Introduction to Condor

Steve Gallo
Lead Software Engineer
Center for Computational Research

What Is Condor?

- Developed at the University of Wisconsin
- Workload Management System
  - Queuing, Scheduling, Monitoring
  - Supports job ordering (workflow)
  - Designed to harness CPU cycles from idle workstations
- High Throughput vs. High Performance
  - Deliver large amounts of computational resources over a long period of time
  - Operations per month vs. FLOPS
- Resource matching via ClassAd mechanism

Features

- Does not require a shared filesystem
  - Optionally stage executables and data
- Application source code does not need to be modified
- ClassAd mechanism for matching jobs to resources
- Machine owner sets rules for when jobs run
- When linked with with Condor libraries
  - Remote system calls
  - Supports job migration and checkpointing
- Job ordering (via DAGMan)
  - Handles job dependencies

Condor Roles

- **Central Manager**
  - Collects information
  - Matches jobs to resources
- **Execute**
  - Runs condor jobs
- **Submit**
  - Users can submit jobs from this machine
- **Checkpoint**
  - Stores checkpoint files

Machine Preferences

- Each execute machine can set rules for who runs jobs on that machine
- Supports preferred users (e.g., users in my own group)
- Can restrict jobs to certain time slots (e.g., 11pm - 6am)
- Jobs are suspended if a local user sits down at the machine
- Jobs can be pre-empted for higher priority jobs
- Jobs can me migrated to other execute hosts

Matchmaking (ClassAds)

- Jobs specify what they require from a resource
  - CPU speed & architecture
  - Memory
  - Disk space
  - OS
- Jobs can specify preferences
  - I'll run on any machine but prefer those with 8+G of memory
- Resources advertise their services and condor selects those that best fit the job

Condor Universes

- Universes describe a condor execution environment
- Vanilla
  - Runs almost any application with few restrictions
- Standard
  - Requires linking with Condor libraries
  - Supports remote system calls and checkpointing
- Grid
  - Supports submission of jobs to remote grid resources
- Others…

Linking With Condor Libraries

- **Re-links application with Condor libraries**

  gcc main.o tools.o -o program

  becomes

  condor_compile gcc main.o tools.o -o program

- **Additional functionality provided**
- **Remote system calls**
  - **All job I/O calls are sent over the network to the submit host**
  - **Data does not need to be staged**
- **Checkpointing & Migration**
  - **Jobs can be checkpointed and migrated to another machine**
  - **Jobs will eventually complete even if a resource fails**

Condor-G

- Combines features of grid computing with Condor
- Grid features
  - Security
  - Cross-domain resource access
  - Existing grid resources
- Condor features
  - ClassAds
  - Job management features of Condor
  - Job interdependencies
  - Logging
  - Credential management (via MyProxy)

Condor and the Grid

- Condor has become the preferred way to submit OSG jobs
- Users take advantage of Condor job management capabilities to submit jobs to grid resources
  - Submit multiple jobs
  - Manage jobs
  - Completion/Failure notification
  - Maintenance of Globus credentials
  - Take advantage of ClassAds
- MyProxy integration
- Limitations
  - No checkpointing
  - No job exit codes (no different than Globus 2 or 3)

Condor Grid Monitor

- Improves scalability of grid jobs
- Reduces load on gatekeepers and submit hosts
  - Globus runs one jobmanager per job submitted (running or queued)
  - Perl scripts run every 10+ seconds to poll job state
  - 400+ jobs means a high gatekeeper load!
- Grid Monitor replaces globus job manager
  - Single monitor for multiple globus jobs
  - Reduces gatekeeper load
  - Gatekeeper must support jobmanager-fork

Resource Selector Service (ReSS)

- Provides Condor ClassAd service for OSG resoruces
- Included in OSG 0.8.0
- Resources use Generic Information Providers (GIP) to publish information about their resources
- ReSS converts this information into ClassAds
- Condor-G allows users to match job requirements to resources
- Helps users to select appropriate resources

Conclusion

- Condor provides
  - Job management
  - Workflow management
  - Resource selection
- Condor-G provides user-friendly features for submitting to grid resources
- OSG tools are integrating ClassAd mechanisms
- Improves ease of use