log files?
  - E.g. GridMonitor logs
  - Would it be possible to sporadically run syslog-ng modules during troubleshooting sessions?
  - Critical for end-to-end troubleshooting
- How difficult would it be to incorporate logs that do not follow defined logging structures?
  - How would syslog-ng handle local batch systems adaptively to their internal logging infrastructures?

NCSA

Thanks!

- Comments and/or questions?