Release Notes

Gold Allocation Manager

Version 2.2.0

Copyright © 2010 Adaptive Computing

This document provides an overview of the changes made in a minor release of Gold Allocation Manager. A minor release will generally include new functionality and features that can break compatibility with previous versions. See the Migration section for migration procedures.

Table of Contents

Overview

This release of Gold Allocation Manager features enhanced charge rate types that give more flexibility in the way charges can be calculated. Other changes were made to allow for better support of incremental charging and cost determination. Command-line clients were added that aid in the importing and manipulation of job records.

New Features

The following is a list of new features added in this release.

- Additional Charge Rate Types — This release added a new Instance field to the ChargeRate object in support of an expanded set of charge rate types (See the chapter on Managing Charge Rates in the Gold User's Guide for details). The following new charge rate types replace the former types (Resource, Usage, Multiplier, other):

  - VBR — Value Based Resource (formerly Resource) [These charge rates define how much it costs per unit of time to use a consumable resource like processors, memory or a telescope, etc.]
  - NBR — Name Based Resource [These charge rates define how much it costs per unit of time to use a named resource such as license=matlab, etc.]
  - VBU — Value Based Usage (formerly Usage) [These charge rates define how much it costs for total resource usage such as cputime, power consumed, generic resources or licenses that are charged flat fees per use, and which will be subject to multiplier charge rates.]
  - NBU — Name Based Usage [These charge rates define how much it costs to use a named attribute having a flat charge such as feature=black, and which will be subject to multiplier charge rates.]
  - VBM — Value Based Multiplier (formerly Multiplier) [These are scaled multipliers which apply a multiplicative charge factor based on a numeric scaling factor (such as Discount=.8).]
  - NBM — Name Based Multiplier (formerly other) [These are quality based multipliers which apply a multiplicative charge factor based on a named quality of the job such as quality of service, nodetype, queue, user, etc.]
  - VBF — Value Based Fee [These charge rates define how much to charge for scaled or enumerated fees such as setup fees, shipping charges, etc. which should be added after the multipliers are applied.]
  - NBF — Name Based Fee [These charge rates define how much it costs to use a named attribute having a flat charge such as feature, etc. which should be added after the multipliers are applied.]
  - <other> — Multi-dimensional Value Based Resource [These charge rates apply a consumable resource cost that varies depending on the value of a separate named job property.]

- New commands to import and manipulate job records — Three new command line clients have been added to import and manipulate job records:
  - gmkjob — Import or create a new job record
  - gchjob — Modify a job record

  - grmjob — Delete a job record

- Distinction between charge properties and job properties — Prior to Gold 2.2, WallDuration, StartTime, EndTime and Description options specified to gcharge, greserve and gquote were taken as both charge and job properties, making it difficult to support incremental charging. In Gold 2.2, these properties have now been disambiguated. The duration, starttime, endtime and description for the charge will be passed via the standard options (-t, -s, -e, -d) while the same properties for the job (duration, starttime, endtime and description) can be passed in separately using the extension option (i.e. -X WallDuration=1234). See the man pages for gcharge, greserve and gquote for additional details.
- CostOnly Quote Option — A job quote now supports a new costOnly option which will return a cost estimate without performing balance checking or other qualifying validation. This is useful if you would like to know how much something would cost, but you are not interested in establishing that a user has an allocation, or belongs to a specified project, etc.
- Improved Client Performance — The gold server was modified to preload the utf8 module (instead of doing so after the fork for each individual client request). This has been shown to give an 10-20% speedup for common scheduler action command invocations. This improvement is also available in the gold 2.1.12.2 fix release.

Migration Procedures

The following recommendations should assist you in migrating from an existing Gold 2.1 installation to a new Gold 2.2 installation.

- Default prefix changed to /opt/gold — In Gold 2.2, the default Gold installation location has been changed to /opt/gold (from /usr/local/gold). Thus, if you were previously running ./configure without the —prefix option to install Gold into /usr/local/gold, when upgrading to Gold 2.2 you will need to explicitly specify your location now with ./configure —prefix=/usr/local/gold in order to have Gold installed in the same place.
- Database migration script — Run the test/bank_2.1_to_2.2.pl script to migrate your Gold 2.1 database to the new Gold 2.2 format. It will perform the following steps:
  - Add a Job Stage Value of Create and add Job Create as a Scheduler Role Action
  - Create new ChargeRate and QuotationChargeRate Instance Attributes
  - Translate Existing ChargeRates to new format (VBR, NBU, etc.)
- Scheduler action commands now use gold_job_id as default argument — The optionless argument in gquote, greserve and gcharge has been changed to Id ([-j] <gold_job_id>) instead of JobId (-J <job_id>) which now must be use an explicit -J flag. In Gold 2.1 and earlier, gcharge 123 has been interpreted to mean a charge against the resource manager job id 123. As of Gold 2.2, gcharge 123 will be interpreted to mean a charge against the gold job id 123. Any scripts (such as a prolog or epilog) which make use of these commands should be modified accordingly.
- Disambiguation of charge and job properties in scheduler action commands — In Gold 2.2, in order to better support incremental charging, the duration, starttime, endtime and description for the charge will be passed via the standard options (-t, -s, -e, -d) while the same properties for the job

(duration, starttime, endtime and description) may be passed in separately using the extension option (i.e. -X WallDuration=1234). If you are are using prolog and epilog scripts to invoke greserve and gcharge, and you are charging for the entire job (not incremental charging), then if you were using gcharge -t 1234 -s "2010-07-01 12:00:00" -e "2010-07-01 12:20:34", in Gold 2.2 you would have to use gcharge -t 1234 -X WallDuration=1234 -X StartTime="2010-07-01 12:00:00" -X EndTime="2010-07-01 12:20:34" to accomplish similar results (the -t option would be used in the charge calculation to determine cost, while the -X options would be sent in as job metadata for the job accounting records).