Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

GlideinWMS

Marco Mambelli

Stakeholders Meeting

July 11, 2018

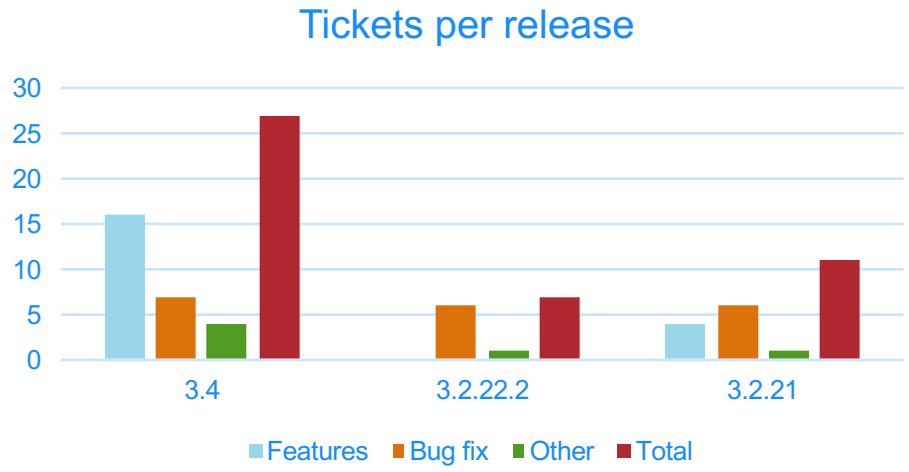
Overview

- Releases since last stakeholder's meeting
- Upcoming releases
- Current focus
- GlideinWMS roadmap
- Developers spotlight
- Reference slides
  - GlideinWMS Architecture
  - Quick Facts

Releases Since Last Stakeholders Meeting

- v3_4 released on June 4
  - Merging of production and development branches (v3.2 and v3.3), will bring Google CE support and policy plugin to the production version
  - Code modernization to Python 2.7 (and 2.6) standards
  - Increase number and coverage of the unit tests

- 16k lines code change
- Doubled unit test coverage
- More than doubled tests



Releases Since Last Stakeholders Meeting (cont)

- v3_4 released on June
  - Glidein lifetime not based anymore on the length of the proxy
  - New option to kill glideins when job requests decrease
  - Estimate in advance the cores provided to glideins discovering cores automatically
  - Add entry monitoring breakdown for metasites
  - Review Factory and Frontend tools, especially glidien_off and manual_glidein_submit.py

- Internal support of condor_switchboard (discontinued by HTCondor). glideinwms-switchboard 1.0 prepared. Will not be released in OSG

Next Planned Release

- v3_4_1 planned for end of July
  - Increase unit tests coverage to 30%
  - Track jobs that spawn multiple nodes, e.g. HPC submission
  - Improve Singularity support with recommendations form the meetings (better mount-points support, custom flags)
  - Update documentation removing references to Corral and GlideinWMS v2
  - Monitoring for frontend: store the number of Job restarts
  - Complete review Factory and Frontend tools, especially glidien_off and manual_glidein_submit.py
  - Fix configuration problem with entry_sets
  - Last version supporting Globus GRAM and last version with multi-user Factory

GlideinWMS: Current Focus (v3.4.1 and 3.5)

- Improve stability
  - More automated testing & CI (pylint, pythoscope, futurize, unittest …) is ongoing focus
  - Developer's test infrastructure to connect to Factory ITB services for scale testing
  - Test of new features on different sites in OSG
  - External contributions should be production ready
- Minimize wastage of resources from over-provisioning and improve auto-discovery
  - Improve handling of multi-node jobs
  - Auto - estimate of expected resources when provisioning
  - Actively follow the requests and adapt as the request goes down
  - Solution addressed in phases
    - First phase of the solution is available in v3.2.21, next in 3.4
    - Consider "transactional provisioning"
- Containerization
  - Singularity support changes
- Security
  - Adapt to sites with tighter security restrictions
    - Support for shorter proxy lifetime
- Move to single user Factory

GlideinWMS Roadmap

- Medium term (2018 – mid 2019)
  - Keep up with the scalability requirements
    - Investigate and incorporate new technologies like pandas dataframes, numpy, etc
  - Optimization of the interactions w/ HTCondor
  - Outsource GlideinWMS functionalities to HTCondor
    - Work with the HTCondor team to provide some of the Frontend functionalities natively through HTCondor
      - Leaner & modular Frontend
    - Adapt to changes/introduction of Acquisition Engine by HTCondor
      - Dependent on the work that will be done in HTCondor in the future
        - Very thin GlideinWMS Factory
  - Support for new HPC sites with stricter policies (e.g. no outbound connection except gateways, MFA)
    - Depends on support from HTCondor.
  - Monitoring Modernization
    - Retire GlideinWMS monitoring pages
    - Move to grafana/graphite/elastic search based solution

GlideinWMS Roadmap

- Long term (> mid-2019)
  - Move to Python 3
    - Start moving the code after v3.5 or following release
    - Have Python 3 version (v3.7) parallel to Python 2 version by end of Summer 2019
  - Move to Decision Engine (DE)
    - Replace the Frontend with the Decision Engine
  - Make Glidein as a service capable of talking to multiple WMS middleware/frameworks

Developers Spotlight

Lorena Lobato Pardavila - My focus on the project

+ Starting Point

- Familiarize with the GlideinWMS Environment
- Install GlideinWMS framework

+ Documentation

- Review, remove obsolete references and update information from the GlideinWMS documentation + Remove Corral documentation
- GlideinWMS ticket review from 2010 to do a first valuation and clarification about them and an importance

+ Review & Testing

- Review: Do not set GLIDEIN_ToDie based on X509 user proxy expiration
- Found issues with the proxy renewal script.

+ Development

- Condor_switchboard is being discontinued, we need a replacement
- Switch child collectors to shared_port
- Add a configurable limit to the rate of jobs running and fail the glidein if the rate is passed

  => http://condor-wiki.cs.wisc.edu/index.cgi/tktview?tn=6698

Lorena Lobato Pardavila - Summary

+ 4 **intensive months** trying to be an sponge

+ More knowledge about the system and already familiarized with the services to keep working in GlideinWMS development

+ Started to implement different features. Interaction with HTCondor , OSG, other teams in my division..etc

+ **Lot of work in documentation** taking advantage of being new comer

  - Lot of effort in **review** documentation and proposing **changes**

  - Spent high amount of time **writing** (helped to the growth)

  - Review of all tickets non-closed from GlideinWMS project since 2010

+ Review and testing of co-worker's work

  - **Fast-up learning** about GlideinWMS and the dependency services

+ Personally I enjoy more the work on **reviewing** and **system analysis**

  - Like to break things ☺

Dennis Box

- Recent Focus has been on code testing/stability
  - Unit tests
  - Integration Tests
  - Misc Code quality tools
- Unit Tests:
  - generate coverage report
  - Use ‘pythoscope’ to generate skeletons of missing tests
  - Skeletons turned into real unit tests
    - Use libraries such as ‘hypothesis’ to fuzz-test input
- Went from 16% to 35% coverage so far
- Coverage reports can be browsed by release at
  - https://home.fnal.gov/~dbox/

Dennis Box

- Integration Tests
  - Automated ‘base line’ or ‘smoke’ test for new releases
  - Verify that rpm install, upgrade, submission works for all combinations
- Misc Code Quality Tools
  - The project is 27000 lines of python and 11000 lines of bash
  - Python code quality tools are mature (autopep8, futurize)
  - Bash is more problematic
    - Shellcheck is best linter found so far
    - Unit testing for bash is difficult to make realistic

Dennis Box

- Lessons learned
  - Unit test generation can be (somewhat) automated
    - Still labor intensive
    - Find/sed/awk work nearly as well as pythoscope
    - Valuable in that it forces you to read the code
  - Dead python code (ex VDT, GUIs) should be pruned
  - Some data structures will require care to convert to python 3

Marco Mascheroni - Factory Ops Requests

- Received list of requests from factory ops
  - Factory ops requests summarized in redmine
- Monitoring
  - Add entry breakdown for metasites
  - Provide json for external monitor integration
- Miscellaneous feature requests
  - Improve handling of glideinCPU=AUTO setting (with EstimatedCpus)
  - Add a scaling factor for all glideins limits in the entries
- Better management of factory queues
  - Periodic remove of long running glideins
  - Improve handling of held pilots
- Review/cleanup/fix tools
- Optimization items (quality of life)
  - Do not restart condor on “service gwms-factory upgrade”
  - Command to cleanup config files from old entries
  - Remove old files to speed up stop/reconfig/restart

Marco Mascheroni - Factory Ops Requests

- Received list of requests from factory ops
  - Factory ops requests summarized in redmine
- Monitoring
  - Add entry breakdown for metasites
  - Provide json for external monitor integration
- Miscellaneous feature requests
  - Improve handling of glideinCPU=AUTO setting (with EstimatedCpus) - in 3.4, working on deployment
  - Add a scaling factor for all glideins limits in the entries <= Still in progress, should make it
- Better management of factory queues
  - Periodic remove of long running glideins
  - Improve handling of held pilots <= Needs decision after feedback from condor devs
- Review/cleanup/fix tools
- Optimization items (quality of life)
  - Do not restart condor on “service gwms-factory upgrade”
  - Command to cleanup config files from old entries
  - Remove old files to speed up stop/reconfig/restart

Will be available in 3.4.1
Moved to 3.5

Marco Mascheroni - Conclusions

- Major items have been taken care of and will be in 3.4.1
- Few minor/low priority things left behind
  - Will take care in 3.5
- Will take care of new requests as they come
  - Some space to start working on something new/big
- Possible items:
  - Took care of scaling limitations in the frontend emerged during CMS scale tests [20302]
  - Automatic generation of GMWS configuration from CRIC (discussions started at CHEP)
  - Other major items if something come in

Jack Lundell, Metcalf Intern

- Undergraduate at the University of Chicago
  - Majoring in Computational and Applied Mathematics, with a minor in Physics
- Hired as an intern for the Summer to create a profile of GWMS queries to HTCondor
  - Objective: determine projection, constraint and frequency of queries, and calculate associate timing statistics
  - Goal: to learn about High Throughput Computing, to improve my software development skills, to identify and remove bottlenecks and unnecessary queries in GWMS interactions between the Frontend, the Factory and HTCondor

Thomas Hein - GlideinWMS Monitoring System

- GlideinWMS provides monitoring on both a Factory and Frontend level
- The project currently uses a RRDB (via RRDtool) for record keeping
- This does not easily port over to time series visualization tools such as Grafana

Fermilab

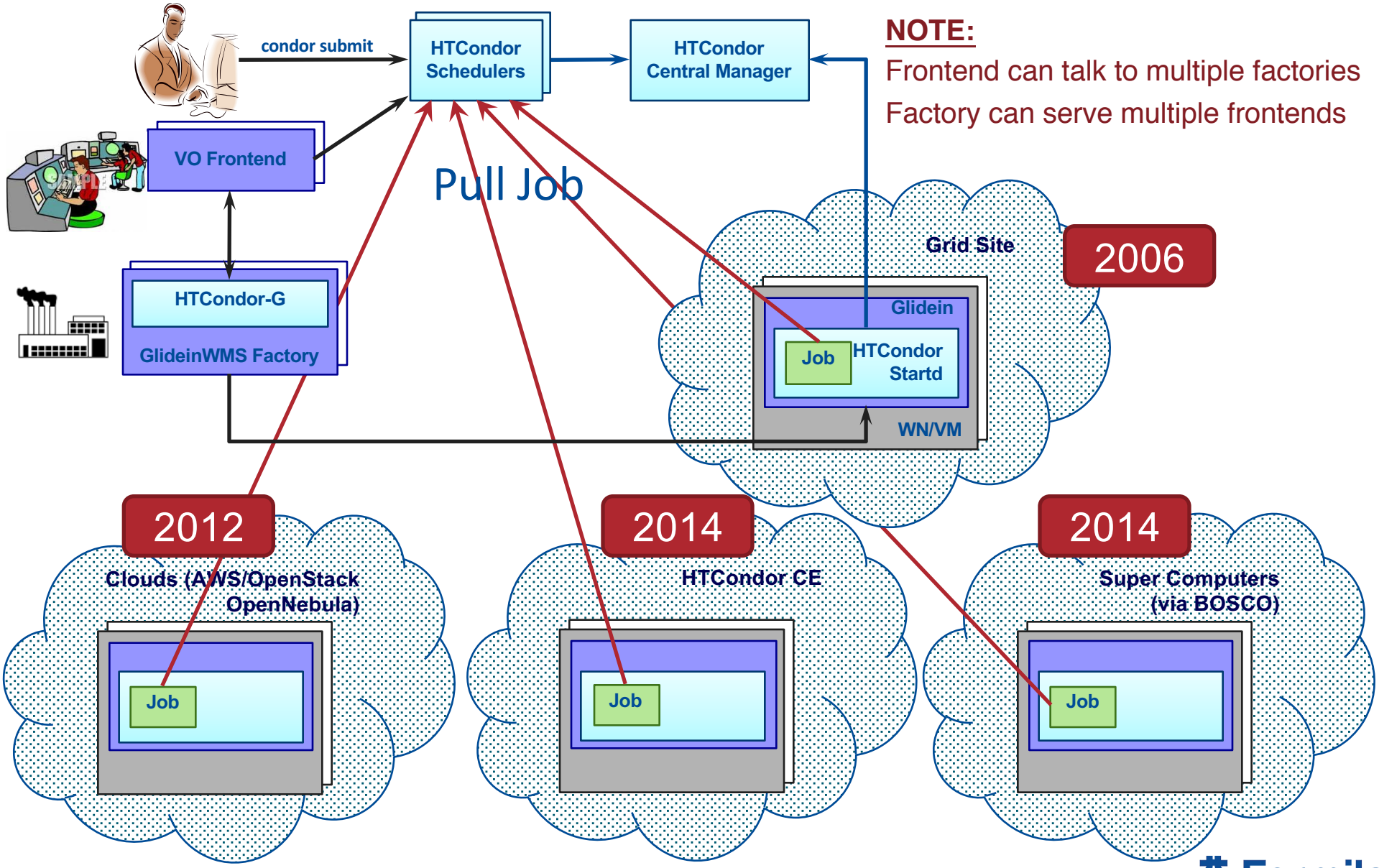
T Hein - GlideinWMS Monitoring System (cont.)

- The goal is to incorporate a more popular time series database
- This database should be able to easily connect to visualization tools such as Grafana
- Collect the same statistics that RRDtool collects
- Time permitting
  - Add more relevant statistics for collection
  - Connect the database to Fermilab's Grafana instance Landscape

Questions/Comments

Reference Slides

GlideinWMS


condor submit
HTCondor Schedulers
HTCondor Central Manager
NOTE:
Frontend can talk to multiple factories
Factory can serve multiple frontends
VO Frontend
Pull Job
HTCondor-G
GlideinWMS Factory
Grid Site
2006
Glidein
Job
HTCondor Startd
WN/VM
2012
Clouds (AWS/OpenStack OpenNebula)
Job
2014
HTCondor CE
Job
2014
Super Computers (via BOSCO)
Job

GlideinWMS: Quick Facts

- GlideinWMS is an open-source product (http://tinyurl.com/glideinWMS)
- Heavy reliance on HTCondor (UW Madison) and we work closely with them
- Effort:

Role	Resources	Effort (FTE)
Project Mgmt/Lead	Parag Mhashilkar (0.15 USCMS)	0.15
Development & Support	Marco Mambelli (1 SCD) Dennis Box (0.25 SCD) Lorena Lobato Pardavila (1 SCD) Marco Mascheroni (0.5 CMS - Contractor)	2.75
TOTAL		**2.90**

Table: Current Resources & Roles

Quick Facts: Releases & Support Structure

- Releases
  - Issues tracked in redmine issue tracker
    - https://cdcvs.fnal.gov/redmine/projects/glideinwms/issues
    - Categorized and prioritized based on impact, urgency and requester
      - Issues are now associated with respective stakeholders
    - Issues are assigned based on developer's expertise and other workload
    - Roadmap for upcoming releases available in redmine (See reference slides)
  - SCM
    - All releases are version controlled and tagged
    - http://glideinwms.fnal.gov/doc.prd/download.html
  - Release notes & history
    - http://glideinwms.fnal.gov/doc.prd/history.html
- Support
  - Entire development team is responsible for support

Quick Facts: Project Status & Communication Channels

- Project meeting: Wednesdays 10 – 11 am
  - Technical discussions & status updates
  - Regular stakeholder participation
  - Contact Parag Mhashilkar if you need invite for this meeting
- Stakeholders Meeting every two months
- Project Management
  - Project Status reported monthly at CS Project status meetings

Area of Interest	Mailing Lists
Support	glideinwms-support@fnal.gov
Stakeholders	glideinwms-stakeholders@fnal.gov
Release Announcements	glideinwms-support@fnal.gov cms-dct-wms@fnal.gov glideinwms-stakeholders@fnal.gov
Future Release plans	See next slide
Discussions	glideinwms-discuss@fnal.gov
Code commits	glideinwms-commit@fnal.gov Twitter Tag: @glideinwms

Tracking Releases in Redmine