glideinWMS: Quick Facts

- glideinWMS is an open-source Fermilab Computing Sector product driven by CMS
- Heavy reliance on HTCondor from UW Madison and we work closely with them
- http://tinyurl.com/glideinWMS

Contributors include:
- Krista Larson (FNAL/Corral)
- Parag Mhashilkar (FNAL/Corral)
- Mats Rynge (ISI/USC/Corral)
- Igor Sfiligoi (UCSD)
- Doug Strain (FNAL)
- Anthony Tiradani (FNAL/CMS)
- John Weigand (CMS)
- Derek Weitzel (UNL)
- Burt Holzman (FNAL)
glideinWMS: Overview

VO Infrastructure

- VO Frontend
- Condor Scheduler
- Condor Central Manager

glideinWMS
Glidein Factory,
WMS Pool

Job

Condor Startd

Worker Node

Grid Site
glideinWMS: version timeline

- **2.4.x**: privilege separation, aggregate monitoring, glexec control, glidein lifetime control
- **2.5.x**: HTCondor TCP bulk updates, efficiency improvements, factory limits per frontend, excess glidein removal, shared ports, better user pool matchmaking
- **2.6.x**: Better multislot support, ARC CE, more glidein lifetime controls, factory limits per frontend security class
- **2.7.x**: Refactor for factory scaling, performance fixes, partitionable slot support
- **3.x**: Cloud support, CorralWMS frontend support

In 2013, v3 becomes production
glideinWMS: OSG Usage

- 2 Factories
- ~50K pilots peak
glideinWMS: CMS Usage

~23K Claimed
glideinWMS: Grid vs Cloud

VO Infrastructure

- VO Frontend
- Condor Scheduler
- Condor Central Manager

- Condor Startd
- glidein

Grid Site

- Worker Node
- glideinWMS VM

Cloud Site

- Condor Startd
- bootstrap
OpenStack

• Why OpenStack? [1]
  • If you want Amazon and VMware alternatives
  • OpenStack is Open
  • OpenStack matures at rapid pace
  • Its fundamental architecture is very sound
  • It’s driven by a diverse, huge community

• Note: We are not affiliated with OpenStack. We just use OpenStack

OpenStack: For Fun and Profit (1)

“People Get Pissed Off About OpenStack And That’s Why It Will Survive” [1]

OpenStack: For Fun and Profit (2)

- Very, very easy to overload the cloud controller
  - 1 node, 1 condor instance, 10 VM requests == DOS
- OpenStack doesn’t completely follow EC2 “standards”
  - Adds new VM states
  - Ignores “terminate on shutdown” flag
  - User data size limit undocumented and much smaller than Amazon
- OpenStack can be fragile internally
  - Many “broken pipe” messages in the nova api log
OpenStack: Improvements

- Submitted patch to OpenStack that brings the default user data size limit in line with Amazon
  - Should be in Grizzly release
  - https://bugs.launchpad.net/nova/+bug/1098646
- Thanks to Adam Huffman for discovering and assisting in troubleshooting efforts
glideinWMS: Adaptations

- Do not release any held jobs
  - For grid, we know some held reasons are transient, so we release jobs held with particular held reason codes
  - For Cloud, specifically OpenStack, all held reasons are fatal to the job

- Throttle polling frequency
  - Did I mention that it is very easy to overload the OpenStack controller?
  - We modify the factory HTCondor configs using existing knobs
HTCondor: Adaptations

- **Batch update requests**
  - Now HTCondor performs a DescribeInstances call with no filter to return all VMs requested and parses state internally
  - Note: OpenStack will return ALL VMs, regardless of who requested them

- **Handle SHUTOFF state**
  - OpenStack adds a new terminal state for VMs

- **OpenStack ignores “terminate on shutdown” flag**
  - InstanceInstantiatedShutdownBehavior
  - OpenStack will never terminate requested VMs automatically
  - HTCondor will explicitly terminate these VMs
HTCondor: Unexpected Features

- All ssh keys are deleted when HTCondor is restarted
  - We are currently running a 7.9.5 pre-release
  - Fixed in 7.9.5 official release? (Have to test)

- `condor_status –xml`
  - Rules changed for the escaping “\”
  - glideinWMS now uses a more robust regex
HTCondor: Enhancement Requests

- Batch VM requests
  - Request N vms in one call
  - When a workflow starts up there are few (if any) VMs running
  - We need to ramp up quickly
  - Currently we can only request one VM at a time
glideinWMS: Enhancement Requests

- Refactoring how pilots handle user data
  - Currently all user data must be in glideinWMS specific format
  - Pilot bootstrap service controls all startup and user data processes
  - Want to use CloudInit and make bootstrap a plugin to CloudInit
Other Considerations

- Must architect your network properly
  - CMS is limited to slightly under 1K jobs due to current network architecture
  - Have ~13k cores, but can’t use them all
  - Network reconfiguration will happen this week

- Tune your controller database carefully
  - One early bottleneck for OpenStack cloud controller was poorly tuned MySQL database
VM Image

BoxGrinder

Scientific Linux

European Middleware Initiative

Open Science Grid

XRootD

CernVM Software Appliance
Acknowledgements

- HTCondor Team
  - Jaime Frey
  - Todd Miller
  - Todd Tannebaum
  - Timothy St. Clair

- Adam Huffman (Imperial College)
Questions?