

GlideinWMS - Feature #22843

Develop a Testing framework for distributed resources and GlideinWMS

06/28/2019 04:40 PM - Marco Mambelli

Status:	New	Start date:	06/28/2019
Priority:	Normal	Due date:	
Assignee:	Marco Mambelli	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	v3_6_x	Spent time:	0.00 hour
Stakeholders:			
Description			
<p>Build a testing application for distributed resources (entries) and GlideinWMS. This application will interact with GlideinWMS components and send Glideins to explore and validate the resources. It will further allow to customize the Glideins and replace standard components of the GlideinWMS framework with new ones than the user decides to test.</p> <p>This project includes participation in the following activities:</p> <ul style="list-style-type: none">• Provide input in the design of the testing application• Getting familiar with distributed computing and GlideinWMS• Development and integration related to distributed computing software for Grids, Clouds and Supercomputers (this will include Python and Web development):<ul style="list-style-type: none">◦ Adapting components to talk the GlideinWMS protocol◦ Adding a UI to drive the testing• Testing on clusters, High-Performance Computers and clouds• Integration and release <p>Some shortcomings of the current system that should be improved and some suggestions:</p> <ul style="list-style-type: none">• No flexible way to send pilots to specific resources• No way to substitute standard scripts• Manual client checking which resources are available, what's the status and letting the user decide how many pilots to send and where• The code written should be in Python 3 (specifically 3.6)• Consider migrating some of the code in GlideinWMS (e.g. protocol sending ClassAds) using python 3.6 and htcondor python bindings <p>Here a project doing something similar with shell scripts: https://cms-htcondor-monitor.web.cern.ch/cms-htcondor-monitor/_book/emulated_pilot/ https://gitlab.cern.ch/CMSSI/cms_emulated_glidein</p>			
Related issues:			
Related to GlideinWMS - Milestone #22673: Summer interns 2019		New	06/03/2019 09/30/2019

History

#1 - 07/03/2019 05:17 PM - Marco Mambelli

- Related to Milestone #22673: Summer interns 2019 added

#2 - 07/09/2019 01:11 PM - Javier Rodriguez

- % Done changed from 0 to 70

#3 - 07/10/2019 09:24 AM - Javier Rodriguez

- % Done changed from 70 to 100

- Status changed from New to Resolved

#4 - 07/15/2019 08:52 AM - Javier Rodriguez

- % Done changed from 100 to 80

- Status changed from Resolved to Assigned

#5 - 07/25/2019 03:52 PM - Marco Mambelli

Here a concrete task:

Start with `manual_glidein_submit.py`

Add the possibility to modify the parameters used in the submission (params dictionary) by reading them from a file in INI format. The file name will be specified in a command option, e.g. `--parameters-file FILE_NAME`, where `FILE_NAME` is the path to the INI file

Suggestion:

<https://docs.python.org/2/library/configparser.html>

Step 2 (more difficult)

Use the YAML format instead of INI and run the `manual_glidein_submit` with `python3`.

This may require some changes in the `GlideinWMS` code imported.

<https://learn.getgrav.org/16/advanced/yaml>

<https://pyyaml.org/wiki/PyYAMLDocumentation>

#6 - 08/05/2019 02:37 PM - Marco Mambelli

- % Done changed from 80 to 0

- Target version set to `v3_5_x`

- Assignee changed from *Javier Rodriguez* to *Marco Mambelli*

- Status changed from *Assigned* to *New*

Javier worked on a change to `manual_glidein_submit.py` to allow extra attr options loaded from an ini file.

The ticket is being opened for future re-assignment

#7 - 10/04/2019 04:57 PM - Marco Mambelli

- Target version changed from `v3_5_x` to `v3_6_x`