GlideCAF – A Late Binding Approach to the Grid

S. Belforte, S. C. Hsu, E. Lipeles, D. Lucchesi, M. Neubauer, S. Sarkar, I. Sfiligoi, F. Wuerthwein

Computing in High Energy and Nuclear Physics
13-17 February, 2006, TIFR, Mumbai, India
Talk Outline

- CDF Computing
  - Resources, requirement
- Moving towards the Grid Computing Paradigm
  - Supporting existing applications
  - Minimizing the impact on the end-users
- GlideCAF
  - Implementation, characteristics and status
- Summary
The CDF Analysis Farm (CAF)

1. Develop, debug and submit from personal PCs

2. Track your running jobs interactively

3. Satisfied? No need to stay connected anymore

4. Follow your jobs on the web monitor

5. On job completion receive an email

2 central farms in Fermilab
10 distributed farms worldwide

User selects the farm, usually
Fermilab for data analysis
Distributed farms for MC production

Output to any place
Most monitoring via regular web interface
- Both running and completed jobs
- System monitoring, including memory usage and CPU load
- Usage history graph
Condor System Overview

- **Schedd** manages user jobs
- **Negotiator** assigns nodes to jobs
- **Starter** manages jobs on the WN
- **Collector** gathers information about other daemons
Interactive Monitoring

Only outgoing connections from desktop

Only outgoing connections from worker nodes

Command Line
- list of jobs
- process list of a section
- listing the working directory
- tail
- debugging a process
CDF Computing Resources

- 2 CAFs at Fermilab
  - 2.6 M SPECint2000
- 10 dCAFs worldwide with ~50% computing load
  - 2.5 M SPECint2000
- Dedicated MC production farms, e.g. Karlsruhe

Computing Requirements:
Current: ~5 M SPECint2000
End of 2007: ~15 M SPECint2000

CDF priorities for computing outside Fermilab
- MC Production
- A few selected sites will become Physics Centres
  - CNAF, the Italian Tier1 Centre chosen for B Physics
CAF Evolution needed

Expansion of dedicated pools no more an option
Turn attention to shared resources

Add CDF resources to a Grid Pool

- Harvest batch slots for CDF using Condor with GSI authentication
  - CDF user jobs pulled onto such slots
- Add a new component to the CAF framework for resource management

Additionally, utilize the idle CPU cycles at the Grid Site
Use Condor glide-ins

**GlideCAF Overview**

A simple extension of the CAF

- glide-ins are regular, properly configured Condor starter daemons submitted as jobs to the Grid CE
  - indistinguishable from a dedicated one to Condor
  - will be matched in the same way to a user job with the best priority

Once a job starts on a WN, it notifies the collector and joins the pool as a new VM

retain all the virtues
Condor System - glide-ins

- Negotiator assigns nodes to jobs
- a secondary schedd manages the glide-ins

Globus assigns jobs to VOs
The glidekeeper is the glide-in factory

- fills the secondary schedd with glide-ins, as needed
  - \texttt{nr\_waiting\_jobs} > \texttt{nr\_queued\_glideins}
- manages glide-ins, e.g., removes held ones
GSI authentication all along

A single GSI CDF service proxy used for all the glide-ins
GlideCAF Details – CDF Jobs

- GSI authentication all along
- Condor knows the real users
- The Grid site knows only about the CDF service certificate
- All jobs run under a single VO specific UID (e.g. cdf001)
GlideCAF Deployment

- Production systems
  - CNAF Tier1 in Bologna, Italy
  - San Diego
  - Fermilab
  - Lyon Tier1 in France
- More GlideCAFs will appear soon
  - Karlsruhe, UCL
GlideCAF at CNAF

First prototype in April '05
In production since Sept '05
- CDF + glide-in resources
Pure GlideCAF since Dec '05

~ 670K jobs processed
~ 200 users
Negligible job loss

Use more than your share (~ 500)
Give back when others need it
GlideCAF Features

1. Late binding of resources
   - User jobs sent only when resources are available, i.e. glide-ins start and create VMs
   - Eliminate risk of long, indefinite wait at the CE queues for a multi-site configuration

2. Black hole removal
   - Defective nodes kill only glide-ins but no user jobs
   - Additional VO specific sanity checks can be performed
   - Nodes failing recurrently can be blacklisted
GlideCAF Features

- Fine grained policy management with two level negotiation
  - VO level at the Grid site
  - user level in Condor for each VO
- No Grid site specific installation required
  - Condor only environment
  - Only a small incremental change to the CAF infrastructure
- Monitoring continues to work natively
  - minor modification was needed
GlideCAF issues

- A single Grid certificate used for all the glide-ins
  - The site gatekeeper finds only one special user for that VO
  - no knowledge about the real users
- Looking at future evolutions of glExec for a solution
  - gets the GSI proxy and authenticates the real user
  - possibly changes the UID

- Real users authorized prior to job execution
- Sites can blacklist certain users
Plain GlideCAF Limitations

- **CDF Software** must be available on the WNs through a shared file-system (NFS, AFS)
- Collector and starters need good network
  - UDP based communication will not work over WAN
  - Require bi-directional traffic, will not work over firewalls
- A GlideCAF must be installed at each Grid site
  - **reasonable** for Large, CDF Friendly sites
    - CNAF, San Diego, Lyon
  - **unmanageable** if accessing scores of small sites
- See Matthew Norman's talk for GlideCAF extensions that lift the above restrictions
Summary

- CDF is able to use Grid resources
  - A number of GlideCAFs in production
- CAF infrastructure preserved
  - Transparent to end-users
  - Monitoring continues to work in a standard manner
- No new Middle-ware introduced
  - Just using Condor differently
  - Ensured a rapid turn-around
- GlideCAF extensions will take us further
Backup Slides
Computing Requirements

Data logging rate increases **3 times** from 2004 to 2006.

Event rate increases **4 times** due to better compression.

**Requirements:**

- **Current:** 5 M SPECint2000
- **End of 2007:** 15 M SPECint2000
Condor System - Condor-G

Monitoring would need re-implementation

- Globus assigns nodes to jobs
- All control at the Grid site
GSI authentication all along
The Web Server distributes glide-in binaries and site specific configuration to WNs

A single GSI CDF service proxy used for all the glide-ins