

GlideinWMS - Feature #15280

Support auto for memory just like we do for cores

01/23/2017 01:38 PM - Parag Mhashilkar

Status:	New	Start date:	01/23/2017
Priority:	Normal	Due date:	
Assignee:	Marco Mambelli	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	v3_5_3	Spent time:	0.00 hour
Stakeholders:			

Description

Brian Bockelman asked for this feature. Idea is to allow for auto discovery of memory available to a glidein when it startups. This is to minimize the number of entries required in the factory to support WN or different configurations for site with larger WN. This will allow for more dynamic setup.

This also suffers from same challenges and drawbacks for auto

- Frontend is no longer in the position to understand the WN configuration to decide the number of glideins required to satisfy the demands
- Auto detection of memory can only be supported in the batch system HTCondor and one or two other batch systems
- Unlike cores/cpus where we auto can default to 1 there is no good default value we can use but it may be possible to assume 2GiB or 2GB per core

To solve some of these problems, we may need to look at this from a different perspective

- Can OSG ensure that required info is always available in the glidein job's environment so we do not need to depend on job system to provide this info and will guarantee that this info is available to glidein irrespective of the batch system
- To get around the issues with the frontend accounting, we may want to look into the possibility of additional config information that will help frontend with required information in case of auto
 - GLIDEIN_CPUS -> GLIDEIN_CPUS & GLIDEIN_DEFAULT_CPUS
 - GLIDEIN_MaxMemMBs -> GLIDEIN_MaxMemMBs & GLIDEIN_DEFAULT_MaxMemMBs

History

#1 - 04/26/2018 04:45 PM - Marco Mambelli

- Target version changed from v3_3_x to v3_4_x

#2 - 07/09/2018 02:28 PM - Marco Mambelli

- Target version changed from v3_4_x to v3_5

Amount of memory per Glidein (this means that is split across all jobs running on the Glidein if it is partitionable or statically split)

Review the current variables and behavior.

If possible bring it to something similar to the CPUs:

GLIDEIN_CPUS (number, auto, node, slot) GLIDEIN_ESTIMATED_CPUS

I.e.

GLIDEIN_MaxMemMBs (number, auto, node, slot, htc) GLIDEIN_ESTIMATED_MaxMemMBs

(instead of GLIDEIN_MaxMemMBs int and GLIDEIN_MaxMemMBs_Estimate boolean)

- auto, same as slot (consistency w/ _CPUS, value could not be there)

- slot, memory that the batch system assigns to the slot

- node, HW memory of the whole machine

- htc, let HTCondor decide (current default, used when GLIDEIN_MaxMemMBs not set and GLIDEIN_MaxMemMBs_Estimate not true) - should we have something similar in _CPUS?

Another possibility for the name is

GLIDEIN_ESTIMATED_MEMORY, GLIDEIN_ESTIMATED_MEMORY

and maybe allow different units? MB/GB

Check w/ Factory ops if the change in behavior/name is OK.

If units would be a welcome addition

In the process make sure that the behavior and documentation are consistent (GB vs GiB)

Verify that the memory can be set to an arbitrary number like the CPUs
Verify that the behavior is correct for the auto/node/slot values of GLIDIEN_CPUS

#3 - 11/12/2018 05:06 PM - Marco Mambelli

- Target version changed from v3_5 to v3_5_1

#4 - 08/16/2019 10:17 AM - Marco Mambelli

- Target version changed from v3_5_1 to v3_5_2

#5 - 09/17/2019 11:15 PM - Marco Mambelli

- Target version changed from v3_5_2 to v3_5_3