

Complete Qube Release Notes

Qube 8.0

```
#####
#
# Qube Release Notes
#
#####

#####
@RELEASE: 8.0-0

==== CL 23707 ====
@NEW: code to add a line that allows "qube" user to connect to DB from any machine on the network, with its
password

See also CL23705,23706

==== CL 23706 ====
@NEW: add qubedb_prep_0055.sql that sets the "qube" DB user's default password.

See also CL23705

==== CL 23705 ====
@CHANGE: Assign default password for "qube" DB user.
* Allow, in pg_hba.conf, access of "qube" user with password from non-localhost
* Make UI connect to DB over the network using the "qube" user

==== CL 23698 ====
@NEW: add python 3.9 and 3.10 support on macOS

==== CL 23680 ====
@FIX: TypeError when options.chunk unspecified

==== CL 23635 ====
@NEW: upgrading DB version to 55-- add qubedb_0055.sql .

==== CL 23608 ====
@CHANGE: make sure the *_sk columns (synthetic keys) in the fact tables are of "integer" data type instead
of "smallint"

==== CL 23600 ====
@NEW: add datawh/upgrade_scripts/upgrade_v4.sql

* also add datawh/alter_sk_to_int.sql which is included by the upgrade_v4.sql script to convert all *_sk
columns from smallint to int data type

ZD: 21636

==== CL 23596 ====
@NEW: add Shotgun (Shotgun) Python API 3.3.3 to qube-core package.

==== CL 23591 ====
@NEW: add Python API routine qb.jobtagvalues() that fetches and returns a list of existing values for job
tags (such as for "prod_show", "prod_seq", ..., "prod_custom1",...)

JIRA: QUBE-3924

==== CL 23589 ====
@FIX: datawh/upgrade_scripts/upgrade_v2.sql hadn't "SET ROLE dwh", resulting in some DBs created with
incorrect ownership

==== CL 23560 ====
@NEW: add C++ API routine "qb.jobtagvalues()" to fetch a list of existing values for job tags (such as for
"prod_show", "prod_seq", ..., "prod_custom1",...)

JIRA: QUBE-3924

==== CL 23516 ====
@NEW: Add Python 3.10 Qube API support (Linux and Windows)
```

JIRA: QUBE-3928

==== CL 23512 ====

@NEW: Add python 3.9 Qube API support (Linux and Windows)

JIRA: QUBE-3926

==== CL 23493 ====

@NEW: add API routines "qbbannedworkers()" (C++) and "qb.bannedworkers()" (Python) to return a list of all currently banned (removed/blacklisted) worker hosts.

* also added a "--listbanned" option, to "qbadmin w".

JIRA: QUBE-2074

==== CL 23475 ====

@CHANGE: Upping DATA_WAREHOUSE_VERSION to 3 in file "datawarehouse_version"

JIRA: QUBE-3920

==== CL 23473 ====

@NEW: add datawh/upgrade_scripts/upgrade_v3.sql . So far, adds calls to new external .sql scripts that create the new DWH subjob and work fact tables and initially populate them.

JIRA: QUBE-3920

==== CL 23472 ====

@NEW: dwh: add work fact table and related SQL code to populate it initially and on a regular basis.

JIRA: QUBE-3920

==== CL 23471 ====

@NEW: dwh: add subjob fact table and related SQL code to populate it initially and on a regular basis.

JIRA: QUBE-3920

==== CL 23470 ====

@FIX: add code to TRUNCATE the worker_dim and cluster_dim tables in populate_slotCount.sql where it initially populates those tables

==== CL 23453 ====

@FIX: tab char used for indentation in Python "helloWorld" jobtype example

==== CL 23419 ====

@CHANGE: add support for Perl v5.30 (especially for macOS 11.3.1+, and 11.5.2+) in api/perl/qb/qb.pm

ZD: 21313

JIRA: QUBE-3916

==== CL 23385 ====

@FIX: Message lengths over 254 resulting in Null characters being sent

==== CL 23363 ====

@CHANGE: removed Qt dependencies from qube core (especially the API)

JIRA: QUBE-3902

ZD: 21232, 21293

==== CL 23341 ====

@NEW: add (back) pyWorkCmdline jobtype, now python3 compatible

JIRA: QUBE-3896

ZD: 21197

==== CL 23340 ====

@CHANGE: convert python-based jobtype pyWorkCmdline from python2 to python3

==== CL 23324 ====

@FIX: data warehouse (dwh) upgrade_v2.sql script had issues with schema scoping, causing supervisor installation to prematurely fail

==== CL 23323 ====

@FIX: syntax error with the upgrade_v2.sql script for datawh, resulting in partially failed supervisor installation

```

==== CL 23290 ====
@CHANGE: support Apple-provided Python 3.8, instead of Home Brew version

==== CL 23262 ====
@CHANGE: postgresql startup scripts no longer control where the logs go (now it's specified in
postgresql.conf)

JIRA: QUBE-3888

==== CL 23261 ====
@NEW: Add Qube-custom logging parameters to postgresql.conf. Now PGSQL logs are written to
DATADIR/pg_log/pgsql.log on ALL platforms.

Formerly, log wasn't written to a file on Windows.

JIRA: QUBE-3888

==== CL 23260 ====
@FIX: some parsing issues in configure_postgresql_conf.py script

* bug where commented out parameters would end up in the end section of the postgresql.conf file, unless
the commented out value exactly matches our value

* fixed regex to match better (had issues with params with empty value '', and with whitespace)

==== CL 23259 ====
@UPDATE: add signal handler for segfault in QbPreForkDaemon to print stack trace upon the said signal

==== CL 23258 ====
@FIX: a couple of issues with python scripts that export MySQL data and import it into PostgreSQL

* export script: quoting issue with the path to MySQL executable
* import script: jobid data must be imported before job data to satisfy a constraint added since 7.5-0

ZD: 21118

==== CL 23253 ====
@FIX: binarySort Python 3 compatibility

==== CL 23227 ====
@FIX: proxy program (proxy.exe) crashing under certain Windows environments

ZD: 21090

==== CL 23203 ====
@UPDATE: Edits for database_checks.py

==== CL 23201 ====
@FIX: Encoding xdrlib integers 128 and greater

==== CL 23163 ====
@CHANGE: make sure that the "pfx" db is created with its Encoding set to 'UTF8'

JIRA: QUBE-3865

==== CL 23160 ====
@NEW: Adding back the slotcount_fact table as well as the cached "data subset" tables to the Data Warehouse
(dwh)

JIRA: QUBE-3867
ZD: 20996

* Add back, to dwh, the slotcount_fact table and all related .sql scripts:
create_hostState_dim.sql
create_slotCount_fact.sql
create_worker_dim.sql
populate_slotCount.sql
regular_slotCount.sql

* Also add upgrade_scripts/upgrade_v2.sql, and updated datawarehouse_version to 2.

In upgrade_v2.sql, reimplemented the PFX_CREATE_DATASUBSET_TABLE function (that was available in our MySQL
dwh), and also added commands to source the create*.sql and populate_slotCount.sql files mentioned above to
build the necessary tables.

```

* Add back cron jobs (Linux, macOS) and scheduledTask (Windows) that periodically run the regular_slotCount.sql collector, and build the "data subset" tables that contain data for a limited time-range (12hr, 36hr, 7day, 3wk, and 3mo) for faster preset queries for charting:

Linux: qube/etc/cron.d/com.pipelinefx.DataWarehouse.cron

macOS: add back qube/datawh/data_collectors/osx*.sh scripts in the data_collectors subdir, and their corresponding macOS cron drivers, qube/etc/com.pipelinefx.DataWarehouse.*.plist, qube/pkg/supepkg.pl and qubepkg.pm to package them up into the supervisor installer

Windows: changes made to qubemsi.pm and rrd_tables.bat to install and enable the scheduledTasks when the supervisor MSI installer is run

==== CL 23146 ====

@FIX: Python API: qb.rangepartition(), and consequently qb.genpartitions(), broken with python3, returning an empty list to valid input.

ZD: 20992

JIRA: QUBE-3863

==== CL 23052 ====

@FIX: include file path to the python execution module in the call to exec() so that qb.backend.utils.getModulePath() works properly

Also modified internals of getModulePath() in qb/backend/utils.py and qb/util/__init__.py to be more compatible across python versions

ZD: 20941

==== CL 23034 ====

@FIX: wildcard and regular expression (regex) filtering features in tools such as "qbjobs" broken since 7.0

The filtering features now work as described in doc at:

<http://docs.pipelinefx.com/display/QUBE/Using+Wildcards+and+Regular+Expressions>

JIRA: QUBE-3844

==== CL 23022 ====

@FIX: Python-based jobtype backends make undesirable calls to qb.reportjob(), rendering instance-based postflight return values useless

pyCmdrange, pyCmdline, appFinder, and pyCmdrangeGPU jobtypes were making a call to qb.reportjob() at the end of their execute.py scripts which is undesirable, and causing the return values of instance-based postflights to be meaningless. For example, a job instance could "complete", but the postflight that runs right after it could return non-zero indicating that the instance should be marked "failed". However, since the job instance was reporting "complete" via its own call to qb.reportjob() before running the postflight, the status that was amended by the postflight never took effect (it was too late to reach the supe).

JIRA: QUBE-3836

==== CL 22906 ====

@NEW:Add VRED 2021 support

==== CL 22873 ====

@FIX: additional fixes for "DataWH collectors aren't run by cron on Ubuntu"

JIRA: QUBE-3846

==== CL 22872 ====

@FIX: DataWH collectors aren't run by cron on Ubuntu

JIRA: QUBE-3846

==== CL 22845 ====

@FIX: add openssl 1.1.1h lib files to qube core package, which are needed by the Qt 5.14.2 network module used by the DRA.

JIRA: QUBE-3833

==== CL 22828 ====

@FIX: supervisor's embedded python3 interpreter fails running callbacks with error: ModuleNotFoundError: No module named '_struct'

ZD: 20850

JIRA: QUBE-3832

```
==== CL 22748 ====
@FIX: issue where some files/folders won't be deleted upon MSI uninstall.

==== CL 22735 ====
@FIX: Uninstalling supervisor on Windows now removes the postgresql software (but preserves the data
directory).

@FIX: Windows: the 7zip self-extraction of PostgreSQL software throws an "ERROR: can not delete output
file" error when upgrading supervisor

@CHANGE: Installation of supervisor on Windows now first makes sure that the previous installation of
postgresql is removed.

JIRA: QUBE-3819, QUBE-3817

ZD: 20811, 20816

#####
```

Qube 7.5

```
#####
#
# Qube Release Notes
#
#####

#####
@RELEASE: 7.5-0

==== CL 22602 ====
@NEW: add one more file for initialization of the central preferences database, qubedb_prep_0054.sql, which creates the "prefs" DB user.

JIRA: QUBE-3795

==== CL 22597 ====
@CHANGE: implemented code to make changes to postgresql.conf and pg_hba.conf files on installation, needed to support the new central
preferences feature.

JIRA: QUBE-3795

==== CL 22596 ====
@CHANGE: "pfx" account's default password changed to a longer one (for new installations, except on Linux)

JIRA: QUBE-3667

==== CL 22593 ====
@NEW: add SQL to initialize the central preferences database.

JIRA: QUBE-3795

==== CL 22592 ====
@FIX: init_supe_db.py: make all calls to the "psql" command using the DB owner

==== CL 22458 ====
@CHANGE: point PYTHONPATH to $QBDIR/lib/python3.8 before supervisor service is started, for its embedded python interpreter (Linux,
macOS)

==== CL 22288 ====
@NEW: add "disable_central_prefs" flag to supervisor_flags

JIRA: QUBE-3778

==== CL 22230 ====
@FIX: a bunch more fixes to make python-based backends to work properly with python3 while maintaining python2 compatibility.

Now if a python-based jobtype's job.conf specifies "execute_binding = Python" or "execute_binding = Python3", python3 will be used. If
"execute_binding = Python2", python2 is used.

JIRA: QUBE-3747

==== CL 22190 ====
@CHANGE: Switch supervisor's embedded Python interpreter to python3.8.
```

JIRA: QUBE-2762, QUBE-3749

==== CL 22179 ====

@CHANGE: made Linux installation (RPM and DEB for CentOS/RHEL and Ubuntu, respectively) require "python3"

JIRA: QUBE-3767

==== CL 22177 ====

@CHANGE: convert python-based jobtypes (appFinder, pyCmdline, pyCmdrange) qube/types/ from python2 to python3

JIRA: QUBE-3747

==== CL 22176 ====

@CHANGE: convert example python scripts in qube/examples/python from python2 to python3

JIRA: QUBE-3746

==== CL 22175 ====

@CHANGE: convert python scripts in qube/scripts from python2 to python3

JIRA: QUBE-3745

==== CL 22174 ====

@CHANGE: convert python scripts in qube/utills from python2 to python3

JIRA: QUBE-3745

==== CL 22081 ====

@FIX: "pfx" user to be created w/o a home directory now, in the install_supervisor script, which is used to do some initialization on DEB-based Linux platforms (i.e. Ubuntu).

Was previously set up to create a home dir, causing the DEB qube-supe installation to exit prematurely when the root user doesn't have write permissions to create "/home/pfx" (e.g. NFS-mounted /home).

Now the "useradd" command points to "/var/tmp" as the pfx user's homedir.

==== CL 22080 ====

@NEW: add perl 5.30 support (for Ubuntu 20.04 LTS)

JIRA: QUBE-3721

==== CL 22077 ====

@FIX: "pfx" user to be created w/o a home directory now, in the install_supervisor script, which is used to do some initialization on RPM-based Linux platforms.

Was previously set up to create a home dir, causing the RPM qube-supe installation to exit prematurely when the root user doesn't have write permissions to create "/home/pfx" (e.g. NFS-mounted /home).

Now the "useradd" command points to "/usr/tmp" as the pfx user's homedir.

==== CL 22057 ====

@NEW: Ubuntu 18.04 and 20.04 support

JIRA: QUBE-3720, QUBE-3721

==== CL 22039 ====

@NEW: Add Python 3.7 (standard) and 3.8 (homebrew) API support on macOS

==== CL 22035 ====

@NEW: Add Python 3.6, 3.7, and 3.8 API support for Windows.

JIRA: QUBE-2762

==== CL 22030 ====

@NEW: Add python 3.6, 3.7, and 3.8 Qube API support on Linux.

==== CL 21802 ====

@CHANGE: macOS to build Qube core with Qt 5.14.2

JIRA: QUBE-3688

==== CL 21801 ====

@CHANGE: remove python 2.6 support from all platforms

JIRA: QUBE-3691

==== CL 21800 ====

@CHANGE: Linux to build with Qt5.14.2

JIRA: QUBE-3688

==== CL 21769 ====

@NEW: add Python 3.8 compatibility to the main Qube Python API , including its supporting .py scripts.

JIRA: QUBE-2762

==== CL 21731 ====

@FIX: Fix crash when no options are given. Now usage message is printed when no args are present.

@CHANGE: Made the "checks" arguments instead of options.

@INTERNAL: refactored a bunch of stuff.

JIRA: QUBE-3206

==== CL 21644 ====

@FIX: not all child processes of job instances sometimes not dying properly when parent thread dies

ZD: 20225

==== CL 21641 ====

@FIX: not all child processes of job instances sometimes not dying properly when parent thread dies

ZD: 20225

==== CL 21556 ====

@FIX: fixed problem where a job can get stuck in "dying" state due to a timing-related issue.

This was causing, among other things, global resources to not be released properly.

ZD: 20307

==== CL 21432 ====

@CHANGE: install_supervisor script: install Data W/H DB by calling \$QBDIR/datawh/install_datawarehouse_db.sh

==== CL 21385 ====

@TWEAK: Don't give up on the first error in enableRequiredPrivileges(), but try enabling all privileges. Also print number of errors.

==== CL 21360 ====

@FIX: Aggressively preempted frames can get missed and left in "pending" while instances all finish

ZD: 20177

==== CL 21351 ====

@CHANGE: add SE_DEBUG_NAME to list of privileges to be enabled; also add more info to print to workerlog

* add SE_DEBUG_NAME to the list of privileges to be enabled

* print WARNING when OpenProcess() fails in cleanup(), and the reason

* add instance name to print in more output lines when available/applicable

==== CL 21242 ====

@FIX: On host reboot, supervisor needs to start after postgresql is started and ready

* added code to check the DB connection at supervisor's boot time, and retry after 10 seconds, up to 6 attempts (1 minute), effectively delaying the supervisor boot until after the DB is ready.

JIRA: QUBE-3637

==== CL 21239 ====

@NEW: add a way to tell qbjobinfo() API routine to only query for and pull selective job data (aka "columns" or "fields").

Developers using the Qube C++ and/or Python API can now tell qbjobinfo() routine (qb.jobinfo() for Python) to only query for and pull selective job data (aka "columns" or "fields"), for leaner, meaner, more economical queries.

* Add support for explicitly specifying needed fields in C++ API's qbjobinfo().

* Add support for explicitly specifying needed fields in Python API's qb.jobinfo(), a la 6.10's direct query API.

* Also add "-fields" option to qbjobs

* qbjobs now makes leaner queries by default (unless an option to display details is specified, like "-long" or "-notes")

[Examples]

C++:

QbString query_fields_str = "id,username,status";

QbStringList query_fields;

QbExpression::split(query_fields_str, query_fields);

QbQuery query;

```
for(int i = 0; i < query_fields.length(); i++) {  
  QbField *f = new QbField(*query_fields.get(i));  
  query_fields().push(f);  
}  
QbJobList jobs;  
qbjobinfo(query, jobs)
```

Python:
jobs = qb.jobinfo(fields = ['id','username','status'])

JIRA: QUBE-3623
ZD: 19955

==== CL 21234 ====
@FIX: Add timeout for agenda-based jobs stuck in "running" status, in a "waiting" loop.

TL;DR:
Sometimes, agenda-based job instances can get stuck in the "running" state, in a "wating" loop. A timeout, currently hardcoded to 60 seconds, has been added to force those jobs to break out of the loop.

Details:
Sometimes, agenda-based job instances can get stuck in a "wating" loop, with messages like the following repeating indefinitely in the job's stdout:

```
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: requesting work for: 424805.0  
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: got work: -1: - waiting  
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: INFO: informing worker[127.0.0.1]  
INFO: told to wait & retry from supe-- sleeping for [7] seconds
```

A job instance stuck in this state can tie up a worker's job slot(s) until it is manually intervened with (killed, migrated, etc), or until it hits its "subjob timeout" (assuming the job was setup with it).

This issue, newly introduced in 7.x, has been found to happen due to race conditions.

It is particularly likely to occur when the following conditions are met:
* jobs have the migrate_on_frame_retry job flag set AND they use retrywork/retrysubjob
* job instances fail quickly (i.e. job process/renderer crashes and exits quickly)
* there are idle workers

(There are other scenarios that this can also happen, such as when aggressive preemption is done rapidly, but there's normally not many idle workers when preemptions do happen, so it's less likely.)

In a nutshell:
* instance fails on a worker
* supe detect the failure, migrates and starts the instance on a new worker
* the new worker reports the instance now "running"
* the first worker finishes cleaning up and reports that the instance is now "pending"
* instance gets stuck in a "wating" loop on the new worker.

A timeout, currently hardcoded to 60 seconds, has been added to force those jobs to break out of the infinite loop.

ZD: 19977, 20094, 19967
JIRA: QUBE-3638

==== CL 21217 ====
@FIX: Add timeout for agenda-based jobs stuck in "running" status, in a "waiting" loop.

TL;DR:
Sometimes, agenda-based job instances can get stuck in the "running" state, in a "wating" loop. A timeout, currently hardcoded to 60 seconds, has been added to force those jobs to break out of the loop.

Details:
Sometimes, agenda-based job instances can get stuck in a "wating" loop, with messages like the following repeating indefinitely in the job's stdout:

```
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: requesting work for: 424805.0  
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: got work: -1: - waiting  
[Dec 20, 2019 18:23:46] HOSTNAME[47572]: INFO: informing worker[127.0.0.1]  
INFO: told to wait & retry from supe-- sleeping for [7] seconds
```

A job instance stuck in this state can tie up a worker's job slot(s) until it is manually intervened with (killed, migrated, etc), or until it hits its "subjob timeout" (assuming the job was setup with it).

This issue, newly introduced in 7.x, has been found to happen due to race conditions.

It is particularly likely to occur when the following conditions are met:

- * jobs have the migrate_on_frame_retry job flag set AND they use retrywork/retrysubjob
- * job instances fail quickly (i.e. job process/renderer crashes and exits quickly)
- * there are idle workers

(There are other scenarios that this can also happen, such as when aggressive preemption is done rapidly, but there's normally not many idle workers when preemptions do happen, so it's less likely.)

In a nutshell:

- * instance fails on a worker
- * supe detect the failure, migrates and starts the instance on a new worker
- * the new worker reports the instance now "running"
- * the first worker finishes cleaning up and reports that the instance is now "pending"
- * instance gets stuck in a "waiting" loop on the new worker.

A timeout, currently hardcoded to 60 seconds, has been added to force those jobs to break out of the infinite loop.

ZD: 19977, 20094, 19967
JIRA: QUBE-3638

==== CL 21060 ====

@FIX: bug where jobs passively preempted while working on the final agenda item don't complete properly but go "pending" with the agenda items 100% done

JIRA: QUBE-3626
ZD: 19967

Also:

@TWEAK: made IP address to also print, in addition to the hostname, when qbwk.conf for a host is loaded (QbSupervisor::loadWrkConfig()). Now prints something like:
loaded config for host: hostname (aaa.bbb.ccc.ddd)

==== CL 21059 ====

@INTERNAL CHANGE/FIX: Removed the special, undocumented "feature" where the "str" passed to "QbField::value(str)" may optionally be prefixed with a special character to specify an operator (or "sign") to be applied for the field.

The special char was one of [~,=,>,<,%,*], and was setting "sign" to a string representing the ASCII value of the op character ("itoa" value, for example '%' to "37").

Instead of just fixing that, we're ditching this special feature, as it is unnecessary and confusing. The operator can always be specified explicitly by calling QbField::sign(str). A scan of the qube code base didn't reveal any use of this feature, but it's possible that peripheral code (namely python apps, like WV or AV) may be using it (but I highly doubt it).

==== CL 20817 ====

@FIX: fix_mysql_column_orders.sql: add back AUTO_INCREMENT to columns ('id' or 'uq') of the following MySQL tables: globalcallback, globalevent, jobid, lostevent

The ALTER TABLE done to modify column orders on these tables were wiping the AUTO_INCREMENT of these tables' specified columns. This was in turn causing issues (job submissions failing, for example) when downgrading 7.0 to 6.10.

ZD: 19833

==== CL 20683 ====

@FIX: qubeproxy on Linux does not use the same standard default password as it does on macOS/Windows

JIRA: QUBE-613

==== CL 20681 ====

@NEW: Add ability to lock/unlock workers by MAC address

JIRA: QUBE-243

==== CL 20658 ====

@CHANGE: Use of FK (foreign keys) in Postgres for job removal.

- * Optimized job removal.
- * Added utils/pgsql/qubedb_0054.sql which will "ALTER TABLE" (via init_supe_db.py) the relevant job-related tables.

JIRA: QUBE-3319

==== CL 20650 ====

@FIX: Setting a *_flags param to an empty string should mean "no flags", not "default flags"

Specifically, the supervisor_job_flags would not behave as expected, and would take on the default values when specified as:

```
supervisor_job_flags = ""
```

or

```
supervisor_job_flags =
```

JIRA: QUBE-620

==== CL 22713 ====

@CHANGE: Disabling "query" in supervisor_verbosity by default, to avoid overcrowding the supelog with the frequent (2 or more per second) "job/host query received" messages generated by the new supervisorProxy queries

```
#####  
#  
# The Qube! Supervisor Proxy Release Notes  
#  
#####
```

The Qube! Supervisor Proxy:

Watches for changes to jobs and workers on the Qube! supervisor and pushes those changes out to the Qube! UI clients running on the network. This reduces load on the supervisor in large farms and the UIs get interactive updates for the jobs and workers panels.

```
#####  
@RELEASE: 7.5-0  
#####
```

What's new:

The Qube! Supervisor Proxy is now available on all Qube! supported platforms.

Qube 7.0

```
#####  
@RELEASE: 7.0-2d  
This maintenance release includes several key fixes (especially on Windows)  
and some changes to the supervisor, core, and worker, and is a recommended  
update to all 7.0-x customers.
```

==== CL 20637 ====

@FIX: install_supervisor script: remove attempt (which fails) to install Data W/H DB by calling \$QBDIR/datawh/install_datawarehouse_db.sh

==== CL 20631 ====

@FIX: "qbadm s --config" doesn't return "supervisor_idle_threads" when that parameter is commented out in qb.conf

JIRA: QUBE-2653

==== CL 20622 ====

@FIX: qbjobs "-flags" option broken

JIRA: QUBE-3520

==== CL 20611 ====

@CHANGE: removed submission check for zero workers (i.e., jobs now submit fine when there are no workers on the farm)

==== CL 20584 ====

@FIX: minor memory leak fix in PCRE usage in QbRegEx.cpp.

==== CL 20582 ====

@FIX: add back perl 5.10 and 5.12 support for CentOS/RHEL 6.x .

ZD: 19384

==== CL 20496 ====

@FIX: modified the way job instances are killed on Windows workers. The old way was causing issues (i.e. jobs not kill-able) in some environments where the worker is running in service mode AND GPO is used.
Some related symptoms/indications included:

* Seeing the following error message in the workerlog:

ERROR: QbWorker::killJob(), PostThreadMessage() Invalid thread identifier

* "qblock --purge <worker>" not killing off locally running job instances as expected.

* Aggressive preemptions not working as expected.

Note: this was previously worked around by tweaking the "UIODetect" (aka Interactive Services Detection service) setting in the registry, but Windows 10 update 1803 removed the service, thus disabling this workaround.

ZD: 19473, 19457, 19485

==== CL 20432 ====

@NEW: added "-chunk <int>" option to job_cleanup.py script, to allow specifying of the query size

ZD: 19307

==== CL 20431 ====

@NEW: add (exposed) new options to Perl API qb.jobinfo(): minid, maxid, limit, and orderby

ZD: 19307

==== CL 20409 ====

@TWEAK: Python API: modified what gets printed when JobValidator.validate is called with verbose flag set

==== CL 20407 ====

@NEW: add new options to qbjobs: -limit, -minid, -maxid, -orderby

@FIX: qbjobs: made "-u" option an alias to "--user"

JIRA: QUBE-3492

==== CL 20356 ====

@FIX: Windows 10: proxy program crashes near end of instance execution, causing it to become "failed"
On some Windows 10 environments, proxy.exe would crash close to the very end of execution, causing any job to fail, even though the job process ran fine.

ZD: 19224

JIRA: QUBE-3493

==== CL 20354 ====

@NEW: add --dbowner=name option so that init_supe_db.py may be used to initialize the DB using a user other than the default "pfx"
Also added "-n|--noexec" option, to allow dry-runs.

JIRA: QUBE-3471

ZD: 19069

==== CL 20347 ====

@FIX: add code to fix column ordering of Qube DB tables in MySQL, which may have gotten mixed up if the DB was ever upgraded, and can cause the data transfer from MySQL to PGSQL to fail

JIRA: QUBE-3474, QUBE-3475

ZD: 19055

==== CL 20346 ====

@FIX: minor, mostly cosmetic error in the logic for callback execution code where it determined whether a given callback language was disabled or not

#####

@RELEASE: 7.0-2c

This maintenance release includes a few key fixes to the supervisor, core, and worker, and is a recommended update to all 7.0-x customers.

Specific to Windows, it also includes a change where all modules and binary files have been rebuilt against a version of Perl that has been built in-house from source.

IMPORTANT: A note about Perl versions on Windows workers

Perl v5.26 is the only supported version on Windows, and must be installed on all workers. All other versions of Perl are unsupported.

==== CL 20342 ====

@CHANGE: made sure installed Perl versions other than 5.26 are detected as "unsupported" and properly rejected as such (Windows).

==== CL 20338 ====

@FIX: Proxy program was responding to orders (such as migrate/interrupt/preempt) prematurely, before the operations had completed, causing race conditions in some (rare) cases, resulting in issues such as agenda-based job instances going into an infinite "wait" loop.

ZD: 19066

==== CL 20303 ====

@FIX: bug in a couple of SQL queries causing issues, such as tons of unnecessary extra auto-complete operations.

==== CL 20302 ====

@FIX: qlogin password registration failing with "ERROR: passwords don't match" on Windows 10

ZD: 18963

JIRA: QUBE-3439

@RELEASE: 7.0-2b

This is a supervisor-only release that fixes a critical bug introduced in 7.0-2a. Recommended for any site currently using a 7.0-x supervisor.

==== CL 20255 ====

@FIX: issue introduced in 7.0-2a supe where agenda-based jobs will not dispatch to instances.

==== CL 20252 ====

@NEW: add README-DB-VERSIONING.txt file, for informational purposes only.

JIRA: QUBE-3190

@RELEASE: 7.0-2a

#####

This is a supervisor-only release that includes a few key fixes, and is recommended for any site currently using a 7.0-x supervisor.

==== CL 20233 ====

@FIX: timeout value set via the API routine qbsettimeout() now respected more accurately

ZD: 18524

JIRA: QUBE-3181

==== CL 20223 ====

@FIX: race-condition can dispatch the same agenda item to multiple instances

ZD: 18980

==== CL 20222 ====

@FIX: "running_monitor" background thread/routine would sometimes try to verify instance-worker combinations that don't exist in reality (and sometimes even non-existent instances), then put those instances back to "pending".

ZD: 18960

==== CL 20218 ====

@FIX: supervisor install fails on Windows where .py files are not associated with the python interpreter

==== CL 20215 ====

@FIX: issue where Supe fails to register dependencies of some jobs at submission, due to DB ERROR: duplicate key value

JIRA: QUBE-3427

ZD: 18957

#####

@RELEASE: 7.0-2

#####

==== CL 20172 ====

@FIX: Multiple false matches for 'regex_outputPaths' when an instance is preempted

==== CL 20151 ====

@FIX: qbhosts, qbjobs, qblock, and qbmodify commands' "-group" option not working as expected.

Note: required a change to both the client programs as well as the supervisor (had a incompatible SQL statement).

JIRA: QUBE-3413

==== CL 20131 ====

@CHANGE: add code to check supervisor_max_threads relative to the DB server's max_connections, and adjust it if it's too large.

"Too Large" is: supervisor_max_threads > max_connections - 25

JIRA: QUBE-3396

==== CL 20115 ====

@FIX: Calling qbstdout or qbstderr API (c++ or Python API) with "pos=-N" on a file of N - 1 bytes or smaller can crash supervisor process

JIRA: QUBE-3145

==== CL 20113 ====

@FIX: qbjobs should be case-insensitive for user name; qbhosts should be case-insensitive for host names

JIRA: QUBE-3320, QUBE-3386

==== CL 20108 ====

@FIX: fix issue where Qube 7 Perl API module doesn't load on Linux, causing Perl based jobs such as the Maya jobtype job not be able to run.

Error messages:

Can't load '/usr/local/pfx/qube/api/perl/qb.so' for module qb: libQt5Core.so.5: cannot open shared object file: No such file or directory at /usr/lib64/perl5/DynaLoader.pm line 190.

Added /etc/ld.so.conf.d/qube-x86_64.conf file to the qube-core package, to add the path /usr/local/pfx/qube/lib/Qt to the runtime library search path

JIRA: QUBE-3403

ZD: 18920

==== CL 20102 ====

@FIX: 'DB ERROR: relation "joblog" does not exist' message in supelog

Fixed issue where ERROR messages like the following would show in the supelog:

QbDatabasePostgreSQL::_raw() DB ERROR: lastError=ERROR: relation "joblog" does not exist
LINE 1: SELECT id, jobid, subid, type, data, host FROM joblog WHERE...

JIRA: QUBE-3186

ZD: 18899

==== CL 20094 ====

@FIX: Qube API module for python 2.6 (_qb26.pyd) not loading

JIRA: QUBE-3387

==== CL 20081 ====

@FIX: On the Mac, the "pfx" user is created every time the supervisor package is installed or upgraded (add_pfx_account_osx.sh script).

#####

@RELEASE: 7.0-1c

#####

Added support for CentOS-7.4 and CentOS-7.5 in this release

==== CL 20071 ====

@FIX: agenda items failing due to timeout won't auto-retry

ZD: 18874

==== CL 20070 ====

@FIX: 7.0-1 supe installer won't work on platforms with python 2.6 (issue with init_supe_db.py script)

JIRA: QUBE-3379

==== CL 20028 ====

@FIX: querying supervisor config never returns database_port unless it's explicitly defined

JIRA: QUBE-3357

==== CL 20025 ====

@FIX: extremely inaccurate cumulative cpu time for agenda items

JIRA: QUBE-3375

ZD: 18841

@RELEASE: 7.0-1a
#####

==== CL 20002 ====

@NEW: add new filtering options to Python API's `qb.jobinfo()`:

- `submittedBefore`
- `submittedAfter`
- `updatedBefore`
- `updatedAfter`

Args specifying time may be in seconds since Unix epoch, or a `datetime.date` or `datetime.datetime` object.

Examples:

```
import qb
qb.jobinfo(updatedAfter=1530680480, updatedBefore=1530680700)

import datetime
weekago = datetime.date.today() - datetime.timedelta(days=7)
qb.jobinfo(submittedAfter=weekago)
```

JIRA: QUBE-3353

==== CL 19999 ====

@NEW: add new filtering options to `qbjobs` command: `submittedBefore`, `submittedAfter`, `updatedBefore`, `updatedAfter`. Option arg specifying time needs to be in seconds since Unix epoch.

JIRA: QUBE-3353

@RELEASE: 7.0-1
#####

This release includes key fixes to the supervisor, core and WV, among other fixes, and is strongly recommended for any site currently using 7.0-0 or above.

==== CL 19967 ====

@FIX: include support for earlier versions of Perl. Now we support Perl 5.14 thru 5.26.

==== CL 19964 ====

@CHANGE: adjust (Windows) perl support to versions 5.14 thru 5.26

JIRA: QUBE-3368

==== CL 19957 ====
@FIX: update perl API's copy of IPRC::Run module to properly support Perl 5.26

==== CL 19938 ====
@FIX: Python 2.6 module (_qb26.pyd) was not included. Python modules were named with a .dll extension, instead of the more preferred .pyd

JIRA: QUBE-3358

==== CL 19937 ====
@FIX: Python API .py files missing and/or incorrectly located w/o proper directory structure under QBDIR/api/lib/python

JIRA: QUBE-3358

==== CL 19935 ====
@CHANGE: ignore if database_port is set to 3300 or 3306 and revert to the default, 50055.

JIRA: QUBE-3359

==== CL 19934 ====
@FIX: init_supe_db.py: add version-aware execution of qubedb_xxxx.sql files

JIRA: QUBE-3336

==== CL 19933 ====
@FIX: add important missing INDEXes to the qube.subjob DB table for performance boost.

==== CL 19931 ====
@FIX:Fix relative movie paths in images_to_move.py

==== CL 19915 ====
@CHANGE: remove direct SQL access from job cleanup script
@CHANGE: find_corrupt_jobs.py removed from qube installation, it checked for the existence of tables which are no longer in the Qube 7 schema

==== CL 19897 ====
@FIX: auto_remove worker flag missing from worker config dialogs

==== CL 19888 ====
@FIX: issue where Windows password couldn't be updated.

==== CL 19849 ====
@CHANGE: add the postgres-based database checks back into the supervisor installers

==== CL 19836 ====
@NEW: convert database_checks cmdline utility from MySQL to PostgreSQL

@RELEASE: 7.0-0a

#####

==== CL 19831 ====
@FIX: jobs submitted that reserves a global resource never runs

Bug introduced at 7.0-0, where jobs specifying any global resource reservation would be stuck indefinitely in "pending" state.

JIRA: QUBE-3328

#####

@RELEASE: 7.0-0

#####

==== CL 19673 ====

@CHANGE: allow Metered Licensing (ML) with just a valid, unexpired supe license, and no worker license

JIRA: QUBE-2823

==== CL 19637 ====

@FIX: Perl API: added proper qb::version() support

==== CL 19636 ====

@NEW: add support for Perl 5.18, 20, 22, 24, and 26 on Windows.

JIRA: QUBE-749

==== CL 19529 ====

@NEW: add paexec.exe and ntrights.exe to aid with proper installation of postgresql server

==== CL 19507 ====

@NEW: add com.pipelinefx.postgresql.plist file to enable launchd support of PostgreSQL DB server on macOS

JIRA: QUBE-3100

==== CL 19504 ====

@NEW: add_pfx_account_osx.sh script that creates the "pfx" account that runs the PostgreSQL DB server.

The script is run from the "postinstall" process of the pkg installer.

JIRA: QUBE-3100

==== CL 19478 ====

@FIX: workers are always "auto-remove"d, even if "auto_remove" is not set in worker_flags.

ZD: 18512

JIRA: QUBE-3174

==== CL 19475 ====

@FIX: issue where instances would be stuck in "QB_PREEMPT_MODE_FAIL", causing the supervisor to tell instances to "wait and retry later" in response to retryWork() indefinitely.

Issue was caused when the preemptJobNetwork() routine determines that the instance has started but has NOT yet started working on an agenda item, in which case it would mark the QB_PREEMPT_MODE_FAIL in order to interrupt (i.e. aggressively preempt) the instance; However, the interrupt was not being triggered properly.

Issue was apparently introduced in CL19126.

==== CL 19462 ====

@FIX: issue where some daemon (supe/worker) threads exit early, after processing less client requests than specified via max_clients (e.g. 65, not 256).

Early exits should now only happen when "max threads" happened earlier.

==== CL 19457 ====

@TWEAK: add a couple of useful supelog lines to pring in assignjob(), regarding result of calling converseWorker() for dispatch

==== CL 19454 ====

@FIX: add call to sendHostReport() so that a statusHost message is sent to the supe when the worker "received kill order for unassigned job". This should eliminate some of the jobs that stay in "dying"(or allow "kill" of jobs that are stuck in "dying")

==== CL 19443 ====

@NEW:Add KeyShot commandline render script for batch rendering

==== CL 19437 ====

@FIX: workid is not duplicated by QbHistory copy constructor

==== CL 19436 ====

@FIX: "down" workers not always detected properly

JIRA: QUBE-3155

ZD: 18425

==== CL 19425 ====

@FIX: issue when supe thread doesn't hear back from worker during a dispatch. Related to CL19243.

Also fixed an issue (probably harmless) where an extra call to queue.releaseJob() was sometimes made in the findSubjobAndReserveJob() method.

==== CL 19415 ====

@CHANGE: add qbsub support for jobtypes "pyCmdrange" and "pyCmdline".

==== CL 19263 ====

@FIX: log directories for jobs submitted after the utility has been started but before the orphaned log removal is begun are erroneously removed

==== CL 19258 ====

@FIX: not running --use-frm when first-pass repair fails when message has different line-endings than OS X

==== CL 19243 ====

@FIX: add code to avoid mixed-up job instance status when worker-supervisor communications are dropped during job dispatch on an intermittently unreliable network

It was found that network hiccups can cause a worker to not respond to the supervisor during the dispatch of a job instance, but still start running the instance anyway. The worker would send the "running" instance report to the supervisor, which is processed by a separate thread, which updates the DB, causing a status mix-up.

Added code to detect such situations, and allowed the system to let the job run (instead of force-removing it from the duty table) on the worker in question.

Also added error-checking code on the worker side-- if worker detects that it couldn't respond to the supe for a dispatch order, it will give up on that job and release resources that it had just reserved for it.

ZD: 17868

==== CL 19236 ====

@FIX: jobs submitted by non-admin user without a specified priority attempt to submit at priority -1

JIRA: QUBE-3015

==== CL 19209 ====

@FIX: "down" workers would not be detected properly by the supervisor even when the supervisor_heartbeat_timeout expired.

ZD: 18057

JIRA: QUBE-3018

==== CL 19178 ====

@FIX: timing issue causing workers to get stuck with job instances.

Issue was seen on a very busy farm with intermittently drops in network communications, when many supe threads would try to dispatch a single instance at the same time.

ZD: 17868

==== CL 19164 ====

@CHANGE: On Unix, by default, supe uses a Unix domain socket to connect to the PostgreSQL server, unless the "database_host" parameter is set.

The default value of database_host is "" on Unix (Linux/macOSX), and "localhost" on Windows.

==== CL 19163 ====

@FIX: fix an issue where a worker can sometimes get stuck with a job instance that it's not running any longer

* Issue was seen when job instances are migrated and there are intermittent networking issues between the supe and worker causing job updates to NO come thru in an expected, orderly fashion.

ZD: 17868

==== CL 19126 ====

@FIX: on a network with intermittent worker-supe communication issues, bad timing can cause job instances to get stuck in "running" state

* In a bunch of routines that handle job-command executions (i.e., migrate, kill, etc.) in QbSupervisorCommand, add code to do one last check when a worker is unreachable, to see if the instance still belongs to the worker before updating the instance on DB. It was found that, since a thread dealing with down workers can spend quite a long time, sometimes instances that a worker was processing can be moved off of it and the DB updated by another thread (for example, assigned and running on another worker)-- the check is designed to prevent our thread from overwriting such updates.

ZD: 17868

==== CL 19121 ====

@FIX: job instances can get into an odd state when dispatch routine doesn't hear back from the worker ("found dead").

Networking hiccups can cause this communication drop, which in turn may cause job instances to be "stuck" in the running state on a worker, and be unkillable.

ZD: 17868

==== CL 19118 ====

@FIX: Systemctl unit files for worker and supervisor not installed into correct location

==== CL 19109 ====

@FIX: optimize job cleanup script

@CHANGE: only scan log directories if log removal necessary

@CHANGE: removal of large number of orphaned log directories does not require skipping sanity checks

==== CL 18985 ====

@FIX: 'No database selected' MySQL error when removing ghost jobs

ZD: 17882

==== CL 18911 ====

@TWEAK: add workerlog to show the host's available properties when inspecting a newly dispatched job (when "checking job requirements").

==== CL 18910 ====

@INTERNAL FIX: supervisor patches to help cut down on the number of threads, and reduce chances of repeated worker rejections on some farms due to race-conditions/timing issues.

ZD17713

==== CL 18831 ====

@NEW: add support for retrieval of only a specified range of jobs (IDs, date, N most recent, etc) in the qbjobinfo() API

Changed the "sign" field of QbFilter class to be a QbString rather than a char, to support SQL operators that are longer than a single character, such as ">=", "<>" or "!=".

Added a "limit" and "order_by" fields to the QbQuery class, so that any query can limit the number of jobs returned, and specify the sort order.

Made change to db-support code (QbDatabase.cpp) and supervisor code (QbSupervisorQuery.cpp and Queue) to take advantage of the above changes and implement the desired range-specific queries.

JIRA: QUBE-2658

==== CL 18822 ====

@FIX: a bug in the startHost() dispatch routine causing the supervisor NOT to always dispatch jobs to workers when they became available.

@INTERNAL: QbServer::printMemUsage() modified to only kick in if QB_DEBUG_SERVER_MEM_USAGE is defined

ZD: 17713

==== CL 18802 ====

@FIX: Correct 'restrictions' variable name and 'Restrictions' label

==== CL 18717 ====

@FIX: Job instances can become unkill-able with QB_PREEMPT_MODE_FAIL internal status

JIRA: QUBE-2819

==== CL 18680 ====

@FIX: supervisor rpm uninstall leaves the mysql/mariadb service in a stopped state instead of restarting it

Qube 6.10

#####

@RELEASE: 6.10-1

#####

==== CL 20025 ====

@FIX: extremely inaccurate cumulative cpu time for agenda items

JIRA: QUBE-3375
ZD: 18841

==== CL 20014 ====

@FIX: worker is added to the worker_dim dimension table as many times as there are expired entries for that same worker

==== CL 19931 ====

@FIX: Fix relative movie paths in images_to_move.py

==== CL 19897 ====

@FIX: auto_remove worker flag missing from worker config dialogs

==== CL 19834 ====

@FIX: supe/worker RPMs should be installable onto a system with core of the same major.minor mode installed

JIRA: QUBE-3332

==== CL 19478 ====

@FIX: workers are always "auto-remove"d, even if "auto_remove" is not set in worker_flags.

ZD: 18512

JIRA: QUBE-3174

==== CL 19475 ====

@FIX: issue where instances would be stuck in "QB_PREEMPT_MODE_FAIL", causing the supervisor to tell instances to "wait and retry later" in response to retryWork() indefinitely.

Issue was caused when the preemptJobNetwork() routine determines that the instance has started but has NOT yet started working on an agenda item, in which case it would mark the QB_PREEMPT_MODE_FAIL in order to interrupt (i.e. aggressively preempt) the instance; However, the interrupt was not being triggered properly.

Issue was apparently introduced in CL19126.

==== CL 19436 ====

@FIX: "down" workers not always detected properly

JIRA: QUBE-3155

ZD: 18425

==== CL 19425 ====

@FIX: issue when supe thread doesn't hear back from worker during a dispatch. Related to CL19243.

Also fixed an issue (probably harmless) where an extra call to queue.releaseJob() was sometimes made in the findSubjobAndReserveJob() method.

==== CL 19263 ====

@FIX: log directories for jobs submitted after the utility has been started but before the orphaned log removal is begun are erroneously removed

==== CL 19258 ====

@FIX: not running --use-frm when first-pass repair fails when message has different line-endings than OS X

==== CL 19243 ====

@FIX: add code to avoid mixed-up job instance status when worker-supervisor communications are dropped during job dispatch on an intermittently unreliable network

It was found that network hiccups can cause a worker to not respond to the supervisor during the dispatch of a job instance, but still start running the instance anyway. The worker would send the "running" instance report to the supervisor, which is processed by a separate thread, which updates the DB, causing a status mix-up.

Added code to detect such situations, and allowed the system to let the job run (instead of force-removing it from the duty table) on the worker in question.

Also added error-checking code on the worker side-- if worker detects that it couldn't respond to the supe for a dispatch order, it will give up on that job and release resources that it had just reserved for it.

ZD: 17868

==== CL 19236 ====

@FIX: jobs submitted by non-admin user without a specified priority attempt to submit at priority -1

JIRA: QUBE-3015

==== CL 19209 ====

@FIX: "down" workers would not be detected properly by the supervisor even when the supervisor_heartbeat_timeout expired.

ZD: 18057

JIRA: QUBE-3018

==== CL 19178 ====

@FIX: timing issue causing workers to get stuck with job instances.

Issue was seen on a very busy farm with intermittently drops in network communications, when many supe threads would try to dispatch a single instance at the same time.

ZD: 17868

==== CL 19163 ====

@FIX: fix an issue where a worker can sometimes get stuck with a job instance that it's not running any longer

* Issue was seen when job instances are migrated and there are intermittent networking issues between the supe and worker causing job updates to NO come thru in an expected, orderly fashion.

ZD: 17868

==== CL 19126 ====

@FIX: on a network with intermittent worker-supe communication issues, bad timing can cause job instances to get stuck in "running" state

* In a bunch of routines that handle job-command executions (i.e., migrate, kill, etc.) in QbSupervisorCommand, add code to do one last check when a worker is unreachable, to see if the instance still belongs to the worker before updating the instance on DB. It was found that, since a thread dealing with down workers can spend quite a long time, sometimes instances that a worker was processing can be moved off of it and the DB updated by another thread (for example, assigned and running on another worker)-- the check is designed to prevent our thread from overwriting such updates.

ZD: 17868

==== CL 19121 ====

@FIX: job instances can get into an odd state when dispatch routine doesn't hear back from the worker ("found dead").

Networking hiccups can cause this communication drop, which in turn may cause job instances to be "stuck" in the running state on a worker, and be unkillable.

ZD: 17868

==== CL 19118 ====

@FIX: Systemctl unit files for worker and supervisor not installed into correct location

==== CL 19109 ====

@FIX: optimize job cleanup script

@CHANGE: only scan log directories if log removal necessary

@CHANGE: removal of large number of orphaned log directories does not require skipping sanity checks

==== CL 18985 ====

@FIX: 'No database selected' MySQL error when removing ghost jobs

ZD: 17882

==== CL 18351 ====

@CHANGE: background helper thread improvements

* limit the number of workers that are potentially recontacted by the background helper routine to 50 per iteration.

* background thread exits and refreshes after running for approximately 1 hour, as opposed to 24 hours

ZD: 17124

#####

@RELEASE: 6.10-0a

#####

@SUMMARY: This is a supervisor-only patch release of 6.10-0 that includes the following key fixes.

NOTE regarding dependencies on Linux:

Installation of this updated supervisor package on a linux system requires the use of **rpm** with the **--nodeps** argument; the **yum** utility does not support disabling the dependency checks during installation, only removal.

==== CL 18910 ====

@INTERNAL FIX: supervisor patches to help cut down on the number of threads, and reduce chances of repeated worker rejections on some farms due to race-conditions/timing issues.

ZD17713

==== CL 18822 ====

@FIX: a bug in the startHost() dispatch routine causing the supervisor NOT to always dispatch jobs to workers when they became available.

ZD: 17713

==== CL 18717 ====

@FIX: Job instances can become unkill-able with QB_PREEMPT_MODE_FAIL internal status
JIRA: QUBE-2819

#####

@RELEASE: 6.10-0

#####

==== CL 18422 ====

@UPDATE: Shotgun API from v3.0.1 to v3.0.32
@CHANGE: images_to_movie.py - simplified options and syntax
@CHANGE: qube_imagesToMovie.py - simplified options and syntax
@CHANGE: simplecmd.py - Add "Upload Movie" option to Shotgun parameters
@CHANGE: shotgun_submitVersion.py - fixed movie upload functionality, general code cleanup

==== CL 18356 ====

@FIX: QBDIR set to null-string in job runtime environment

JIRA: QUBE-2611

==== CL 18351 ====

@CHANGE: background helper thread improvements

* limit the number of workers that are potentially recontacted by the background helper routine to 50 per iteration.

* background thread exits and refreshes after running for approximately 1 hour, as opposed to 24 hours

ZD: 17124

==== CL 18340 ====

@FIX: allow special characters in job name field at submissions

JIRA: QUBE-2748

==== CL 18324 ====

@CHANGE: output of "qbadmin s -config" and "qbadmin w -config hostname" now sorted alphabetically.

JIRA: QUBE-2654

==== CL 18285 ====

@FIX: add better error-checks in cmdrange jobtype's log-parsing code, in case the log file is not readable.

In some situations, fseek() was causing crashes in the parseFileStream() routine.

ZD: 17442

==== CL 18221 ====

@FIX: prevent "host.processors" to be unset when jobs are modified.

JIRA: QUBE-2649

==== CL 18185 ====

@CHANGE: make deferred table creation ON by default for all submissions via the APIs (C++: qbsubmit() , Python: qb.submit())

JIRA: QUBE-2603

==== CL 18157 ====

@FIX: shortened the timeout for "qbreportwork" when it reports a "failed" work that has migrate_on_frame_retry from 600 seconds to 20.

This was causing long 10-minute pauses on the job instance when a frame fails after exhausting all of its retry counts.

Original change was made in CL17206, for QUBE-2202/ZD16553.

ZD: 17447

==== CL 18147 ====

@FIX: Windows worker wouldn't properly release automounted drives at the end of running a job instance

ZD: 17400

==== CL 18107 ====

@FIX: memory leak in a DB-querying supervisor routine.

==== CL 18001 ====

@FIX: Pytnon API's qb.ping(asDict=True) was broken when metered licensing was unauthorized, because of the minus sign

==== CL 17984 ====

@CHANGE: add description of "disable_submit_check" flag to qb.conf.template comment

JIRA: QUBE-2560

==== CL 17982 ====

@CHANGE: Python API: license_provider_name and license_provider_key added to data returned by qb.hostinfo()

JIRA: QUBE-2549

==== CL 17944 ====

@CHANGE: Disable the two free worker licenses for any Qube installation.

JIRA: QUBE-2554

==== CL 17942 ====

@FIX: Some agenda item's "timestart" field doesn't reset when they are killed and then later retried.

JIRA: QUBE-2555

==== CL 17938 ====

@CHANGE: added verbosity in log entries about jobs that are "modified"

JIRA: QUBE-1473

==== CL 17898 ====

@NEW: add "no_defaults" job flag support to Python API files

JIRA: QUBE-2365

==== CL 17897 ====

@NEW: add no_defaults job flag, which tells the system to bypass the supervisor_job_flags.

If a job is submitted with no_defaults set in the job flag, the supervisor will NOT apply supervisor_job_flags.

JIRA: QUBE-2365

==== CL 17889 ====

@CHANGE: job queries requesting for subjob and/or work details now must explicitly provide job IDs.

Both qbjobinfo() C++ and qb.jobinfo() Python APIs now reject such submissions and return an error.

For example, the Python call "qb.jobinfo(subjobs=True)" will raise a runtime exception. It must be now called like "qb.jobinfo(subjobs=True, id=12345)" or "qb.jobinfo(subjobs=True, id=[1234,5678])"

JIRA: QUBE-244

==== CL 17863 ====

@FIX: Qube language callback command "mail-status" wasn't working properly, setting the smtp "TO" field to an incorrect string.

==== CL 17858 ====

@FIX: qb.deleteworkerproperties() and qb.deleteworkerresources() fn should return an error when used with the wrong 2nd arg (must be a list)

ZD: 16932
JIRA: QUBE-2381

==== CL 17856 ====
@FIX: misleading "invalid key" error message in supelog when supervisor_max_metered_licenses set to 0

JIRA: QUBE-2397

==== CL 17821 ====
@FIX: data warehouse worker table updates throttled to a single record at a time when multiple workers simultaneously change their defined slot counts

==== CL 17797 ====
@FIX: ignore any ethernet interface with "virutal" in its description when detecting the primary MAC address on Windows.

ZD 17072

==== CL 17790 ====
@FIX: issue where the background helper thread frequently sends 2 or more update requests (QB_MESSAGE_REQUEST_UPDATE) to a single "questionable" worker (i.e., one that has missed enough heartbeats, and potentially down) at once.

ZD: 17124

==== CL 16491 ====
@NOTES: Add support for AfterEffects point release scheme (2015.3)

==== CL 17763 ====
Supervisor and worker now use correct startup scripts for CentOS 7+, untested yet on CentOS 6.

==== CL 17744 ====
@CHANGE: Add a third paramter, "user", to Custom Policy's qb_approve_modify() routine, so the policy script can allow/disallow modification to a job based on the user name of the requestor.

For example, the routine can now allow certain users to only change priority between 7000 and 8000.

Note that ordinary users are still only allowed to modify his/her own jobs, while admins are allowed to modify anybody's jobs in any way, and are NOT subject to the "approve modify" custom policy routine.

With user groups defined (via "qbusers"), group admins are allowed to modify any job within its group. In that case, the "approve modify" routine does come into play.

JIRA: QUBE-2277

==== CL 17737 ====
@NEW: add 'pgrp' to job data stored in the data warehouse job_fact table.

==== CL 17735 ====
@FIX: badlogin jobs can't be retried or killed (previously fixed in CL15011, but regressed)

JIRA: QUBE-642
ZD: 12699, 17010

==== CL 17696 ====
@UPDATE: add explanation for "deferTableCreation" to the python qb.submit() API routine.

JIRA: QUBE-2400

==== CL 17692 ====
@FIX: another memory leak plugged in the startHost()-related routine, startQualifiedJobsOnHost(). This was causing successful iterations of startHost() (i.e., an instance was dispatched to a worker) to cause memory bloats. Among other places, it was affecting the background helper thread (when it does the "requeuing host" routine).

JIRA: QUBE-2382

==== CL 17649 ====
@FIX: memory leak in preemption code, especially when preemption policy is set to passive or is disabled by the algorithm.

QUBE: JIRA-2382

==== CL 17634 ====
@FIX: memory leak in one of the host-triggered dispatch routines startQualifiedJobsOnHost(), which is called from startHost().

Among other things, this was bloating the memory usage inside the helper routine running in a background thread/process (cleanermain()).

JIRA: QUBE-2382
ZD: 16952

==== CL 17610 ====

@FIX: memory corruption that would cause python or perl to crash when the function was called inside jobs.

JIRA: QUBE-2389

==== CL 17595 ====

@FIX: fixed memory leak in QbPack::store() and storeXML() methods, which were causing, among other things, supervisor threads to bloat when processing large job submissions

JIRA: QUBE-2382

==== CL 17594 ====

@FIX: plugged a potential memory leak in QbDaemon communication code, affecting all server (supervisor, worker) programs

JIRA: QUBE-2382

==== CL 17593 ====

@FIX: plugged memory leak in dispatch code

JIRA: QUBE-2382

==== CL 17592 ====

@FIX: plugged potential memory leak in user permission-check routine, specifically in the group-access check code

JIRA: QUBE-2382

==== CL 17566 ====

@NEW: qbwrk.conf loading optimization (and thus "qbadmin w -reconfig" speed up) by explicitly listing template names and non-existing hostnames in the new [global_config] section

* added [global_config] section to the qbwrk.conf file, and allow new config parameters "templates" to list all qbwrk.conf template section names, and "non_existent" to list all non-existent hostnames

* supe skips ip-address resolution for all section names included in "templates" and "non_existent", and all reserved names, i.e.: "global_config", "default", "linux", "osx", and "winnt", thus speeding up the loading of qbwrk.conf file, which in turn speeds up supervisor boot time and "qbadmin w -reconfig" operation.

JIRA: QUBE-2346

==== CL 17540 ====

@CHANGE: removed unnecessary submit-time check/rejection of omithosts and omitgroups.

ZD: 16907, 16908

JIRA: QUBE-2366

==== CL 17450 ====

@INTEG: rel-6.9 -> main

@FIX: directory deletion during log cleanup can fail if the supervisor is updating the job history file at the same time

==== CL 17449 ====

@FIX: directory deletion during log cleanup can fail if the supervisor is updating the job history file at the same time

==== CL 17435 ====

@FIX: supervisor process handling a qbping request should always reread the license file before replying

There was a code path that instructs the supe thread to force-read the license file, but the read was not happening under certain conditions; the code was returning the old cached data if available, or the default count of 2 if the cache isn't available.

* add a few more informational lines to print to the supelog at license re-reading.

JIRA: QUBE-2317

==== CL 17422 ====

@FIX: make formatting and object instantiation compatible with Python 2.6

==== CL 17416 ====

@FIX: remove unnecessary error message in the schema upgrade routine

JIRA: QUBE-2283

==== CL 17414 ====

@CHANGE: Add more text to describe the subtle yet significant difference between "retry" and "requeue" Python API routines

JIRA: QUBE-2049

==== CL 17403 ====

@FIX: jobs with status "registering" appears when submissions are rejected due to incorrect requirements specifications

ZD: 16408

JIRA: QUBE-2034

==== CL 17402 ====

@FIX: intermittent bug where some supe threads won't properly read the supervisor license key from qb.lic

* add warning message to print to supelog when the license file reader
returns zero-length data

ZD: 16828

JIRA: QUBE-2317

==== CL 17399 ====

@CHANGE: MSI no longer starting the worker service, qubeInstaller will start if required

==== CL 17390 ====

@FIX: post-flight should only be run when qbreportwork() is invoked with an agenda-item with terminal-state

JIRA: QUBE-2032

ZD: 16412

==== CL 17376 ====

@FIX: Triggers incorrectly executing multiple times

When a composite (i.e, using && or ||) trigger is specified for a job's callback, such as "done-job-job1 && done-job-job2",
the callback would erroneously get run multiple times.

ZD: 16282

JIRA: QUBE-1881

==== CL 17375 ====

LEGACY>>>>

@RELNOTES : NO

@INTERNAL: remove even more left-over files from initial metered license tracking

==== CL 17374 ====

LEGACY>>>>

@RELNOTES : NO

@INTERNAL: remove even more left-over files from initial metered license tracking

==== CL 17373 ====

LEGACY>>>>

@RELNOTES : NO

@INTERNAL: remove more left-over files from initial metered license tracking, where db was local to each machine

==== CL 17369 ====

@FIX: issue introduced in 6.9 where requestwork() jobtype backend routine will crash when frame padding is 40 or greater.

Python jobtype backend, in particular, was found to crash during a call to
the API routine qb.requestwork(), with a "**** stack smashing detected ***."
error message and a backtrace.

ZD: 16759

JIRA: QUBE-2318

==== CL 17290 ====

@TWEAK: license-reading routine prints the total license count to the supelog

JIRA: QUBE-2003

==== CL 17289 ====

@TWEAK: "ping" handler to print out more info to supelog

Every "qbping" will print out something like the following supelog now:

[Nov 18, 2016 16:25:55] shinyambp[11662]: INFO: responded to ping request from [127.0.0.1]: 6.9-0 bld-custom osx - - host - 0/11 unlimited
licenses (metered=0/0) - mode=0 (0)

JIRA: QUBE-2002

==== CL 17286 ====

@NEW: exposed Python's qb.admincommand() API routine, and add support for "reverify"

---- Sample Usage ----

```
cmd = {}
cmd['action'] = qb.CONST("QB_ADMIN_ORDER_ACTION_REVERIFY_WORKERS")
cmd['workers'] = ["shinyambp"] # optional
```

```
ret = qb.admincommand(cmd);
if(ret == None) :
print "ERROR: qb.admincommand() returned None";
else:
print "INFO: successfully sent admin order";
```

JIRA: QUBE-2159

==== CL 17285 ====

@NEW: add support for "reverify" in Perl's qb::admincommand() API routine

---- Sample Usage ----

```
my $command =
{
"action" => qb::CONST("QB_ADMIN_ORDER_ACTION_REVERIFY_WORKERS"),
"workers" => ["shinyambp"] # optional;
};
my $result = qb::admincommand($command);
if(not defined($result)) {
print STDERR "ERROR: qb::admincommand() returned undef\n";
} else {
print "INFO: successfully sent admin order\n";
}
```

JIRA: QUBE-2159

==== CL 17281 ====

@NEW: add 'qbadmin w -reverify [worker,...]' option to force the supervisor to reverify workers' license provider info.

JIRA: QUBE-2159

==== CL 17231 ====

@FIX: disabled verbose option for logging libcurl actions

==== CL 17208 ====

@CHANGE: Populate the subjob (instance) objects with more data (like status), and not just the IDs, when subjob info is requested via "qbhostinfo" (qb.hostinfo(subjobs=True) for python API)

Previously, only jobid, subid, and host info (name, address, macaddress) were filled. Now, things like "status", "timestart", "allocations", etc. are properly filled in.

JIRA: QUBE-2073

ZD: 16541

==== CL 17206 ====

@FIX: When "migrate_on_frame_retry" job flag is set, prevent backend from doing further processing (especially another requestwork()) after a work failed

This was causing race-conditions that will get agenda items to be stuck in "retrying" state, while there are no instances processing them.

Now the reportwork() API routine is modified so that if it's invoked to report that a work "failed", and the "migrate_on_frame_retry" is set on the job, it will stop processing (does a long sleep), and let the worker/proxy do the process clean up.

JIRA: QUBE-2202

ZD: 16553

==== CL 17199 ====

@NEW: add "auto_remove" worker_flag, which indicates to the supervisor that this worker should be automatically removed when it goes "down"

JIRA: QUBE-1058

==== CL 17198 ====

@NEW: add Partner Licensing support to supervisor

JIRA: QUBE-1911, QUBE-1912, QUBE-1913, QUBE-1914, QUBE-1915

==== CL 17186 ====

@FIX: "VirtualBox Host-Only Ethernet Adapter" now when daemons (supe, worker) try to pick a primary mac address

JIRA: QUBE-2149

ZD: 16561

==== CL 17182 ====

@CHANGE: all classes that inherit from QbObject print as a regular dictionary, no longer have a __repr__ which prints the job data as a single flat string

@NEW: add qb.validatejob() function to python API, help find malformed jobs that crash the user interfaces

==== CL 17141 ====

@FIX: Any job submitted from within a running job picks up the pgrp of the submitting job

By design, if the submission environment has QBGRPID and QBJOBID set, the API's submission routine will set the job's pgrp and pid, respectively to the values specified in the environment variables.

One couldn't override this "inheritance" behavior even by explicitly specifying "pgrp" or "pid" in the job being submitted, for instance with the "-pgrp" command-line option of qbsub.

Fixed, so that setting "pgrp" to 0 on submission means that the job should generate its own pgrp instead of inheriting it from the environment.

JIRA: QUBE-2141

ZD: 16545

==== CL 17101 ====

@NEW: add "-dying" and "-registering" options to qbjobs.

@CHANGE: also add dying and registering jobs to the "-active" filter.

JIRA: QUBE-2091

ZD: 16469

==== CL 17083 ====

@FIX: Python API: qbbing(asDict=True) crashes when used against older (pre-6.9) supe

Among other things, this was causing WV to crash and AV to note an exception (but not crash) when starting up with an older supervisro.

JIRA: QUBE-2084

Qube 6.9

#####

@RELEASE: 6.9-2b

#####

@SUMMARY: This is a supervisor-only patch release of 6.9-2 that includes the following key fixes.

==== CL 18910 ====

@INTERNAL FIX: supervisor patches to help cut down on the number of threads, and reduce chances of repeated worker rejections on some farms due to race-conditions/timing issues.

ZD17713

==== CL 18822 ====

@FIX: a bug in the startHost() dispatch routine causing the supervisor NOT to always dispatch jobs to workers when they became available.

ZD: 17713

#####

@RELEASE: 6.9-2a

#####

@SUMMARY: 6.9-2a is a patch release of 6.9-2, and includes the following fixes.

==== CL 18717 ====

@FIX: Job instances can become unkill-able with QB_PREEMPT_MODE_FAIL internal status

JIRA: QUBE-2819

==== CL 18351 ====

@CHANGE: background helper thread improvements

* limit the number of workers that are potentially recontacted by the background helper routine to 50 per iteration.

* background thread exits and refreshes after running for approximately 1 hour, as opposed to 24 hours

ZD: 17124

==== CL 18340 ====

@FIX: allow special characters in job name field at submissions

JIRA: QUBE-2748

==== CL 18324 ====

@CHANGE: output of "qbadm s -config" and "qbadm w -config hostname" now sorted alphabetically.

JIRA: QUBE-2654

==== CL 18285 ====

@FIX: add better error-checks in cmdrange jobtype's log-parsing code, in case the log file is not readable.

In some situations, fseek() was causing crashes in the parseFileStream() routine.

ZD: 17442

==== CL 18221 ====

@FIX: prevent "host.processors" to be unset when jobs are modified.

JIRA: QUBE-2649

==== CL 18157 ====

@FIX: shortened the timeout for "qbreportwork" when it reports a "failed" work that has migrate_on_frame_retry from 600 seconds to 20.

This was causing long 10-minute pauses on the job instance when a frame fails after exhausting all of its retry counts.

Original change was made in CL17206, for QUBE-2202/ZD16553.

ZD: 17447

==== CL 18147 ====

@FIX: Windows worker wouldn't properly release automounted drives at the end of running a job instance

ZD: 17400

==== CL 18001 ====

@FIX: Pytnon API's qb.ping(asDict=True) was broken when metered licensing was unauthorized, because of the minus sign

==== CL 17889 ====

@CHANGE: job queries requesting for subjob and/or work details now must explicitly provide job IDs.

Both qbjobinfo() C++ and qb.jobinfo() Python APIs now reject such submissions and return an error.

For example, the Python call "qb.jobinfo(subjobs=True)" will raise a runtime exception. It must be now called like "qb.jobinfo(subjobs=True, id=12345)" or "qb.jobinfo(subjobs=True, id=[1234,5678])"

JIRA: QUBE-244

==== CL 17863 ====

@FIX: Qube language callback command "mail-status" wasn't working properly, setting the smtp "TO" field to an incorrect string.

==== CL 17858 ====

@FIX: qb.deleteworkerproperties() and qb.deleteworkerresources() fn should return an error when used with the wrong 2nd arg (must be a list)

ZD: 16932

JIRA: QUBE-2381

==== CL 17856 ====

@FIX: misleading "invalid key" error message in supelog when supervisor_max_metered_licenses set to 0

JIRA: QUBE-2397

==== CL 17797 ====

@FIX: ignore any ethernet interface with "virutal" in its description when detecting the primary MAC address on Windows.

ZD 17072

==== CL 17790 ====

@FIX: issue where the background helper thread frequently sends 2 or more update requests (QB_MESSAGE_REQUEST_UPDATE) to a single "questionable" worker (i.e., one that has missed enough heartbeats, and potentially down) at once.

ZD: 17124

==== CL 17735 ====

@FIX: badlogin jobs can't be retried or killed (previously fixed in CL15011, but regressed)

JIRA: QUBE-642

ZD: 12699, 17010

==== CL 16491 ====

@NOTES: Add support for AfterEffects point release scheme (2015.3)

#####

@RELEASE: 6.9-2

#####

@SUMMARY: This is a maintenance release of 6.9, and includes a few fixes and improvements to 6.9-1. Recommended upgrade for all 6.9 customers.

#####

==== CL 17763 ====

Supervisor and worker now use correct startup scripts for CentOS 7+.

==== CL 17735 ====

@FIX: badlogin jobs can't be retried or killed (previously fixed in CL15011, but regressed)

JIRA: QUBE-642

ZD: 12699, 17010

#####

@RELEASE: 6.9-1

#####

@SUMMARY: This is a maintenance release of 6.9, and includes a number of fixes and improvements to 6.9-0. Recommended upgrade for all 6.9 customers.

#####

==== CL 17696 ====

@UPDATE: add explanation for "deferTableCreation" to the python qb.submit() API routine.

JIRA: QUBE-2400

==== CL 17692 ====

@FIX: another memory leak plugged in the startHost()-related routine, startQualifiedJobsOnHost(). This was causing successful iterations of startHost() (i.e., an instance was dispatched to a worker) to cause memory bloats. Among other places, it was affecting the background helper thread (when it does the "requeuing host" routine).

JIRA: QUBE-2382

==== CL 17649 ====

@FIX: memory leak in preemption code, especially when preemption policy is set to passive or is disabled by the algorithm.

QUBE: JIRA-2382

==== CL 17634 ====

@FIX: memory leak in one of the host-triggered dispatch routines
startQualifiedJobsOnHost(), which is called from startHost().

Among other things, this was bloating the memory usage inside the helper
routine running in a background thread/process (cleanermain()).

JIRA: QUBE-2382

ZD: 16952

==== CL 17610 ====

@FIX: memory corruption that would cause python or perl to crash when the function was called inside jobs.

JIRA: QUBE-2389

==== CL 17595 ====

@FIX: fixed memory leak in QbPack::store() and storeXML() methods, which were causing, among other things, supervisor threads to bloat when
processing large job submissions

JIRA: QUBE-2382

==== CL 17594 ====

@FIX: plugged a potential memory leak in QbDaemon communication code, affecting all server (supervisor, worker) programs

JIRA: QUBE-2382

==== CL 17593 ====

@FIX: plugged memory leak in dispatch code

JIRA: QUBE-2382

==== CL 17592 ====

@FIX: plugged potential memory leak in user permission-check routine, specifically in the group-access check code

JIRA: QUBE-2382

==== CL 17566 ====

@NEW: qbwrk.conf loading optimization (and thus "qbadmin w -reconfig" speed up) by explicitly listing template names and non-existing
hostnames in the new [global_config] section

* added [global_config] section to the qbwrk.conf file, and allow new config parameters "templates" to list all qbwrk.conf template section names,
and "non_existent" to list all non-existent hostnames

* supe skips ip-address resolution for all section names included in "templates" and "non_existent", and all reserved names, i.e.: "global_config",
"default", "linux", "osx", and "winnt", thus speeding up the loading of qbwrk.conf file, which in turn speeds up supervisor boot time and "qbadmin w
-reconfig" operation.

JIRA: QUBE-2346

==== CL 17540 ====

@CHANGE: removed unnecessary submit-time check/rejection of omithosts and omitgroups.

ZD: 16907, 16908

JIRA: QUBE-2366

==== CL 17449 ====

@FIX: directory deletion during log cleanup can fail if the supervisor is updating the job history file at the same time

==== CL 17435 ====

@FIX: supervisor process handling a qbping request should always reread the license file before replying

There was a code path that instructs the supe thread to force-read the
license file, but the read was not happening under certain conditions; the
code was returning the old cached data if available, or the default count
of 2 if the cache isn't available.

* add a few more informational lines to print to the supe log at license
re-reading.

JIRA: QUBE-2317

==== CL 17422 ====

@FIX: make formatting and object instantiation compatible with Python 2.6

==== CL 17416 ====

@FIX: remove unnecessary error message in the schema upgrade routine

JIRA: QUBE-2283

==== CL 17414 ====

@CHANGE: Add more text to describe the subtle yet significant difference between "retry" and "requeue" Python API routines

JIRA: QUBE-2049

==== CL 17403 ====

@FIX: jobs with status "registering" appears when submissions are rejected due to incorrect requirements specifications

ZD: 16408

JIRA: QUBE-2034

==== CL 17402 ====

@FIX: intermittent bug where some supe threads won't properly read the supervisor license key from qb.lic

* add warning message to print to supelog when the license file reader
returns zero-length data

ZD: 16828

JIRA: QUBE-2317

==== CL 17390 ====

@FIX: post-flight should only be run when qbreportwork() is invoked with an agenda-item with terminal-state

JIRA: QUBE-2032

ZD: 16412

==== CL 17376 ====

@FIX: Triggers incorrectly executing multiple times

When a composite (i.e, using && or ||) trigger is specified for a job's callback, such as "done-job-job1 && done-job-job2",
the callback would erroneously get run multiple times.

ZD: 16282

JIRA: QUBE-1881

==== CL 17369 ====

@FIX: issue introduced in 6.9 where requestwork() jobtype backend routine will crash when frame padding is 40 or greater.

Python jobtype backend, in particular, was found to crash during a call to
the API routine qb.requestwork(), with a "**** stack smashing detected ***."
error message and a backtrace.

ZD: 16759

JIRA: QUBE-2318

==== CL 17290 ====

@TWEAK: license-reading routine prints the total license count to the supelog

JIRA: QUBE-2003

==== CL 17289 ====

@TWEAK: "ping" handler to print out more info to supelog

Every "qbping" will print out something like the following supelog now:

[Nov 18, 2016 16:25:55] shinyambp[11662]: INFO: responded to ping request from [127.0.0.1]: 6.9-0 bld-custom osx - - host - 0/11 unlimited
licenses (metered=0/0) - mode=0 (0)

JIRA: QUBE-2002

==== CL 17231 ====

@FIX: disabled verbose option for logging libcurl actions

==== CL 17208 ====

@CHANGE: Populate the subjob (instance) objects with more data (like status), and not just the IDs, when subjob info is requested via
"qbhostinfo" (qb.hostinfo(subjobs=True) for python API)

Previously, only jobid, subid, and host info (name, address, macaddress)
were filled. Now, things like "status", "timestart", "allocations",

etc. are properly filled in.

JIRA: QUBE-2073
ZD: 16541

==== CL 17206 ====

@FIX: When "migrate_on_frame_retry" job flag is set, prevent backend from doing further processing (especially another requestwork()) after a work failed

This was causing race-conditions that will get agenda items to be stuck in "retrying" state, while there are no instances processing them.

Now the reportwork() API routine is modified so that if it's invoked to report that a work "failed", and the "migrate_on_frame_retry" is set on the job, it will stop processing (does a long sleep), and let the worker/proxy do the process clean up.

JIRA: QUBE-2202
ZD: 16553

==== CL 17186 ====

@FIX: "VirtualBox Host-Only Ethernet Adapter" now when daemons (supe, worker) try to pick a primary mac address

JIRA: QUBE-2149
ZD: 16561

==== CL 17182 ====

@CHANGE: all classes that inherit from QbObject print as a regular dictionary, no longer have a __repr__ which prints the job data as a single flat string

@NEW: add qb.validatejob() function to python API, help find malformed jobs that crash the user interfaces

==== CL 17141 ====

@FIX: Any job submitted from within a running job picks up the pgrp of the submitting job

By design, if the submission environment has QBGRPID and QBJOBID set, the API's submission routine will set the job's pgrp and pid, respectively to the values specified in the environment variables.

One couldn't override this "inheritance" behavior even by explicitly specifying "pgrp" or "pid" in the job being submitted, for instance with the "-pgrp" command-line option of qbsub.

Fixed, so that setting "pgrp" to 0 on submission means that the job should generate its own pgrp instead of inheriting it from the environment.

JIRA: QUBE-2141
ZD: 16545

==== CL 17101 ====

@NEW: add "-dying" and "-registering" options to qbjobs.

@CHANGE: also add dying and registering jobs to the "-active" filter.

JIRA: QUBE-2091
ZD: 16469

==== CL 17083 ====

@FIX: Python API: qbping(asDict=True) crashes when used against older (pre-6.9) supe

Among other things, this was causing WV to crash and AV to note an exception (but not crash) when starting up with an older supervisro.

JIRA: QUBE-2084

#####

@RELEASE: 6.9-0

#####

==== CL 16804 ====

@TWEAK: added code to print what operation was requested, when printing out "permission granted to user..."

==== CL 16776 ====

@FIX: Python API should handle exception for when gethostbyname() doesn't work in mysqlConnect

JIRA: QUBE-1965

==== CL 16770 ====

@CHANGE: Ensure that the pending reasons returned by qb.hostorder (or qbhostorder command) take metered licensing into account

JIRA: QUBE-1986

==== CL 16696 ====

@NEW: add supervisor_max_metered_licenses support to qb.conf, which enables site-admins to customize the effective limit of metered licenses that can be used at any given time.

This number must be smaller than the metered account's limit, or it will be capped at the account limit.

Setting this to 0 effectively disables metered licensing, while setting it to -1 (default), allows usage up to the metered account's limit .

JIRA: QUBE-1867

==== CL 16668 ====

@NEW: made available some frame-padding related environment variables during the execution of job instances and pre/postflights:

QB_FRAME_PADDING
QB_PADDED_FRAME_NUMBER
QB_PADDED_FRAME_START
QB_PADDED_FRAME_END
QB_PADDED_FRAME_STEP

JIRA: QUBE-1841

==== CL 16665 ====

@CHANGE: All "subjob" sections in qbsummary output show "instance" in the title

@CHANGE: renamed "*vs" options to "*vi" (such as "pvi" or "cvi"). For compatibility, the older names still work, just not advertised in the "help" output

@FIX: const-ness of QbString::replacevalue() method

JIRA: QUBE-1617

==== CL 16643 ====

@FIX: added dependency on mysql-libs (or mariadb-libs) to the supervisor RPM

JIRA: QUBE-1784

==== CL 16642 ====

@CHANGE: automatic capping of priorities to supervisor_highest_user_priority

if an ordinary (non-admin) user tries to submit jobs at a higher priority (i.e. lower numerical value) than supervisor_highest_user_priority, the jobs will be accepted but with the priority automatically (and silently, except for a WARNING message in the supelog) capped at supervisor_highest_user_priority

JIRA: QUBE-1804

==== CL 16629 ====

@CHANGE: "kill work" on a running agenda item will now put the instance processing the agenda item back to "pending", instead of also killing it.

JIRA: QUBE-627

==== CL 16628 ====

@FIX: "qb_default_string()" warning printed during linux qube-core installation

Corrected code so that warnings like the following won't print any more:

WARNING: qb_default_string() unknown value[1001]
WARNING: qb_default_string() unknown value[1002]

JIRA: QUBE-1894

==== CL 16602 ====

@FIX: misleading database name printed in error handler for MySQL stored procedures PFX_CALC_CPU_TIME() and PFX_CALC_AVG_WORK_TIME(); "ERROR: TABLE NOT FOUND IN DB pfx_dw.<actual_database_name>"

==== CL 16517 ====

@FIX: C4D appFinder jobs don't apply path translation properly on Windows, backslashes are converted too early

==== CL 16407 ====

@NEW: add SMTP Auth support over SSL and TLS connections.

@CHANGE:

* add new mail config qb.conf parameters: mail_user, mail_password, mail_connection_type

* modified mail_port to be 0 by default, which means use the standard port depending on connection type: 25, 465 (SSL), or 587 (TLS)

==== CL 16389 ====

@FIX: calls to qb.reportwork that happen very close together can cause the supervisor to deadlock on a single frame's status

==== CL 16379 ====

@FIX: case-insensitive parsing of template names in qbwrc.conf when listed for template inheritance

The following now works (hostA will be in the "big" group):

```
[BigNode]
```

```
worker_groups = "big"
```

```
[hostA] : bignode
```

JIRA: QUBE-1809

==== CL 16369 ====

@FIX: don't mark the instance as failed if there is one more command to run, the child process has already exited, and the command is sys.exit(0); happens when maya is shut down with its native quit() function.

==== CL 16338 ====

@CHANGE: database checks script splits logging levels between stdout and stderr

==== CL 16308 ====

@CHANGE: fixed every reference to "subjob" to "instance"

JIRA: QUBE-1768

==== CL 16303 ====

@CHANGE: add supervisor mode settings (such as "disable_metered") to display in qbping output, and be returned in the qb.ping(asDict=True) Python API invocation

JIRA: QUBE-1759

==== CL 16286 ====

@FIX: checkDiskUsage fails when --mysql option is used and root can't authenticate

==== CL 16269 ====

@FIX: properly support timeouts on socket connections

@NEW: add "-timeout N" option to the qbping command, and the API qbping(), qbworkerping(), and qbhostping() API routines now honor the timeout set via "qbsettimeout()".

QUBE-1746

==== CL 16266 ====

@NEW: a new command-line utility for performing both database health checks and data integrity checks

==== CL 16247 ====

@FIX: fixed qb.workid() in callbacks to return the correct workid of the current callback context (it had been always returning None)

Also changed qb.jobstatus(), workstatus(), and subjobstatus() so that, if invoked in a callback giving no args (like a jobid and workid or subjobid), they return the status of the respective thing (job, work, or subjob) of the current callback context.

JIRA: QUBE-1763

ZD: 16105

==== CL 16235 ====

@FIX: a problem with the filtering added to avoid jobs with an ID of 0, in CL15821

This was causing preemption to not function in many cases.

ZD: 16006

==== CL 16229 ====

@FIX: On Windows, daemons (supe, worker) now ignore VMWare Virtual Ethernet Adapters when trying to pick a primary mac address (QbConnection.cpp) for the host, which is used to uniquely identify hosts

ZD: 14481

==== CL 16214 ====

@FIX: aerender AppFinder mangling first path conversion on Windows when using UNC

==== CL 16177 ====

@NEW: add metered_max and metered_used fields to the dict returned by qb.ping(asDict=True)

JIRA: QUBE-1745

==== CL 16145 ====

@NEW: add support for Metered Licensing

==== CL 16139 ====

@FIX: Fixed the duplicate instance of "stop_activity" (i.e., it was listed twice), to "enforce_password" in qb_supervisor_mode_flag_string(), which was causing string to int conversion of the mode flags to be incorrect

==== CL 16064 ====

@FIX: when job 'dev' attribute True, printing the job package with regex_errors causes the logParser to generate a false positive for the regex_errors match

==== CL 16049 ====

@NEW: add 'outputPath match required' to python-based jobs, frame/work is failed if no match is found

==== CL 15974 ====

@CHANGE: add support for "-conf PATH" to specify qb.conf for worker (phase 1)

QUBE-253

==== CL 15970 ====

@FIX: modified (un)install_supervisor scripts to properly support CentOS/RHEL 7+ with mariadb and systemd.

Also modified configure_mysql script (for Linux) to be able to detect the version of mysql installed on the system, even when the server is not running

QUBE-1663

==== CL 15964 ====

@NEW: changes to code that generates/modifies my.cnf

@CHANGE: some refactoring of the configure_mysql script (run on linux on (un)installation of the supervisor to modify my.cnf.

@NEW: make sure "default-storage-engine=MyISAM" is set on Linux too

@NEW: add "query_cache_type=0" to my.cnf on all platforms

JIRA: QUBE-1663

==== CL 15960 ====

@FIX: jobs submitted with pgrp set to a (null) string end up having a pgrp of 0

JIRA: QUBE-1668

==== CL 15957 ====

@FIX: use of single-quotes in job dependency "info-*" syntax results in hung job instances

JIRA: QUBE-1571

==== CL 15947 ====

@CHANGE: adding "default-storage-engine=MYISAM" to the my.cnf generated for Linux/OSX supe installations

JIRA: QUBE-1663

==== CL 15936 ====

@CHANGE: add InnoDB to MyISAM conversion code in upgrade_supervisor program for all "qube" tables

JIRA: QUBE-1664

==== CL 15909 ====

@CHANGE: change flaw in auto-wrangling logic in which it sometimes won't detect a bad worker, and allows it to fail many job agendas.

When a single job instance/worker has failed all of its assigned frames (at least aw_activation_work_count frames) for a job, while other workers are still processing their first frame (i.e., no other worker/instance has finished a frame), the system deems this worker "bad", locks it, and migrates the failed frames and instance, and notify the admin.

JIRA: QUBE-1475
ZD: 15219

==== CL 15865 ====
@CHANGE: Made section headers (such as "[default]" or "[node[001-199]]") case-insensitive in config files such as qbwk.conf

JIRA: QUBE-1356

==== CL 15821 ====
@FIX: add code to the DB routines and doPreemption() routine to silently ignore job records with job ID of 0 (likely due to corrupt DB records), which was spewing out many warning messages into the supelog

ZD:15739

==== CL 15809 ====
@FIX: backslashed characters in VRED jobs get treated as escape characters

==== CL 15700 ====
@NEW: add "--conf filename" option to supervisor to specify an alternate location and name for the qb.conf file

JIRA: QUBE-253

==== CL 15673 ====
@FIX: orphaned job processes left behind on Windows workers, especially when the proxy.exe program dies unexpectedly

ZD: 15518

==== CL 15653 ====
@FIX: setting jobss "pgrp" value prior to submission is ignored for all but the first job when submitting a list of jobs via a single call to the qbsubmit() API routine

JIRA: QUBE-1536
ZD: 15528

==== CL 15650 ====
@FIX: Explicitly setting "host.memory" in worker_resources broken on Linux

ZD: 15505

JIRA: QUBE-1531

==== CL 15642 ====
@FIX: Unix (Linux/OSX) workers, when running a cleanup process for a terminating job instance (via removeJob()), would sometimes inadvertently kill processes belonging to other job instances, due to process IDs once owned by the terminating job being reused by the system.

ZD: 15548

==== CL 15567 ====
@FIX: supervisor_default_max_cpus value was not being applied properly

ZD: 15503

JIRA: QUBE-1528

==== CL 15560 ====
@CHANGE: "modify" operation will print, into the supelog and the job's .hst file, the values of the newly modified parameters

JIRA: QUBE-1318
ZD: 14979

==== CL 15531 ====
@NEW: add [run_program_and_convert_encoding.pl](#) script, which is a wrapper to run any given program and convert its stdout from and to specified encodings (like UTF-16le to UTF-8).

Added to support 3dsmax batch (i.e., "cmdrange") submissions.

JIRA: QUBE-1210

==== CL 15462 ====
@FIX: removed submission-time check for jobtype existence on the farm, as it was causing false negatives in certain cases and disallowing submissions

ZD: 15328, 15831

==== CL 15423 ====
@FIX: KeyError: "regex_outputPaths" is raised when min file size check is specifec, but no outputPath regular expression is defined

==== CL 15384 ====
@NEW: add Mac OS X 10.11, aka "El Capitan" support

==== CL 15380 ====

@CHANGE: modification now allowed on "done" jobs

ZD: 15281

==== CL 15351 ====

@FIX: Windows issue where wireless network interfaces are ignored when licenses are verified, causing license keys bound to such interfaces to not work.

==== CL 15347 ====

@FIX: Windows issue where wireless network interfaces are ignored when licenses are verified, causing license keys bound to such interfaces to not work.

==== CL 15324 ====

@CHANGE: supervisor on Win32 to build against Perl 5.8 (upgraded from 5.6) to avoid build issues on new build platform.

Qube 6.8

#####

@RELEASE: 6.8-4a

#####

This is a cumulative patch release of the qube-core, supervisor, and worker packages, for all platforms, including several key fixes.

==== CL 17208 ====

@CHANGE: Populate the subjob (instance) objects with more data (like status), and not just the IDs, when subjob info is requested via "qbhostinfo" (qb.hostinfo(subjobs=True) for python API)

Previously, only jobid, subid, and host info (name, address, macaddress) were filled. Now, things like "status", "timestart", "allocations", etc. are properly filled in.

JIRA: QUBE-2073

ZD: 16541

==== CL 17206 ====

@FIX: When "migrate_on_frame_retry" job flag is set, prevent backend from doing further processing (especially another requestwork()) after a work failed

This was causing race-conditions that will get agenda items to be stuck in "retrying" state, while there are no instances processing them.

Now the reportwork() API routine is modified so that if it's invoked to report that a work "failed", and the "migrate_on_frame_retry" is set on the job, it will stop processing (does a long sleep), and let the worker/proxy do the process clean up.

JIRA: QUBE-2202

ZD: 16553

==== CL 17186 ====

@FIX: "VirtualBox Host-Only Ethernet Adapter" now when daemons (supe, worker) try to pick a primary mac address

JIRA: QUBE-2149

ZD: 16561

==== CL 17182 ====

@CHANGE: all classes that inherit from QbObject print as a regular dictionary, no longer have a __repr__ which prints the job data as a single flat string

@NEW: add qb.validatejob() function to python API, help find malformed jobs that crash the user interfaces

==== CL 17141 ====

@FIX: Any job submitted from within a running job picks up the pgrp of the submitting job

By design, if the submission environment has QBGRPID and QBJOBID set, the API's submission routine will set the job's pgrp and pid, respectively to the values specified in the environment variables.

One couldn't override this "inheritance" behavior even by explicitly

specifying "pgrp" or "pid" in the job being submitted, for instance with the "-pgrp" command-line option of qbsub.

Fixed, so that setting "pgrp" to 0 on submission means that the job should generate its own pgrp instead of inheriting it from the environment.

JIRA: QUBE-2141
ZD: 16545

==== CL 17101 ====
@NEW: add "-dying" and "-registering" options to qbjobs.
@CHANGE: also add dying and registering jobs to the "-active" filter.

JIRA: QUBE-2091
ZD: 16469

==== CL 16804 ====
@TWEAK: added code to print what operation was requested, when printing out "permission granted to user..."

#####

@RELEASE: 6.8-4

#####

This is a cumulative patch release for all platforms.

==== CL 16628 ====
@FIX: "qb_default_string()" warning printed during linux qube-core installation

Corrected code so that warnings like the following won't print any more:

WARNING: qb_default_string() unknown value[1001]
WARNING: qb_default_string() unknown value[1002]

JIRA: QUBE-1894

==== CL 16602 ====
@FIX: misleading database name printed in error handler for MySQL stored procedures PFX_CALC_CPU_TIME() and PFX_CALC_AVG_WORK_TIME(); "ERROR: TABLE NOT FOUND IN DB pfx_dw.<actual_database_name>"

==== CL 16517 ====
@FIX: C4D appFinder jobs don't apply path translation properly on Windows, backslashes are converted too early

==== CL 16491 ====
@NOTES: Add support for AfterEffects point release scheme (2015.3)

#####

@RELEASE: 6.8-3c

#####

This is a patch release of core/supe/worker only, with some critical fixes to 6.8-3 for all platforms.

#####

==== CL 16389 ====
@FIX: calls to qb.reportwork that happen very close together can cause the supervisor to deadlock on a single frame's status

==== CL 16379 ====
@FIX: case-insensitive parsing of template names in qbwrk.conf when listed for template inheritance

The following now works (hostA will be in the "big" group):

```
[BigNode]
worker_groups = "big"
```

```
[hostA] : bignode
```

JIRA: QUBE-1809

==== CL 16369 ====

@FIX: don't mark the instance as failed if there is one more command to run, the child process has already exited, and the command is sys.exit(0); happens when maya is shut down with its native quit() function.

==== CL 16338 ====

@CHANGE: database checks script splits logging levels between stdout and stderr

==== CL 16286 ====

@FIX: checkDiskUsage fails when --mysql option is used and root can't authenticate

==== CL 16266 ====

@NEW: a new command-line utility for performing both database health checks and data integrity checks

==== CL 16247 ====

@FIX: fixed qb.workid() in callbacks to return the correct workid of the current callback context (it had been always returning None)

Also changed qb.jobstatus(), workstatus(), and subjobstatus() so that, if invoked in a callback giving no args (like a jobid and workid or subjobid), they return the status of the respective thing (job, work, or subjob) of the current callback context.

JIRA: QUBE-1763

ZD: 16105

==== CL 16235 ====

@FIX: a problem with the filtering added to avoid jobs with an ID of 0, in CL15821

This was causing preemption to not function in many cases.

ZD: 16006

==== CL 16229 ====

@FIX: On Windows, daemons (supe, worker) now ignore VMWare Virtual Ethernet Adapters when trying to pick a primary mac address (QbConnection.cpp) for the host, which is used to uniquely identify hosts

ZD: 14481

==== CL 16214 ====

@FIX: aerender AppFinder mangling first path conversion on Windows when using UNC

==== CL 16064 ====

@FIX: when job 'dev' attribute True, printing the job package with regex_errors causes the logParser to generate a false positive for the regex_errors match

==== CL 16049 ====

@NEW: add 'outputPath match required' to python-based jobs, frame/work is failed if no match is found

#####

@RELEASE: 6.8-3

#####

==== CL 15964 ====

@NEW: changes to code that generates/modifies my.cnf

@CHANGE: some refactoring of the configure_mysql script (run on linux on (un)installation of the supervisor to modify my.cnf.

@NEW: make sure "default-storage-engine=MyISAM" is set on Linux too

@NEW: add "query_cache_type=0" to my.cnf on all platforms

JIRA: QUBE-1663

==== CL 15960 ====

@FIX: jobs submitted with pgrp set to a (null) string end up having a pgrp of 0

JIRA: QUBE-1668

==== CL 15957 ====

@FIX: use of single-quotes in job dependency "info-*" syntax results in hung job instances

JIRA: QUBE-1571

==== CL 15947 ====

@CHANGE: adding "default-storage-engine=MYISAM" to the my.cnf generated for Linux/OSX supe installations

JIRA: QUBE-1663

==== CL 15936 ====

@CHANGE: add InnoDB to MyISAM conversion code in upgrade_supervisor program for all "qube" tables

JIRA: QUBE-1664

==== CL 15909 ====

@CHANGE: change flaw in auto-wrangling logic in which it sometimes won't detect a bad worker, and allows it to fail many job agendas.

When a single job instance/worker has failed all of its assigned frames (at least `aw_activation_work_count` frames) for a job, while other workers are still processing their first frame (i.e., no other worker/instance has finished a frame), the system deems this worker "bad", locks it, and migrates the failed frames and instance, and notify the admin.

JIRA: QUBE-1475

ZD: 15219

==== CL 15865 ====

@CHANGE: Made section headers (such as "[default]" or "[node[001-199]]") case-insensitive in config files such as `qbwrk.conf`

JIRA: QUBE-1356

==== CL 15848 ====

@NEW: add Ubuntu 16.04 LTS support

==== CL 15821 ====

@FIX: add code to the DB routines and `doPreemption()` routine to silently ignore job records with job ID of 0 (likely due to corrupt DB records), which was spewing out many warning messages into the `supelog`

ZD:15739

==== CL 15809 ====

@FIX: backslashed characters in VRED jobs get treated as escape characters

==== CL 15761 ====

@NEW: add CentOS 7.2 support

JIRA: QUBE-1482

==== CL 15700 ====

@NEW: add "--conf filename" option to supervisor to specify an alternate location and name for the `qb.conf` file

JIRA: QUBE-253

#####

@RELEASE: 6.8-2

#####

==== CL 15673 ====

@FIX: orphaned job processes left behind on Windows workers, especially when the `proxy.exe` program dies unexpectedly

ZD: 15518

==== CL 15653 ====

@FIX: setting `jobss "pgrp"` value prior to submission is ignored for all but the first job when submitting a list of jobs via a single call to the `qbsubmit()` API routine

JIRA: QUBE-1536

ZD: 15528

==== CL 15650 ====

@FIX: Explicitly setting "host.memory" in `worker_resources` broken on Linux

ZD: 15505

JIRA: QUBE-1531

==== CL 15642 ====

@FIX: Unix (Linux/OSX) workers, when running a cleanup process for a terminating job instance (via `removeJob()`), would sometimes inadvertently kill processes belonging to other job instances, due to process IDs once owned by the terminating job being reused by the system.

ZD: 15548

==== CL 15587 ====

@FIX: cmdline and cmdrange jobtypes don't report the jobtype version in the job logs

==== CL 15567 ====

@FIX: supervisor_default_max_cpus value was not being applied properly

ZD: 15503

JIRA: QUBE-1528

==== CL 15560 ====

@CHANGE: "modify" operation will print, into the supelog and the job's .hst file, the values of the newly modified parameters

JIRA: QUBE-1318

ZD: 14979

==== CL 15555 ====

@FIX: prevent "upgrade_worker --reset" from printing out "table does not exist" error message.

JIRA: QUBE-817

==== CL 15531 ====

@NEW: add [run_program_and_convert_encoding.pl](#) script, which is a wrapper to run any given program and convert its stdout from and to specified encodings (like UTF-16le to UTF-8).

Added to support 3dsmax batch (i.e., "cmdrange") submissions.

JIRA: QUBE-1210

#####

@RELEASE: 6.8-1a

#####

==== CL 15462 ====

@FIX: removed submission-time check for jobtype existence on the farm, as it was causing false negatives in certain cases and disallowing submissions

ZD: 15328, 15831

#####

@RELEASE: 6.8-1

#####

==== CL 15384 ====

@NEW: add Mac OS X 10.11, aka "El Capitan" support

==== CL 15380 ====

@CHANGE: modification now allowed on "done" jobs

ZD: 15281

==== CL 15347 ====

@FIX: Windows issue where wireless network interfaces are ignored when licenses are verified, causing license keys bound to such interfaces to not work.

#####

@RELEASE: 6.8-0

#####

==== CL 15324 ====

@CHANGE: supervisor on Win32 to build against Perl 5.8 (upgraded from 5.6) to avoid build issues on new build platform.

==== CL 15154 ====

@CHANGE: supervisor now rejects workers that have newer major/minor version than itself.

Such workers will essentially stay in "down" state, or never appear in the host list.

JIRA: QUBE-1341

==== CL 15137 ====

@FIX: Windows qbservice tool to back up existing my.cnf file before writing a new one when invoked with the "--mysqlprepare" option (i.e., via the supervisor installer)

For consistency with the Mac OS X supe installer, the back up file is named "mysql.qubebak.\$\$" where \$\$ is the current process ID (pid).

JIRA: QUBE-1229

==== CL 15077 ====

@NEW: add bin/qbdeleteworkerresources and qbdeleteworkerproperties programs

==== CL 15053 ====

@NEW: Basic admin UI for central prefs

==== CL 15052 ====

@CHANGE: automatically adjust host.processors of all jobs on farms with Designer licensing to 1.

==== CL 15048 ====

@FIX: "ERROR: unable to contact worker." - checkDiskUsage.py throws error when run on a machine which is not running as a worker.

==== CL 15014 ====

@FIX: fixed Python API docstring for deleteworkerresources and deleteworkerproperties

JIRA: QUBE-1322

==== CL 15011 ====

@CHANGE: allow "retry" of "badlogin" jobs (attempts to change their status to "pending")

JIRA: QUBE-642

==== CL 14948 ====

@FIX: "scoped" global resources aren't being tracked in the data warehouse

==== CL 14923 ====

@FIX: decrease the frequency of reporting progress and errors

@CHANGE: only do a file size check on the first 5 frames in a chunk

@FIX: setting fileSizeMin validation size to 0 disables the size checking.

==== CL 14919 ====

@FIX: log parsing not finding any matches in stderr, only stdout

==== CL 14751 ====

@CHANGE: decrease sampling and polling intervals to allow for consecutive fast-running commands to complete quicker, cuts down on application startup time for some apps

==== CL 14750 ====

@CHANGE: python job classes can take option 'prototype' arg in the constructor

==== CL 14749 ====

@CHANGE: child_bootstrapper for python loadOnce jobs is passed in as an argument, allows for application-specific bootstrappers

==== CL 14702 ====

@FIX: add code so that python27.zip is also added to 64-bit supe MSI builds

JIRA: QUBE-1228

==== CL 14698 ====

@NEW: adding python27.zip to be shipped with supervisor's MSI package

JIRA: QUBE-1228

==== CL 14691 ====

@FIX: add code to properly load python 2.7 modules shipped with the supervisor, in python27.zip (which contains files from Python 2.7.10 distribution)

==== CL 14657 ====

@FIX: add missing python27.dll file to supervisor MSI package

JIRA: QUBE-1228

==== CL 14581 ====

@CHANGE: changed ("new") worker behavior when auto-mount drives are unmountable due to duplicate drives.

Now, failed attempts to auto-mount a drive due to the drive letter already being in use will only generate a WARNING message in the workerlog, instead of rejecting the job and sending it back to the supe as "pending".

==== CL 14579 ====

@CHANGE: add more useful info to print to the workerlog when a job is rejected due to duplicate drive mounting (attempt to mount to a drive letter that's already mounting something else)

==== CL 14574 ====

@FIX: Secondary jobs were being dispatched even when supervisor_smart_share_mode is set to NONE

ZD: 14613

==== CL 14528 ====

@FIX: issue when modifying job's "env": "cwd", "umask", and "drivemap" are wiped-- additional fix to allow "env" modification of multiple jobs with a single call to qbmodify()

See also CL14516.

JIRA: QUBE-1161

ZD: 14549

==== CL 14523 ====

@CHANGE: upgraded supervisor's embedded Python to version 2.7.2 on Windows

JIRA: QUBE-1164

==== CL 14518 ====

@CHANGE: worker_boot_delay defaults to 10 seconds on workers running in service mode, on ALL platforms.

JIRA: QUBE-989

==== CL 14516 ====

@FIX: issue when modifying job's "env": "cwd", "umask", and "drivemap" are wiped

JIRA: QUBE-1161

ZD: 14549

==== CL 14514 ====

@FIX: add agenda item (aka "work") status to print properly to the job's history log when it's recalled, because of the instance that's processing being migrated, interrupted, failed, killed, or blocked.

There will be a line like the following in the .hst history log:

```
[Sep 15, 2015 17:09:05] 495670145 work 45765 1 __QUBE_SYSTEM__@supervisor recalled in supervisor by user[] from host[supervisor] on host[shinyambp] (127.0.0.1)
```

Note that this will also show, as expected, when a job instance reaches timeout (if specified) and "failed" by the system.

JIRA: QUBE-829

ZD: 13521

==== CL 14507 ====

@FIX: issue where subst mounted local drives will disappear from Explorer after a job finishes on DU mode workers.

@FIX: also fixed a bug where an already-mounted network/subst drives weren't being detected properly

ZD: 14009

JIRA: QUBE-1030

==== CL 14500 ====

@FIX: issue where cmd* jobtype jobs fail when paths given to QB_CONVERT_PATH() include parentheses

Note: problem was with the command-line tokenizer, QbExpressions::commandtokenize() routine, commonly used by all cmd* jobtypes, not respecting double-quoted and single-quoted strings.

JIRA: QUBE-1139

==== CL 14479 ====

@FIX: QB_CONVERT_PATH() runtime path conversion fails when the path to be converted contains parentheses

==== CL 14473 ====

@FIX: Allow custom algorithms to decide how to preempt SmartShare secondary instances, or just default to using value set in supervisor_smart_share_preempt_policy.

Custom algorithms may define a qb_preemptcmp_secondary() routine to control how secondary jobs are preempted.

ZD: 14472

JIRA: QUBE-1145

==== CL 14406 ====

@FIX: fixed missing job parameters in the job object returned by "qb.jobobj()" (qb.jobobj() in python) in jobtype backends.

The following parameters were added:

queue
max_cpus
omithosts
omitgroups
notes
cpustally
todotally
automigratecount
retrysubjob
retrywork
retrywork_delay
dependency
mailaddress
sourcehost
prod_show
prod_shot
prod_seq
prod_client
prod_dept
prod_custom1
prod_custom2
prod_custom3
prod_custom4
prod_custom5

==== CL 14397 ====

@FIX: performance tweak, cut down on the number of times backends and automated scripts fetch the supervisor config

==== CL 14360 ====

@CHANGE: agenda-based job instance is immediately interrupted, even if the global preemption policy is set to passive, if it hasn't started processing an agenda item

JIRA: QUBE-1077

ZD: 14109

==== CL 14352 ====

@FIX: added QB_FRAME_NUMBER, QB_FRAME_START, QB_FRAME_END, QB_FRAME_STEP, and QB_FRAME_RANGE to be defined in the environment just before a frame is processed

ZD: 14203

==== CL 14326 ====

@FIX: make appropriate invocation of approvemodify (qb_approvemodify() perl routine) for Custom Policy

ZD: 14173

JIRA: QUBE-1082

==== CL 14320 ====

@FIX: catch case in checkUserPermission where traceback error "e" is not defined and an attempt is made to report the error message - occurs when user running the script is not a qube admin

==== CL 14305 ====

@TWEAK: print queuing policy (Internal or custom/Perl) message to supelog

==== CL 14273 ====

@FIX: properly report back failing status when an regex_error is matched early on, but then not found in the last pass through the logs.

==== CL 14207 ====

@FIX: log sections that match an error regex from before an auto-retry are being scanned and matching for errors; now either "qube! - retry/requeue" or "auto-retry" messages trigger a reset

==== CL 14204 ====

@NEW: a script and modules to sync external 3rd-party license server counts with Qube's global resources

@NEW: first external license server modules are for FLEXlm and sesinetd servers

==== CL 14191 ====

@NEW: add path translation to all python-based loadOnce jobtypes

==== CL 14162 ====

@FIX: issue where the supervisor, when starting secondary instances for a job, can preempt more instances than necessary-- i.e., preempt more instances than there are agenda items for the job.

ZD: 13969
JIRA: QUBE-1007

==== CL 14064 ====

@FIX: issue where global time-based callbacks (i.e., "dummy-time-self" callbacks) sometimes not triggering

ZD 13366
JIRA: QUBE-807

==== CL 13971 ====

@CHANGE: add job "name" and "lastupdate" columns to be added at time of job ID creation (available while job is still in "registering" state).

Qube 6.7

#####

@RELEASE: 6.7-3

#####

==== CL 15531 ====

@NEW: add run_program_and_convert_encoding.pl script, which is a wrapper to run any given program and convert its stdout from and to specified encodings (like UTF-16le to UTF-8).

Added to support 3dsmax batch (i.e., "cmdrange") submissions.

JIRA: QUBE-1210

==== CL 15380 ====

@CHANGE: modification now allowed on "done" jobs

ZD: 15281

==== CL 15077 ====

@NEW: add bin/qbdeleteworkerresources and qbdeleteworkerproperties programs

#####

@RELEASE: 6.7-2

#####

==== CL 14581 ====

@CHANGE: changed ("new") worker behavior when auto-mount drives are unmountable due to duplicate drives.

Now, failed attempts to auto-mount a drive due to the drive letter already being in use will only generate a WARNING message in the workerlog, instead of rejecting the job and sending it back to the supe as "pending".

==== CL 14579 ====

@CHANGE: add more useful info to print to the workerlog when a job is rejected due to duplicate drive mounting (attempt to mount to a drive letter that's already mounting something else)

==== CL 14574 ====

@FIX: Secondary jobs were being dispatched even when supervisor_smart_share_mode is set to NONE

ZD: 14613

==== CL 14528 ====

@FIX: issue when modifying job's "env": "cwd", "umask", and "drivemap" are wiped-- additional fix to allow "env" modification of multiple jobs with a single call to qbmodify()

See also CL14516.

JIRA: QUBE-1161
ZD: 14549

==== CL 14523 ====

@CHANGE: upgraded supervisor's embedded Python to version 2.7.2 on Windows

JIRA: QUBE-1164

==== CL 14518 ====

@CHANGE: worker_boot_delay defaults to 10 seconds on workers running in service mode, on ALL platforms.

JIRA: QUBE-989

==== CL 14516 ====

@FIX: issue when modifying job's "env": "cwd", "umask", and "drivemap" are wiped

JIRA: QUBE-1161

ZD: 14549

==== CL 14514 ====

@FIX: add agenda item (aka "work") status to print properly to the job's history log when it's recalled, because of the instance that's processing being migrated, interrupted, failed, killed, or blocked.

There will be a line like the following in the .hst history log:

```
[Sep 15, 2015 17:09:05] 495670145 work 45765 1 __QUBE_SYSTEM__@supervisor recalled in supervisor by user[] from host[supervisor] on host[shinyambp] (127.0.0.1)
```

Note that this will also show, as expected, when a job instance reaches timeout (if specified) and "failed" by the system.

JIRA: QUBE-829

ZD: 13521

==== CL 14507 ====

@FIX: issue where subst mounted local drives will disappear from Explorer after a job finishes on DU mode workers.

@FIX: also fixed a bug where an already-mounted network/subst drives weren't being detected properly

ZD: 14009

JIRA: QUBE-1030

==== CL 14500 ====

@FIX: issue where cmd* jobtype jobs fail when paths given to QB_CONVERT_PATH() include parentheses

Note: problem was with the command-line tokenizer, QbExpressions::commandtokenize() routine, commonly used by all cmd* jobtypes, not respecting double-quoted and single-quoted strings.

JIRA: QUBE-1139

==== CL 14479 ====

@FIX: QB_CONVERT_PATH() runtime path conversion fails when the path to be converted contains parentheses

==== CL 14473 ====

@FIX: Allow custom algorithms to decide how to preempt SmartShare secondary instances, or just default to using value set in supervisor_smart_share_preempt_policy.

Custom algorithms may define a qb_preemptcmp_secondary() routine to control how secondary jobs are preempted.

ZD: 14472

JIRA: QUBE-1145

==== CL 14397 ====

@FIX: performance tweak, cut down on the number of times backends and automated scripts fetch the supervisor config

==== CL 14360 ====

@CHANGE: agenda-based job instance is immediately interrupted, even if the global preemption policy is set to passive, if it hasn't started processing an agenda item

JIRA: QUBE-1077

ZD: 14109

==== CL 14352 ====

@FIX: added QB_FRAME_NUMBER, QB_FRAME_START, QB_FRAME_END, QB_FRAME_STEP, and QB_FRAME_RANGE to be defined in the environment just before a frame is processed

ZD: 14203

==== CL 14326 ====

@FIX: make appropriate invocation of approvemodify (qb_approvemodify() perl routine) for Custom Policy

ZD: 14173

JIRA: QUBE-1082

==== CL 14320 ====

@FIX: catch case in checkUserPermission where traceback error "e" is not defined and an attempt is made to report the error message - occurs when user running the script is not a qube admin

==== CL 14305 ====

@TWEAK: print queuing policy (Internal or custom/Perl) message to supelog

==== CL 14273 ====

@FIX: properly report back failing status when an regex_error is matched early on, but then not found in the last pass through the logs.

==== CL 14207 ====

@FIX: log sections that match an error regex from before an auto-retry are being scanned and matching for errors; now either "qube! - retry/requeue" or "auto-retry" messages trigger a reset

==== CL 14204 ====

@NEW: a script and modules to sync external 3rd-party license server counts with Qube's global resources

@NEW: first external license server modules are for FLEXlm and sesinetd servers

#####

@RELEASE: 6.7-1

#####

==== CL 14162 ====

@FIX: issue where the supervisor, when starting secondary instances for a job, can preempt more instances than necessary-- i.e., preempt more instances than there are agenda items for the job.

ZD: 13969

JIRA: QUBE-1007

==== CL 14064 ====

@FIX: issue where global time-based callbacks (i.e., "dummy-time-self" callbacks) sometimes not triggering

ZD 13366

JIRA: QUBE-807

#####

@RELEASE: 6.7-0

#####

==== CL 13971 ====

@CHANGE: add job "name" and "lastupdate" columns to be added at time of job ID creation (available while job is still in "registering" state).

==== CL 13959 ====

@CHANGE: More efficient job refresh. Less tax on the supervisor during refresh

==== CL 13871 ====

@FIX: all cmds are passed to subprocess.Popen as raw strings now, no longer attempt to trap escape characters in Windows cmdlines,

==== CL 13870 ====

@NEW: add support for new 'registering' job status to data warehouse

==== CL 13869 ====

@NEW: add support for new 'registering' status to WranglerView

==== CL 13845 ====

@FIX: the upgrade_supervisor upgrade/pre-install DB converter program (pre-6.5 to 6.5) was incorrectly adding the subjobN table's "allocations" column with the type set to "integer", where it should have been "long text".

JIRA: QUBE-804

==== CL 13843 ====

@FIX: fixing newly discovered malloc/free (delete) error (essentially a segfault) when handling job submissions of jobs with "children" jobs.

==== CL 13840 ====

@TWEAK: added code to print, for each submission, what mode of table creation was requested (deferred or immediate), and how many jobs were submitted, into the supelog.

==== CL 13839 ====

@CHANGE: Python API: merged the "len" and "pos" features of qb.stdout_len() and qb.stderr_len() functions into the original qb.stdout() and

qb.stderr() functions.

* Python API: merged the features of qb.stdout_len() and qb.stderr_len() functions into the original qb.stdout() and qb.stderr() functions.

* Python API: removed qb.stdout_len() and qb.stderr_len()

* Python API: refactored and cleaned up some code

JIRA: QUBE-655

==== CL 13834 ====

@NEW: add new job status "registering" for jobs submitted with "deferTableCreation" qbsubmit() API option.

Jobs are initially assigned this status when they are submitted with the "defer table creation" option enabled in the call to the qbsubmit() API routine. A job will be in this status until the system finishes creating all DB tables for it in the background, which is when the job finally comes into being.

JIRA: QUBE-743

==== CL 13828 ====

@NEW: modify supervisor to allow deferring DB table creation when handling job submissions, to speed up response time to the caller/submitter.

JIRA: QUBE-743

==== CL 13805 ====

@NEW: For 6.8 - ArtistView Submission UI now works with central preferences server. Submission parameters can have their values set, values mandated, label changed, or be hidden via central prefs.

==== CL 13776 ====

@NEW: slight improvements to the utility of the new "len" and "pos" parameters for stdout/stderr log querying APIs (C++ API's qbstdout() and qbstderr(), and Python API's qb.stdout_len() and qb.stderr_len())

* an empty 'data' is returned when len=0 (used to return all data-- that is now achieved by specifying a negative len)

* "len" defaults to -1 (which means "get me untruncated data"). ("pos" still defaults to 0, which means the beginning of the log file)

* When a negative value is specified for "pos", the file seek is done backwards from the end of the file. (QbFile::loadpos())

==== CL 13774 ====

@NEW: add "fullsize" to QbLog object, so the full size of the stdout/err log file for a job instance is always returned on qbstdout/err queries. (and consequently on Python API's qb.stdout_len and stderr_len calls also).

==== CL 13758 ====

@CHANGE: Qube-installed version of MySQL is now stopped and started by launchctl, allows for the installation of a supervisor on OS X Yosemite and later.

==== CL 13751 ====

@FIX: infinite-loop bug in C++ core API when "pos" value given to qbstdout/err() is negative.

==== CL 13750 ====

@NEW: add "qb.stdout_len()" and "qb.stderr_len()" routines where the position to start reading the job's stdout/err log file, as well as how many bytes to read, may optionally be specified.

Example:

```
>>> qb.stdout_len("1234.0", pos=0, len=20)
{'subid': 0, 'data': '[Apr 3, 2015 15:34:4', 'jobid': 1234]
```

@FIX: also fixed a minor bug in an error message that gets printed

JIRA: QUBE-655

==== CL 13737 ====

@FIX: add code to prevent premature retiring of running instances in requestwork(), due to the system incorrectly determining that a job has decreased the "cpus" count.

ZD: 13452

==== CL 13736 ====

@TWEAK: Adding comments and slightly better logging messages for worker heartbeat related areas of code.

==== CL 13735 ====

@CHANGE: V-Ray DBR jobs in 3dsMax can now start immediately, hosts can join in as they become available

==== CL 13717 ====

@FIX: Sketchup 2015 on Windows is now a 64-bit application, don't just look in C:\Program Files (x86) for Sketchup executable

==== CL 13698 ====

@INTEG: rel-6.6>main,CL13694

@NEW: add modify capability for job's "env".

* qbmodify() C++ API

* qb.modify() Python API

Python example:

```
import qb
jobID = 1234
modJobProps = {}
modJobProps['env'] = {"MYVAR": "MYVAL", "MYVAR2": "OOGA"}
qb.modify(modJobProps, jobID)
```

QUBE-221

==== CL 13667 ====

@INTERNAL FIX: fixed const-ness of some method arguments in QbPolicy and QbPolicyPerl modules

==== CL 13666 ====

@NEW: add perl 5.18 support for platforms that come preloaded with it (i.e., MacOS X 10.10 "Yosemite")

QUBE-756

==== CL 13658 ====

@FIX: problem with custom queuing algorithms where the qb_jobcmp, qb_hostcmp, and qb_reject perl routines are not properly being invoked when necessary.

ZD: 13231

==== CL 13598 ====

@CHANGE: starting up a pythonChildHandler to manage a "LoadOnce" application does not implicitly start the application, must now call the handler's start(), and pass server port number to the subprocess args.

Qube 6.6

#####

@RELEASE: 6.6-4

#####

==== CL 14162 ====

@FIX: issue where the supervisor, when starting secondary instances for a job, can preempt more instances than necessary-- i.e., preempt more instances than there are agenda items for the job.

ZD: 13969

JIRA: QUBE-1007

==== CL 14064 ====

@FIX: issue where global time-based callbacks (i.e., "dummy-time-self" callbacks) sometimes not triggering

ZD 13366

JIRA: QUBE-807

==== CL 13871 ====

@FIX: all cmds are passed to subprocess.Popen as raw strings now, no longer attempt to trap escape characters in Windows cmdlines,

==== CL 13845 ====

@FIX: the upgrade_supervisor upgrade/pre-install DB converter program (pre-6.5 to 6.5) was incorrectly adding the subjobN table's "allocations" column with the type set to "integer", where it should have been "long text".

JIRA: QUBE-804

==== CL 13823 ====

@FIX: timing issue causing resetting/zero-ing the start time of agenda items when they are preempted, that can result in WV to incorrectly display huge elapsed frame times of 5500+ days.

ZD: 13409

==== CL 13737 ====

@FIX: add code to prevent premature retiring of running instances in requestwork(), due to the system incorrectly determining that a job has decreased the "cpus" count.

ZD: 13452

==== CL 13736 ====

@TWEAK: Adding comments and slightly better logging messages for worker heartbeat related areas of code.

==== CL 13717 ====

@FIX: Sketchup 2015 on Windows is now a 64-bit application, don't just look in C:\Program Files (x86) for Sketchup executable

==== CL 13667 ====

@INTERNAL FIX: fixed const-ness of some method arguments in QbPolicy and QbPolicyPerl modules

==== CL 13666 ====

@NEW: add perl 5.18 support for platforms that come preloaded with it (i.e., MacOS X 10.10 "Yosemite")

QUBE-756

==== CL 13658 ====

@FIX: problem with custom queuing algorithms where the qb_jobcmp, qb_hostcmp, and qb_reject perl routines are not properly being invoked when necessary.

ZD: 13231

#####

@RELEASE: 6.6-3

#####

==== CL 13391 ====

@FIX: checkDiskUsage.py missing from installation packages on linux, should be installed by qube-core.rpm

==== CL 13381 ====

@NEW: Added "worker_mode = desktop" or "worker_mode = service" to print to the workerlog at worker startup

==== CL 13365 ====

@FIX: "IOError: [Errno 5] Input/output error" occurs when trying to print to stdout or stderr in job instance with very verbose logging occurring

==== CL 13363 ====

@FIX: catch case where memusage datacollector returns agenda item name as a space

==== CL 13320 ====

@INTEG: main -> rel-6.6

@FIX: add support for new 'exiting' state into data warehouse schema

==== CL 13252 ====

@NEW: add support for C4D R16

==== CL 13245 ====

@FIX: 'regex_error' matches against an error message that precedes a retry operation, fails the frame or instance even if it completes successfully this time.

==== CL 13233 ====

@FIX: don't report memory usage for items where the datacollector can't determine the agenda item's name

==== CL 13214 ====

@FIX: modified rpm spec file creating script to NOT assume .pyc/.pyo files for CentOS/RHEL 6.6 and above.

==== CL 13208 ====

@FIX: bug where preemption code will not properly work when running jobs have a "greedy" host.processors reservations (such as "1+")

For example, this bug was causing high priority jobs with a requirement of "host.processors.used eq 0" to NOT preempt low priority jobs running on a multi-jobslot host with "host.processors = 1+".

ZD: 12512

JIRA: QUBE-632

==== CL 13126 ====

@FIX: add code to prevent more random worker crashes on Windows DU mode.

DU mode worker was occasionally found to kill it's own worker.exe and workertray.exe processes when removing job processes.

==== CL 13066 ====

@NEW: Add SketchUp batch-render integration as an AppFinder job

==== CL 13058 ====

@FIX: When performing path translation, double-quote a converted path if it contains spaces and is not already quoted

#####

@RELEASE: 6.6-2

#####

==== CL 13031 ====

@FIX: manually reverting change made in CL12255 (for 6.5-3) to prevent possible job zombie processes, as it can cause a worse issue (BSOD) at times.

ZD12198

==== CL 13018 ====

@FIX: leading backslash on UNC paths is erroneously trimmed in python-based jobtypes during worker_path_map path conversion

==== CL 13017 ====

@FIX: path conversion not being done, is only applied if worker_path_map is present. Now is always attempted, will return the unaltered path if no path mappings defined.

==== CL 12998 ====

@FIX: corrected the name from "license_mode" to "license_model" in the return value of qb.ping(asDict=True) Python API call.

==== CL 12996 ====

@CHANGE: added "designer" or "unlimited" to be included in the string returned by the qbping() API call

@CHANGE: The python API routine, qb.ping(), when invoked as "qb.ping(asDict=True)", will return new dict elements "license_type" and "licence_mode", to represent the license type ("unlimited" or "designer"), and license mode ("subjob" or "host"), respectively. ("licenses_type", which incorrectly used to represent "license_mode", has been deprecated).

JIRA: QUBE-544

==== CL 12989 ====

@NEW: add supervisor_license_model and _license_type to qb.utils.ENUMS api module to provide mapping between integer and human-readable values

==== CL 12975 ====

@NEW: add "designer" and "unlimited" licensing support.

When "designer" licensing is in effect, as determined by the "type" field in the license file's supervisor section, all worker nodes are forced to have worker_cpus (i.e., jobslots) of 1. Setting that parameter in the qb.conf or qbwrk.conf has no effect.

JIRA: QUBE-538

#####

@RELEASE: 6.6-1

#####

==== CL 12839 ====

@FIX: install_worker.ubuntu would halt the installation process if the "qubeproxy" user exists but is not in /etc/passwd (which is valid when using networked authentications such as NIS and LDAP)

==== CL 12837 ====

@CHANGE: Linux supervisor install scripts fixes

@FIX: added code so that the install_supervisor.ubuntu script will not halt the entire installation if mysqld wasn't already running

@FIX: Removed stale code that added GRANT for root@'%'

==== CL 12831 ====

@FIX: added fix to work around a MySQL issue on Linux, where Unix socket connections can corrupt queries passed into it, making jobs disappear from GUI, or otherwise leave them in odd states ("failed" jobs with "running" instances, etc.)

The fix prohibits the use of Unix sockets on Linux, by overriding the value

of "database_socket" if set, and by disallowing setting "database_host" to "localhost" in qb.conf

Also changed the default values of database_host to "127.0.0.1" and database_socket to "" on Linux.

Modified so that the database parameters (not all) are printed to the supelog and in the output of "qbadmin s -conf".

#####

@RELEASE: 6.6-0

#####

==== CL 12767 ====

@CHANGE: add/modified code so that qbadmin prints out human-readable strings, instead of just the integer representation, for supervisor_smart_share_mode, supervisor_smart_share_preempt_policy, supervisor_preempt_policy, supervisor_verbosity, supervisor_license_model, supervisor_default_security, and supervisor_default_group_security

@INTERNAL: changed the license model to be represented by an enum list, instead of just a #define.

JIRA: QUBE-501

==== CL 12700 ====

@FIX: fixed "WARNING: qb_default_string() unknown value[502]" message printed to supelog and workerlog

JIRA: QUBE-477

==== CL 12651 ====

@CHANGE: mxi cleanup operation deletes all but the target (merged) .mxi, uses a (required) external script

==== CL 12640 ====

@TWEAK: add human-readable message to print to the workerlog when task_for_pid() and host_statistics() routines return failure (for OS X)

==== CL 12609 ====

@NEW: added environment variables QB_FRAME_STATUS and QB_INSTANCE_STATUS to be set just before postflights are run.

These will be set to the status of the last-processed agenda item and instance, respectively, to values such as "complete" and "failed".

==== CL 12608 ====

@NEW: added code to set "QB_SYSTEM_EXIT_CODE" environment variable to the return value of the "system()" function, when invoked via "qbsystem()".

This, in particular, should be useful when writing postflight programs for cmdline or cmdrange backend jobs, to see if the last job process ran successfully or not.

==== CL 12594 ====

@FIX: removed reference to the now-removed "dispatch_one_subjob" flag

==== CL 12566 ====

@FIX: failure when backing up or creating a new version of a config file should raise an error dialog, rather than just print the error to the WV logPane.

==== CL 12561 ====

@NEW: add Universal Callback feature. See online docs for details on usage.

JIRA: QUBE-233

==== CL 12557 ====

@NEW: add supervisor_universal_callback_path qb.conf parameter, which defaults to \$QBDIR/callback

JIRA: QUBE-233

==== CL 12551 ====

@FIX: Fix Linux rpm uninstall for worker and supervisor so that their service is stopped during uninstall

==== CL 12547 ====

@CHANGE: modified the default my.cnf file that gets created on Windows, which includes getting rid of the "skip-grant-tables" option

JIRA: QUBE-251, QUBE-405

==== CL 12546 ====

@INTERNAL: modified how the supe MSI (actually, the qbservice command that gets invoked by it) stages things in order to call the "mysql_upgrade" utility.

Two ALTER statements were also added per the recommendation in the MySQL 5.1 -> 5.5.32+ upgrade doc.

JIRA: QUBE-251

@INTERNAL: modification

modified added (a lot of) code to make the stop_service() routine to block until the service has actually stopped, or the operation times out.

==== CL 12544 ====

@CHANGE: regex matches in all python-based jobtypes are now case-insensitive

==== CL 12528 ====

@FIX: qb.conf.template's commented-out default values for supervisor_heartbeat_interval and supervisor_heartbeat_timeout

==== CL 12527 ====

@INTERNAL: adding code to run "mysql_upgrade" when the supervisor installer installs mysql.

JIRA: QUBE-251

==== CL 12517 ====

@NEW: migrating Windows platforms (both 32- and 64-bit) to MySQL 5.5.37

modified .vcxproj files to point to the new version

JIRA: QUBE-251

==== CL 12509 ====

@NEW: Supervisor should check the disk space free for MySQL datadir (and log dir if local) on a regular basis, mail or log warnings if they're getting full

==== CL 12497 ====

@NEW: add "Smart Share" feature (aka "balanced auto-expand")

JIRA: QUBE-167

==== CL 12488 ====

@NEW: add '-g/--grep' to qbtail.py, acts like "tail -f <file> | grep", supports basic regex syntax

==== CL 12482 ====

@NEW: Supervisor and worker init scripts for Ubuntu.

==== CL 12478 ====

@TWEAK: added code to print supe config params after a "reread" is done.

==== CL 12472 ====

@NEW: python-based "Load Once" jobs now supported on Windows

==== CL 12467 ====

@FIX: timing issue causing instances to enter "QB_PREEMPT_MODE_FAIL" limbo state.

Job instances being preempted (interrupted, killed, etc) before their proxy process had a chance to properly start up would cause the supe to put the instance in a "QB_PREEMPT_MODE_FAIL" limbo, as evidenced in repeated error messages like the following in the supelog:

"ERROR: requestWork(): subjob[2550.7] has preempt mode of QB_PREEMPT_MODE_FAIL. advising subjob to wait (and retry later)"

==== CL 12457 ====

@CHANGE: removed the "dispatch_one_subjob" flag

==== CL 12442 ====

@FIX: Fixed a corner-case MySQL permission problem with OS X & Linux supervisor and the qube_readonly user.

Fixed by adding a "GRANT SELECT" with an explicit hostname (fetched via "SELECT @@hostname"), as in:

GRANT SELECT ON *.* TO 'qube_readonly'@'mysqlserverhostname'

JIRA: QUBE-438

==== CL 12358 ====

@FIX: Fixed example python scripts so import of qb module will work in most cases.

==== CL 12347 ====

@FIX: pyCmd* jobtypes report all subsequent frames as failing when a 'regex_error' is matched and a frame is marked as failed

==== CL 12339 ====

@FIX: fixed inaccurate worker host memory reporting on Windows platforms

ZD: 11367

==== CL 12338 ====

@TWEAK: added [work.id](#) to also print to the log in addition to the [work.name](#) in QbDistribute.cpp updateWork() in the code that examines retry

==== CL 12333 ====

@FIX: worker shutdown code (QbWorker::hostShutdown() and sendHostReport()) will now give up a lot quicker when being unable to contact the supervisor, instead of retrying for a long time.

==== CL 12322 ====

@FIX: issue where job instances don't terminate properly when very early kill/interrupt orders come in.

Sometimes interrupts and kills can come in before the worker has a chance to properly complete the launching process of the proxy.exe process and its main thread, causing unexpected behavior, such as a never-dying job instance.

ZD: 11409

==== CL 12315 ====

@FIX: bug in initialization code of the QbJob class that messed up comparisons of jobs when sorting, which, among other things, prevented FIFO/FCFS ordering to be compromised. Now FIFO dispatching behavior should be more closely followed by jobs of equal priority (although not 100% strictly, due to the nature of the multithreaded architecture of the supervisor).

ZD: 11259

@INTERNAL TWEAK: added more debugging code to QbSupervisorQueue module.

==== CL 12311 ====

@FIX: adding in ubuntu support: use bash explicitly rather than sh, specify 'awk' in location found on all OS's

==== CL 12306 ====

@FIX: issue where auto-expanded subjobs (instances) don't inherit the "retrysubjob" value set in the parent job, causing them NOT to auto-retry properly on failure.

ZD: 11292

==== CL 12298 ====

@FIX: Python API routines, such as qb.retrywork(), expecting workID as input would behave erroneously (such as retrying ALL agenda items on ALL jobs) when input a subjobID instead. Vice versa for routines expecting subjobIDs, such as qb.retry().
These

ZD: 11372

==== CL 12295 ====

@NEW: add support for new 'exiting' status

==== CL 12286 ====

@INTERNAL: moving QbJobHostReference module to "common"

==== CL 12272 ====

@FIX: unreliable behavior when frequently modifying "cpus" of jobs up and down.

ZD: 11288

==== CL 12257 ====

@FIX: bug where auto-expand subjobs are incorrectly auto-retired, and in turn caused them NOT to expand any more.

ZD: 11217

==== CL 12255 ====

@FIX: issue where, if some intermediate job processes crash and die unexpectedly, other job processes may be missed by the cleanup code and left behind as zombies.

ZD: 11236

==== CL 12250 ====

@FIX: WorkerConfigFile makes a better effort at finding the worker config file, previously would save to default location when the file is actually in a non-default location as specified by supervisor_worker_configfile.

==== CL 12242 ====

@FIX: fixed incorrect const-ness in C++ QbJob module

==== CL 12237 ====

@FIX: avoid inserting duplicated values into the 'outputPaths' for a frame when retried

==== CL 12232 ====

@FIX: "UnboundLocalError: local variable 'qb' referenced before assignment" - issue experienced by single customer on linux, re-importing qb module in main() resolves the issue. ZD# 11218

==== CL 12230 ====

@FIX: additional fixes to remedy "retrywork" issue with maya (and possibly other Perl-API based) jobs. See also the previous CL12228

==== CL 12228 ====

@FIX: automatic retry of agenda via "retrywork" not working properly in perl-based backends.

ZD: 11167

==== CL 12226 ====

@FIX: Fix issue where job_cleanup script would fail if run on a supervisor that did not have the MySQLdb python module installed.

==== CL 12219 ====

@FIX: "sre_constants.error: bogus escape (end of line)" - python-based jobs can crash on Windows at startup if path wrapped in QB_CONVERT_PATH() ends with a fwd-slash and has being converted to a back-slash

==== CL 12215 ====

@CHANGE: allow Perl/Python API access to "agenda_timeout" value using the symbol "agendatimeout" as an alias.

JIRA: QUBE-395

==== CL 12211 ====

@CHANGE: add "perl" and "python" to the default supervisor_language_flags

@CHANGE: add "auto_wrangling" to the default value of supervisor_job_flags, to turn ON auto-wrangling by default

ZD: QUBE-386, QUBE-229

==== CL 12210 ====

@CHANGE: add "admin" privilege to default users, but for new installs only

@INTERNAL: refactored/tidied up the config_main code a bit.

JIRA: QUBE-248

==== CL 12207 ====

@CHANGE: made the supervisor_language_flags dynamically modifiable (i.e. "qbadmin s -reread"-able)

ZD: QUBE-357

==== CL 12206 ====

@NEW: allow "qbmodify" of the following additional fields: agenda_timeout, retrysubjob, retrywork, retrywork_delay, mailaddress

@CHANGE: the qbmodify command, and the modify() routines in the C++, Perl, and Python API were also changed to complete this improvement.

QUBE-368

==== CL 12177 ====

@FIX: Additional changes to support proper Windows privilege enabling, added in CL12176

==== CL 12176 ====

@FIX: Add call to Windows' AdjustTokenPrivileges() to explicitly enabled required privileges before launching job instance (proxy) process

==== CL 12123 ====

@CHANGE: made "stub_optimize" supervisor flag to be disabled by default.

==== CL 12117 ====

@INTERNAL: add QbTableVersion9 to upgrade_worker.vcxproj for Windows builds

==== CL 12098 ====

@FIX: support negative frame range in QB_* token parsing

==== CL 12082 ====

@FIX: issue where "modify"-ing the "cpus" value of a running job may incorrectly retire more instances than asked for.

This was due to race conditions of supe threads, and in extreme cases, was prematurely retire-ing ALL instances of a job while there are still pending agendas, resulting in the job's instances to be all "complete" but the job itself to become "failed" since there are still pending agendas.

ZD: 10868

==== CL 12072 ====

@NEW: added flight-check support for pythonChildBackEnd.py-based jobtypes (i.e., pyHoudini and pyNuke)

@NEW: modified pyframe and helloWorld examples to properly support flight-checks

QUBE-254

==== CL 12065 ====

@INTERNAL TWEAK: added/modified/corrected comments and symbol names for readability

==== CL 12064 ====

@INTERNAL TWEAK: modified/corrected comments and symbol names for readability

==== CL 12056 ====

@NEW: add flight-check support to Perl and Python APIs (accessors for the job object parameters, "preflights", "postflights", "agenda_preflights" and "agenda_postflights").

QUBE-254

==== CL 12055 ====

@NEW: added flight-check feature, both job-level and agenda-level pre- and post-flights that run on workers before/after running the actual job instance or agenda item.

* site-admins may install flight-check scripts/programs to a location on the worker, pointed to by "worker_flight_check_path", a new [qb.conf/qbwrk.conf](#) parameter, which defaults to \$QBDIR/flightCheck/. Job preflights and postflights, and agenda preflights and postflights, will be searched at the execution of every job, in \$worker_flight_check_path/{instance,agenda}/{pre,post}.

* Note: flight-check scripts/programs must be executable (have the executable bit set) on Unix (OS X & Linux) platforms.

* Tip: *.txt files found in the flight-check folders are ignored.

* Jobs may also specify any number of job- and/or agenda-level pre/postflights at submission time. With the qbsub command, for example, the "-preflights", "-postflights", "-agenda_preflights" and "-agenda_postflights" can be used.

* flight-check programs should return 0 to indicate success, and non-zero for failure.

* if a job-level preflight fails, the instance is reported as failed, and the actual instance returns without running the job process.

* if a job-level postflight fails, the instance is reported as failed.

* if an agenda-level preflight fails, the agenda item is reported as failed, and its processing is skipped and the instance will move on to the next agenda item.

* if an agenda-level postflight fails, the agenda item is reported as failed.

QUBE-254

==== CL 12052 ====

@NEW: qbjobs to print flight-check info when "-l" option is given

QUBE-254

==== CL 12050 ====

@CHANGE: modified pyCmdrange back-end to handle the "failed" status that may be now returned by qb.requestwork() when an agenda preflight fails.

QUBE-254

==== CL 12045 ====

@FIX: fixed a bug where, for an empty string DB record, the DB::string() (e.g. called as in "itm.string(ELEM_NEXT)") routine sometimes returns the value of the previous non-empty record.

==== CL 12016 ====

@FIX: worker and supervisor install do not register for all users on Windows

==== CL 12006 ====

@FIX: ERROR 1146 (42S02) at line 87 in file: './create_job_fact.sql': Table 'pfx_stats.memusage' doesn't exist - swap order of table assignment and creation, some versions of MySQL are error'ing

==== CL 11993 ====

@CHANGE: modify QbApi.cpp's qbsystem() routine to return, as with system(3), -1 on error, or the exit code of the command that was run.

@CHANGE: modify all internal calls to qbsystem() (in types/cmd{range,line,file,grid,multi}/execute.cpp) to reflect the above change.

@CHANGE: general clean up of the code that determines the return value of qbsystem() routines in QbApi.cpp

@CHANGE: modify QbProxy::run() to expect the execute() functions in the exec_binding library (i.e., perl, python, dll, dso, dylib) to return 0 for success and non-zero for errors.

@CHANGE: modify the execute() routine in Qb{Python,Perl,Dso,Dll,Dylib}Lang.cpp modules to reflect the above change-- i.e., they return 0 for success and non-zero for errors.

==== CL 11989 ====

@FIX: worker_drive_map and worker_path_map not correctly saved via "Configure local host", format to match API update local config expectations

==== CL 11987 ====

@FIX: localized the _user_duties and _prgp_duties IntHash variables to the queue_reject() routine for thread-safety, from being data members of the supervisor class.

ZD: 10342

==== CL 11986 ====

@FIX: added code to appropriately handle timing issues where a command, such as preemption, can be issued multiple times by different threads on the same running subjob, leaving those jobs to be in odd states. One common symptom was seeing the "aberrant report" message in the supelog, and those jobs getting stuck in the "running" state despite all the frames being 100% done.

==== CL 11985 ====

@FIX: converseWorkerWithRetries() and converseSubSupervisorWithRetries() routines were fixed so that they properly return success when there are no communication errors. These routines were retrying when the server responded with a rpy.tag() of QB_MESSAGE_ERROR, which doesn't mean there was a communication error, but rather means that the server encountered some general internal error, causing unwanted retries.

ZD: 10527

==== CL 11982 ====

@FIX: contradictory job log entries saying a failed frame is being reported as complete when a few lines ago it was actually (correctly) reported as failed.

==== CL 11980 ====

@FIX: QB_CONVERT_PATH() not getting evaluated when worker_path_map is undefined or empty

==== CL 11963 ====

@FIX: catch jobs with package data the cause _qb.packageStrToDict to raise an exception

==== CL 11961 ====

@CHANGE: add additional sanity checks to cleanup script, limit number of log directory deletions to a fraction of total jobs in qube, can be overridden by option flag.

==== CL 11957 ====

@CHANGE: refactored and cleaned up proxy program's run() routine that dispatches different execution module depending on the "execute_binding" of the jobtype.

Removed the following legacy bindings: StaticPerl, Net (dot net), Tcl, and qbsystem.

==== CL 11931 ====

@CHANGE: create the backfill_fact (supervisor dispatch efficiency) dataWarehouse "12-hour" table every 5 minutes rather than every 15 to keep the chart data more current - full-range table is small enough to support this

==== CL 11915 ====

@FIX: fixed cross-dependency created in CL11893.

JIRA: QUBE-176

==== CL 11908 ====

@CHANGE: changed/added code to set up the following default my.cnf parameters

all OSs:

query_cache_size = 0 # disable the query cache, hit rate is almost 0% due to qube being very write-intensive
thread_cache_size = 16 # acts like supervisor_idle_threads
Linux-only

table_open_cache = 2500 # mysql will cache the file handles necessary to hold this number of tables f/h's
open_files_limit = 50000 # table_open_cache will drive the number of open files, MyISAM needs a max of 2 per table, but MySQL can also open other files past the table_open_cache*2 value - refer to: <http://dev.mysql.com/doc/refman/5.1/en/table-cache.html>

JIRA: QUBE-175

==== CL 11899 ====

@FIX: made the path map translations case-insensitive on OS X and Windows platforms.

@NEW: added 3rd optional parameter to QbString::replace(), which specifies the case-sensitivity, which defaults to TRUE.

JIRA: QUBE-177

==== CL 11898 ====

@NEW: add "scripts/find_corrupt_jobs.py" script, which finds jobs with corrupt database records (i.e. missing sub-tables, such as Nsubjob and Nwork) in the supervisor MySQL DB.

ZD: 10438

==== CL 11895 ====

@NEW: exposed the C API routine "qbisadmin()" as "qb.isadmin()" in Python API and "qb::isadmin()" in Perl API.

JIRA: QUBE-174

==== CL 11893 ====

@CHANGE: "qbadmin {s|w} -configuration" now displays both the integer AND string values of all "_flags" (such as "supervisor_flags") parameters for readability

JIRA: QUBE-176

==== CL 11856 ====

@FIX: added code to fix jobs getting stuck in the "dying" state, that can occur due to race conditions.

Dispatched instances of jobs that were requested to be "killed" before they properly finished starting up on the workers were ending up getting stuck in the "dying" state.

ZD: 10369

==== CL 11850 ====

@FIX: C4D AppFinder jobs crash when paths or filenames wrapped in QB_CONVERT_PATH() start with a number

==== CL 11829 ====

@FIX: Issue with grid jobs where some instances would start running multiple times on the dispatched host, causing the job to eventually fail.

ZD: 10325

==== CL 11828 ====

@FIX: graceful worker shutdown on Windows (service mode)

==== CL 11820 ====

@FIX: disable permission check of worker_logpath, as it was creating false-alarms and putting the worker to be in panic mode unnecessarily.

ZD: 5445 5236

BUGZID: 63683

See also CL9234

==== CL 11815 ====

@FIX: on Linux in the /etc/init.d/worker script, we're now allowing a longer timeout (15 seconds) for the worker to shutdown cleanly before forcefully killing (i.e. "kill -9") the processes.

The default short timeout of 3 seconds was not sufficient on many systems for all child worker threads to exit and the main thread to release the running subjobs and report to the supervisor that it's "down".

JIRA: QUBE-90

==== CL 11807 ====

@FIX: added dependency ("requires") on the "expat" package for qube-core RPM packages.

JIRA: QUBE-68

ZD: 8499

==== CL 11801 ====

@FIX: fixed qjoborder() routine so that it respects the queuing algorithm's job-host pair rejection routine, queuereject().

This bug, for example, was causing the routine to return jobs that shouldn't qualify to run on the given host because of the "worker_restrictions" settings of the worker.

ZD: 10231

JIRA: QUBE-158

==== CL 11795 ====

@FIX: issue where the python API qb.convertpath() will cause a bus error (crash) in the caller, if called with no args.

@FIX: issue where the 2-argument invocation of qb.convertpath() was not working, and may cause a bus error. Turned out to be a bug in the internal conversion routine _qb_py_dict_pathmap().

==== CL 11793 ====

@FIX: bug with modifying user and group permissions.

Operations such as adding or deleting a group or users would generate an error message in the supelog, like the following:

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: group permissions modified for user foobar by user USERNAME

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: ERROR: database query error: 127.0.0.1 via TCP/IP - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ') = 'test' AND LOWER(user) = 'foobar'" at line 1 (1064)

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: SELECT access FROM grp WHERE valid = 1 AND LOWER(name)) = 'test' AND LOWER(user) = 'foobar'

JIRA: QUBE-157

==== CL 11790 ====

@FIX: fixed inaccurately reported host.processor_speed (CPU frequency in MHz) property on OS X workers.

JIRA: QUBE-153

==== CL 11788 ====

@CHANGE: added "GRANT" statement to "GRANT SELECT ON *.*" to the qube_readonly user on "localhost".

JIRA: QUBE-105

==== CL 11771 ====

@FIX: problem where it was impossible to undefine worker_properties and worker_resources once they were defined in qbwrk.conf or qb.conf, even if the lines were removed from the config files.

JIRA: QUBE-85

ZD: 10227

==== CL 11767 ====

@FIX: Setting "worker_cpus=0" or removing a "worker_cpus=N" line from qbwrk.conf had no effect, and the previous setting would get stuck.

JIRA: QUBE-80, QUBE-112

==== CL 11748 ====

@FIX: helloWorld example jobtype can't create job archive file job.qja below QBDIR/examples, area is read-only, write to a temp directory

==== CL 11740 ====

@CHANGE: remove dependencies between Windows MSI installers, any qube component can be installed or uninstalled independent of the others

==== CL 11733 ====

@FIX: qbtail.py now prints out the help screen if run without arguments.

==== CL 11720 ====

@FIX: add array size check before "delete [] _data;" to workaround MS compiler bug where delete[]-ing a zero-length array of objects causes crashes if the object's class has a virtual destructor.

This was causing mysterious supervisor crashes on Windows only.

ZD: 9718

==== CL 11687 ====

@NEW: add support for Adobe's Creative Cloud 'CC' version numbering scheme

==== CL 11685 ====

@FIX: erroneous "timecumulative" of instances (subjobs)

JIRA: QUBE-148

==== CL 11682 ====

@FIX: removed "Configure Qube" menu item from workertray.exe

==== CL 11654 ====

@NEW: added worker_boot_delay to qb.conf.template

==== CL 11631 ====

@CHANGE: modified thhe new worker_boot_delay (CL11605) to default to 0 seconds on ALL platforms.

ZD: 9386

JIRA: QUBE-118

==== CL 11614 ====

@FIX: The "Administrator" user didn't properly get permissions to manipulate others' jobs, due to case sensitivity.

JIRA: QUBE-142

==== CL 11605 ====

@FIX: added worker_boot_delay to qb.conf, which specifies the number of seconds to artificially delay (i.e. sleep) before the worker boots.

This was added in order to work around a boot-time timing issue with networking/hostname assignment on the Mac OS X platform, where many machines are incorrectly identified as "localhost".

As such, by default, worker_boot_delay is set to 30 seconds on the Mac, and 0 (i.e., no delay) on other platforms.

To override the default, worker_boot_delay *must* be set in the local qb.conf file on each worker-- setting it in qbwrk.conf won't work.

ZD: 9386

JIRA: QUBE-118

==== CL 11484 ====

@FIX: appFinder jobs error out at setup with "raise error, v # invalid expression, bad group name" exception raised by python re module

==== CL 11470 ====

@FIX: random Worker crashes on Windows DU mode-- worker "committing suicide", i.e. killing it's own worker.exe process when removing job processes.

==== CL 11460 ====

@FIX: added code to "generate movie" jobs, to support frame ranges that don't start at 1 with conversions using ffmpeg

@FIX: also added a leading "." to the movie_ext choice strings, which is required.

ZD: 9745

==== CL 11428 ====

@CHANGE: large re-write, not backward-compatible with previous argument spec

@CHANGE: will only delete jobs where all jobs in the job's pgrp completed more than X days ago

@CHANGE: added --removeOrphanedLogs functionality, will delete log directories for jobs no longer present in Qube

@CHANGE: does not require MySQLdb module, but runs faster with it.

==== CL 11424 ====

@FIX: patched crash bug in supervisor (QbSupervisorQueue::_subjobProcReservation())

ZD: 9654

==== CL 11394 ====

@CHANGE: add deprecation warnings to python qb module accessor method, prints once per location

==== CL 11390 ====

@FIX: Global resource collector error occurs on MySQL servers running in STRICT mode: "Field 'total' doesn't have a default value"

==== CL 11319 ====

@FIX: queryIsAdmin() routine is now properly case-insensitive

JIRA: QUBE-128

==== CL 11311 ====

@FIX: case-sensitive user name issue with admin commands

JIRA: QUBE-128

==== CL 11307 ====

@TWEAK: fixed typo "Stoping" -> "Stopping"

Qube 6.5

#####

@RELEASE: 6.5-3

#####

==== CL 12442 ====

@FIX: Fixed a corner-case MySQL permission problem with OSX/Linux supervisor and the qube_readonly user.

Fixed by adding a "GRANT SELECT" with an explicit hostname (fetched via "SELECT @@hostname"), as in:

GRANT SELECT ON *.* TO 'qube_readonly'@'mysqlserverhostname'

JIRA: QUBE-438

==== CL 12358 ====

@FIX: Fixed example python scripts so import of qb module will work in most cases.

==== CL 12347 ====

@FIX: pyCmd* jobtypes report all subsequent frames as failing when a 'regex_error' is matched and a frame is marked as failed

==== CL 12339 ====

@FIX: fixed inaccurate worker host memory reporting on Windows platforms

ZD: 11367

==== CL 12333 ====

@FIX: worker shutdown code (QbWorker::hostShutdown() and sendHostReport()) will now give up a lot quicker when being unable to contact the supervisor, instead of retrying for a long time.

==== CL 12322 ====

@FIX: issue where job instances don't terminate properly when very early kill/interrupt orders come in.

Sometimes interrupts and kills can come in before the worker has a chance to properly complete the launching process of the proxy.exe process and its main thread, causing unexpected behavior, such as a never-dying job instance.

ZD: 11409

==== CL 12315 ====

@FIX: bug in initialization code of the QbJob class that messed up comparisons of jobs when sorting, which, among other things, prevented FIFO/FCFS ordering to be compromised. Now FIFO dispatching behavior should be more closely followed by jobs of equal priority (although not 100% strictly, due to the nature of the multithreaded architecture of the supervisor).

ZD: 11259

@INTERNAL TWEAK: added more debugging code to QbSupervisorQueue module.

==== CL 12311 ====

@FIX: adding in ubuntu support: use bash explicitly rather than sh, specify 'awk' in location found on all OS's

==== CL 12306 ====

@FIX: issue where auto-expanded subjobs (instances) don't inherit the "retrysubjob" value set in the parent job, causing them NOT to auto-retry properly on failure.

ZD: 11292

==== CL 12298 ====

@FIX: Python API routines, such as qb.retrywork(), expecting workID as input would behave erroneously (such as retrying ALL agenda items on ALL jobs) when input a subjobID instead. Vice versa for routines expecting subjobIDs, such as qb.retry().
These

ZD: 11372

==== CL 12295 ====

@NEW: add support for new 'exiting' status

==== CL 12272 ====

@FIX: unreliable behavior when frequently modifying "cpus" of jobs up and down.

ZD: 11288

==== CL 12257 ====

@FIX: bug where auto-expand subjobs are incorrectly auto-retired, and in turn caused them NOT to expand any more.

ZD: 11217

==== CL 12255 ====

@FIX: issue where, if some intermediate job processes crash and die unexpectedly, other job processes may be missed by the cleanup code and left behind as zombies.

ZD: 11236

==== CL 12250 ====

@FIX: WorkerConfigFile makes a better effort at finding the worker config file, previously would save to default location when the file is actually in a non-default location as specified by supervisor_worker_configfile.

==== CL 12237 ====

@FIX: avoid inserting duplicated values into the 'outputPaths' for a frame when retried

==== CL 12232 ====

@FIX: "UnboundLocalError: local variable 'qb' referenced before assignment" - issue experienced by single customer on linux, re-importing qb

module in main() resolves the issue. ZD# 11218

==== CL 12230 ====

@FIX: additional fixes to remedy "retrywork" issue with maya (and possibly other Perl-API based) jobs. See also the previous CL12228

==== CL 12228 ====

@FIX: automatic retry of agenda via "retrywork" not working properly in perl-based backends.

ZD: 11167

==== CL 12226 ====

@FIX: Fix issue where job_cleanup script would fail if run on a supervisor that did not have the MySQLdb python module installed.

==== CL 12219 ====

@FIX: "sre_constants.error: bogus escape (end of line)" - python-based jobs can crash on Windows at startup if path wrapped in QB_CONVERT_PATH() ends with a fwd-slash and has being converted to a back-slash

==== CL 12177 ====

@FIX: Additional changes to support proper Windows privilege enabling, added in CL12176

==== CL 12176 ====

@FIX: Add call to Windows' AdjustTokenPrivileges() to explicitly enabled required privileges before launching job instance (proxy) process

==== CL 12098 ====

@FIX: support negative frame range in QB_* token parsing

==== CL 12082 ====

@FIX: issue where "modify"-ing the "cpus" value of a running job may incorrectly retire more instances than asked for.

This was due to race conditions of supe threads, and in extreme cases, was prematurely retire-ing ALL instances of a job while there are still pending agendas, resulting in the job's instances to be all "complete" but the job itself to become "failed" since there are still pending agendas.

ZD: 10868

==== CL 12065 ====

@INTERNAL TWEAK: added/modified/corrected comments and symbol names for readability

#####

@RELEASE: 6.5-2

#####

==== CL 12016 ====

@FIX: worker and supervisor install do not register for all users on Windows

==== CL 12006 ====

@FIX: ERROR 1146 (42S02) at line 87 in file: './create_job_fact.sql': Table 'pfx_stats.memusage' doesn't exist - swap order of table assignment and creation, some versions of MySQL are error'ing

==== CL 11989 ====

@FIX: worker_drive_map and worker_path_map not correctly saved via "Configure local host", format to match API updatelocalconfig expectations

==== CL 11987 ====

@FIX: localized the _user_duties and _prgp_duties IntHash variables to the queuereject() routine for thread-safety, from being data members of the supervisor class.

ZD: 10342

==== CL 11986 ====

@FIX: added code to appropriately handle timing issues where a command, such as preemption, can be issued multiple times by different threads on the same running subjob, leaving those jobs to be in odd states. One common symptom was seeing the "aberrant report" message in the supelog, and those jobs getting stuck in the "running" state despite all the frames being 100% done.

==== CL 11985 ====

@FIX: converseWorkerWithRetries() and converseSubSupervisorWithRetries() routines were fixed so that they properly return success when there are no

communication errors. These routines were retrying when the server responded with a `py.tag()` of `QB_MESSAGE_ERROR`, which doesn't mean there was a communication error, but rather means that the server encountered some general internal error, causing unwanted retries.

ZD: 10527

==== CL 11982 ====

@FIX: contradictory job log entries saying a failed frame is being reported as complete when a few lines ago it was actually (correctly) reported as failed.

==== CL 11980 ====

@FIX: `QB_CONVERT_PATH()` not getting evaluated when `worker_path_map` is undefined or empty

==== CL 11963 ====

@FIX: catch jobs with package data the cause `_qb.packageStrToDict` to raise an exception

==== CL 11961 ====

@CHANGE: add additional sanity checks to cleanup script, limit number of log directory deletions to a fraction of total jobs in qube, can be overridden by option flag.

==== CL 11931 ====

@CHANGE: create the backfill_fact (supervisor dispatch efficiency) dataWarehouse "12-hour" table every 5 minutes rather than every 15 to keep the chart data more current - full-range table is small enough to support this

==== CL 11915 ====

@FIX: fixed cross-dependency created in CL11893.

JIRA: QUBE-176

==== CL 11908 ====

@CHANGE: changed/added code to set up the following default my.cnf parameters

all OSs:

query_cache_size = 0 # disable the query cache, hit rate is almost 0% due to qube being very write-intensive

thread_cache_size = 16 # acts like supervisor_idle_threads

Linux-only

table_open_cache = 2500 # mysql will cache the file handles necessary to hold this number of tables f/h's

open_files_limit = 50000 # table_open_cache will drive the number of open files, MyISAM needs a max of 2 per table, but MySQL can also open other files past the table_open_cache*2 value - refer to:<http://dev.mysql.com/doc/refman/5.1/en/table-cache.html>

JIRA: QUBE-175

==== CL 11899 ====

@FIX: made the path map translations case-insensitive on OSX and Windows platforms.

@NEW: added 3rd optional parameter to `QbString::replace()`, which specifies the case-sensitivity, which defaults to TRUE.

JIRA: QUBE-177

==== CL 11895 ====

@NEW: exposed the C API routine `qbisadmin()` as `qb.isadmin()` in Python API and `qb::isadmin()` in Perl API.

JIRA: QUBE-174

==== CL 11893 ====

@CHANGE: `qbadmin {s|w} -configuration` now displays both the integer AND string values of all `"*_flags"` (such as `"supervisor_flags"`) parameters for readability

JIRA: QUBE-176

==== CL 11856 ====

@FIX: added code to fix jobs getting stuck in the "dying" state, that can occur due to race conditions.

Dispatched instances of jobs that were requested to be "killed" before they properly finished starting up on the workers were ending up getting stuck in the "dying" state.

ZD: 10369

==== CL 11850 ====

@FIX: C4D AppFinder jobs crash when paths or filenames wrapped in `QB_CONVERT_PATH()` start with a number

==== CL 11829 ====

@FIX: Issue with grid jobs where some instances would start running multiple times on the dispatched host, causing the job to eventually fail.

ZD: 10325

==== CL 11820 ====

@FIX: disable permission check of worker_logpath, as it was creating false-alarms and putting the worker to be in panic mode unnecessarily.

ZD: 5445 5236

BUGZID: 63683

See also CL9234

==== CL 11815 ====

@FIX: on Linux in the /etc/init.d/worker script, we're now allowing a longer timeout (15 seconds) for the worker to shutdown cleanly before forcefully killing (i.e. "kill -9") the processes.

The default short timeout of 3 seconds was not sufficient on many systems for all child worker threads to exit and the main thread to release the running subjobs and report to the supervisor that it's "down".

JIRA: QUBE-90

==== CL 11807 ====

@FIX: added dependency ("requires") on the "expat" package for qube-core RPM packages.

JIRA: QUBE-68

ZD: 8499

==== CL 11801 ====

@FIX: fixed qbjborder() routine so that it respects the queuing algorithm's job-host pair rejection routine, queuerject().

This bug, for example, was causing the routine to return jobs that shouldn't qualify to run on the given host because of the "worker_restrictions" settings of the worker.

ZD: 10231

JIRA: QUBE-158

==== CL 11795 ====

@FIX: issue where the python API qb.convertpath() will cause a bus error (crash) in the caller, if called with no args.

@FIX: issue where the 2-argument invocation of qb.convertpath() was not working, and may cause a bus error. Turned out to be a bug in the internal conversion routine _qb_py_dict_pathmap().

==== CL 11793 ====

@FIX: bug with modifying user and group permissions.

Operations such as adding or deleting a group or users would generate an error message in the supelog, like the following:

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: group permissions modified for user foobar by user USERNAME

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: ERROR: database query error: 127.0.0.1 via TCP/IP - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ') = 'test' AND LOWER(user) = 'foobar' at line 1 (1064)

[Sep 20, 2013 11:39:35] HOSTNAME[25107]: SELECT access FROM grp WHERE valid = 1 AND LOWER(name)) = 'test' AND LOWER(user) = 'foobar'

JIRA: QUBE-157

==== CL 11790 ====

@FIX: fixed inaccurately reported host.processor_speed (CPU frequency in MHz) property on OSX workers.

JIRA: QUBE-153

==== CL 11788 ====

@CHANGE: added "GRANT" statement to "GRANT SELECT ON *.*" to the qube_readonly user on "localhost".

JIRA: QUBE-105

==== CL 11771 ====

@FIX: problem where it was impossible to undefine worker_properties and worker_resources once they were defined in qbwrk.conf or qb.conf, even if the lines were removed from the config files.

JIRA: QUBE-85

ZD: 10227

==== CL 11767 ====

@FIX: Setting "worker_cpus=0" or removing a "worker_cpus=N" line from qbwrk.conf had no effect, and the previous setting would get stuck.

JIRA: QUBE-80, QUBE-112

==== CL 11748 ====

@FIX: helloWorld example jobtype can't create job archive file job.qja below QBDIR/examples, area is read-only, write to a temp directory

==== CL 11740 ====

@CHANGE: remove dependencies between Windows MSI installers, any qube component can be installed or uninstalled independent of the others

==== CL 11733 ====

@FIX: qbtail.py now prints out the help screen if run without arguments.

#####

@RELEASE: 6.5-1

#####

==== CL 11687 ====

@NEW: add support for Adobe's Creative Cloud 'CC' version numbering scheme

==== CL 11685 ====

@FIX: erroneous "timecumulative" of instances (subjobs)

JIRA: QUBE-148

==== CL 11682 ====

@FIX: removed "Configure Qube" menu item from workertray.exe

==== CL 11654 ====

@NEW: added worker_boot_delay to qb.conf.template

==== CL 11631 ====

@CHANGE: modified the new worker_boot_delay (CL11605) to default to 0 seconds on ALL platforms.

ZD: 9386

JIRA: QUBE-118

==== CL 11614 ====

@FIX: The "Administrator" user didn't properly get permissions to manipulate others' jobs, due to case sensitivity.

JIRA: QUBE-142

==== CL 11605 ====

@FIX: added worker_boot_delay to qb.conf, which specifies the number of seconds to artificially delay (i.e. sleep) before the worker boots.

This was added in order to work around a boot-time timing issue with networking/hostname assignment on the Mac OS X platform, where many machines are incorrectly identified as "localhost".

As such, by default, worker_boot_delay is set to 30 seconds on the Mac, and 0 (i.e., no delay) on other platforms.

To override the default, worker_boot_delay *must* be set in the local qb.conf file on each worker-- setting it in qbwrk.conf won't work.

ZD: 9386

JIRA: QUBE-118

==== CL 11484 ====

@FIX: appFinder jobs error out at setup with "raise error, v # invalid expression, bad group name" exception raised by python re module

==== CL 11470 ====

@FIX: random Worker crashes on Windows DU mode-- worker "committing suicide", i.e. killing it's own worker.exe process when removing job processes.

==== CL 11460 ====

@FIX: added code to "generate movie" jobs, to support frame ranges that don't start at 1 with conversions using ffmpeg

@FIX: also added a leading "." to the movie_ext choice strings, which is required.

ZD: 9745

==== CL 11428 ====

@CHANGE: large re-write, not backward-compatible with previous argument spec

@CHANGE: will only delete jobs where all jobs in the job's pgrp completed more than X days ago

@CHANGE: added --removeOrphanedLogs functionality, will delete log directories for jobs no longer present in Qube

@CHANGE: does not require MySQLdb module, but runs faster with it.

==== CL 11424 ====

@FIX: patched crash bug in supervisor (QbSupervisorQueue::_subjobProcReservation())

ZD: 9654

==== CL 11394 ====

@CHANGE: add deprecation warnings to python qb module accessor method, prints once per location

==== CL 11390 ====

@FIX: Global resource collector error occurs on MySQL servers running in STRICT mode: "Field 'total' doesn't have a default value"

#####

@RELEASE: 6.5-0

#####

==== CL 11260 ====

@NEW: add Python/Perl API access to the new "timecumulative" subjob and work data

==== CL 11257 ====

@NEW: add support to the data warehouse for the new "cumulative time" instance and work columns

==== CL 11254 ====

@NEW: added ReadMe.rtf for the osx supervisor installer, so that the person installing the supervisor is made aware of the upgrade potentially taking a long time to complete.

==== CL 11247 ====

@NEW: add tracking of cumulative time spent in the "running" state for subjobs (instances) and work (agenda items), stored as an additional DB field in the Nsubjob and Nwork tables.

These values are computed by taking the elapsed number of seconds since the last start time of the instance/work, and then multiplying it by the number of actual job slots occupied by the instance, as dictated by the "host.processors" reservation.

JIRA: QUBE-124, QUBE-125

==== CL 11240 ====

@NEW: add support for job tags to the data warehouse schema

==== CL 11237 ====

@NEW: qb.query module's jobinfo function now accepts a "where" argument that will pass through a mysql "where" statement (without the "where" word). Example: where="name like 'foo' OR user like 'foo'"

==== CL 11219 ====

@FIX: issue with automount in desktop user mode.

@INTERNAL: also cleaned up some Win32 automount code.

ZD: 9434

==== CL 11217 ====

@CHANGE: Python API change: all python classes can now be constructed without data. In other words, one can create empty objects.

==== CL 11216 ====

@NEW: add support for real-time log parsing and progress percentages to job instances for non-agenda-based jobtypes, currently only supported by python-based jobtypes

==== CL 11208 ====

@NEW: label text for worker parameter widget changes colour to indicate a value which will be saved

@FIX: properly support worker_path_map and worker_drive_map in new qbwrk.conf configuration dialog

@FIX: properly indicate when the value for a given parameter varies between workers, even if it's not defined for a worker but defined for others

@FIX: support greater than 5 mapping definitions for worker_path_map and worker_drive_map

==== CL 11201 ====

@NEW: add DB conversion scripts (to add new subjob table columns introduced in 6.5, DB version 32) that run when the rpm/pkg/msi supe installer runs, to provide forward-compatibility for old (pre-6.5) job/subjob data

JIRA: QUBE-119

==== CL 11200 ====

@TWEAK: Minor Python API change: You can now create an empty qb.Job()

==== CL 11190 ====

@CHANGE: added "enable_windows_job_object" flag, and deprecated "disable_windows_job_object"

The "disable_windows_job_object" flag is silently ignored. Windows Job

Objects are always disabled now, unless the job explicitly specifies otherwise with the new "enable_windows_job_object" flag.

JIRA: QUBE-117

==== CL 11185 ====

@CHANGE: modified user name and group name authentications to be case-insensitive

JIRA: QUBE-98

==== CL 11183 ====

@NEW: add supervisor_default_hostorder parameter to qb.conf

JIRA: QUBE-113

==== CL 11178 ====

@CHANGE: Added "post" to the default callback language list, supervisor_language_flags

==== CL 11175 ====

@NEW: advanced worker resource reservations, including N-M, N+M and N* reservation specifiers, and live tracking of resource "allocations" and "slots" (actually allocated host.processor value) of running job instances.

JIRA: QUBE-91

==== CL 11174 ====

@NEW: add new worker DB table schema, QbTableVersion8

==== CL 11155 ====

@NEW: Add QB_JOBSLOTS and QB_ALLOCATIONS environment variables to be set, indicating the initial jobslot allocation and the more general initial resource allocations, respectively, when jobs execute.

==== CL 11127 ====

@FIX: editing supervisor config with WranglerView->Admin->Configure causes the 'submit_job' privilege to be removed from supervisor_default_security

@CHANGE: Admin->Display Config (local) changed to Admin->Display Running Config, shows supervisor and/or worker running config if these services are running locally

@CHANGE: Admin->Configure (Local) changed to Admin->Configure Local Host

@INTERNAL: add all supervisor and worker flag values to qb.utils.flags, now used directly by configuration dialog, instead of the config dlg items being order-dependent. Allows for sorting configuration dialog items alphabetically for ease of use.

@NEW: Admin->Configure Local Host now creates a timestamped backup of the qb.conf file in the same location as the original

@NEW: Admin->Configure Local Host is now disabled on Windows and Linux if not invoked by root (linux) or Admin-equivalent (Windows)

@COSMETIC: File->Install AppUI menu items now sorted alphabetically

==== CL 11122 ====

@NEW: New utility qbtail.py in \$QBDIR/utils. This is a *nix tail-like utility implemented in Python that runs on OS X, Linux, and Windows.

==== CL 11096 ====

@NEW: Updated and cleaned up all Python API examples in QBDIR/examples/python

==== CL 11088 ====

@INTEG: dev-supervisor-additional-job-params>main,CL11087

@NEW: added prod_{show,shot,seq,client,dept,custom[1-5]} fields to the job object.

The C++, Python, and Perl APIs have been updated.

The qbsub, qbjobs, and qbmodify commands and their online help text have been updated.

JIRA: QUBE-79

==== CL 11024 ====

@NEW: supervisor reread qb.conf file feature.

The qbadmin command has been updated with a "-reread" option, to be called as in "qbadmin s -reread" to instruct the supervisor to reread its qb.conf file and update the dynamically modifiable parameters, which are as of this writing:

- * qb_domain
- * supervisor_default_group_security
- * supervisor_default_p_agenda_priority
- * supervisor_default_pgrp_subjob_limit
- * supervisor_default_priority
- * supervisor_default_security
- * supervisor_default_user_subjob_limit
- * supervisor_flags
- * supervisor_global_resources
- * supervisor_highest_user_priority

- * supervisor_job_flags
- * supervisor_max_priority
- * supervisor_p_agenda_max
- * supervisor_pgrp_subjob_limits
- * supervisor_user_subjob_limits
- * supervisor_verbosity
- * supervisor_worker_configfile

(Note: DB schema change was involved, and QbTableVersion32 was added)

JIRA: QUBE-92

==== CL 10972 ====

@CHANGE: modified qbworkerpathmap() to return the localhost's worker_path_map when called outside of a jobtype back-end environment.

JIRA: QUBE-95

==== CL 10953 ====

@FIX: remove digit/number from worker's journal file name (worker6.jnl -> worker.jnl)

==== CL 10946 ====

@TWEAK: error code now prints when QbTrackOSX in trackAssignment() encounters an error.

==== CL 10945 ====

@FIX: fix yet another issue with previous CL, concerning graceful worker shutdown.

Also switched a couple of calls to qbvcout to qbvout so that more useful info such as timestamp and pid print.

==== CL 10943 ====

@INTERNAL TWEAK: added/corrected comments, and removed unneeded #ifdef/endif macro, while working on previous CL

==== CL 10942 ====

@CHANGE: modified worker shutdown code so that it immediately returns all running subjobs to the supe, and report a status of "down", so the supervisor marks it "down" promptly.

JIRA: QUBE-90

==== CL 10927 ====

@FIX: changed the initial state of a newly dispatched instance on a worker to "running" instead of "pending", so that early calls to qbjobobject() in the back-end code will return "running" as the job's status

JIRA: QUBE-45

==== CL 10827 ====

@NEW: a new interface for configuring multiple workers and writing out the qbwrk.conf

@FIX: add "convert_path" flag to client_ and supervisor_job_flags control

@FIX: default value in config UI for worker_max_threads was 8, now 256

@FIX: "Configure on Supervisor" worker menu item is only enabled if user has the qube "admin" privilege AND is on the supervisor

@COSMETIC: all checkbox lists from "Choices" buttons are now sorted alphabetically

==== CL 10667 ====

@CHANGE: Python API-level change: QbObject is now simply a python dictionary. It no longer re-implements any functions

@CHANGE: Python API-level change: qb.updatelocalconfig now uses the subprocess module rather than the deprecated popen2.Popen4.

==== CL 10589 ====

@FIX: job list not updating when switching supervisors, always show jobs from the default supervisor.

==== CL 10255 ====

@INTEG: rel-6.4 -> main

@NEW: pyCmdline and pyCmdrange do run-time path translation

==== CL 10061 ====

@CHANGE: index all datawarehouse fact tables on time_sk column, since it's so frequently accessed.

==== CL 10056 ====

@FIX: PFX_CREATE_DATASUBSET_TABLE doesn't use an indexed column for the WHERE clause, now does an INNER JOIN to the time dimension table

==== CL 10037 ====

@FIX: allow qube-core to be repaired if other qube products are already installed

==== CL 9730 ====

@TWEAK: modified so that worker name and IP print when job is accepted by worker, in assignJob()

Qube 6.4

```
#####  
@RELEASE: 6.4-5  
#####
```

==== CL 11108 ====

@FIX: supe's built-in perl library's C++ host object to Perl hash conversion routine to properly include "properties", "stats", "reason", "locks", "flags", "flagsstring", "groups", "description", "jobtypes", "address", "macaddress", "lastupdate"

@FIX: typo in Perl API's _qb_host_hash() routine when converting the "description" field.

==== CL 11093 ====

@NEW: add various helper functions to qb.utils; addToSysPath(), getModulePath(), pyVerAsFloat(), formatExc()

==== CL 11067 ====

@FIX: timing issue where a subjob of an agenda-based job can be incorrectly left in the "blocked" or "pending" state even though there are no more agenda items to be processed.

@INTERNAL: Checker code was added to the statusJob() routine to force the status to "complete" of such jobs.

ZD: 9190

==== CL 11066 ====

@INTEG:main>rel-6.4,rel-6.3,CL11024, CL11056, CL11057

This is a partial integration of CL11024,11056,andCL11057. Namely, the "const"-ness fix in the QbDatabase* classes are being integrated into rel-6.3 and rel-6.4 so they will compile cleanly. Also, the change in the logging behavior (so that MySQL logs are timestamped) is integrated.

==== CL 11062 ====

@FIX: fixed unreliable "modify" behavior. Multiple modifies (for example, up then down) were behaving oddly.

@CHANGE: added code to automatically retire pending/blocked/running jobs when "modify" reduces the "cpus" ("instances") count.

ZD: 9205

@FIX: fixed a subtle off-by-one error in auto-retire code in assignJob()

==== CL 11058 ====

@FIX: patched a timing issue where the requestWork() handler can sometimes put a running subjob back to "pending" (because it's marked to be passively preempted) even if there are no more agenda items left to process.

ZD: 9132

==== CL 11054 ====

@CHANGE: made all error messages from the QbDatabaseMySQL class prints with a timestamp.

==== CL 11016 ====

@FIX: fixed return data type of qb.submit() to be a list of job objects

ZD: 9314

==== CL 11008 ====

@FIX: issue where modifying a job to reduce the number of instances can sometimes incorrectly retire ALL instances.

==== CL 10984 ====

@FIX: control-characters in C4D Windows paths can break path translation, get evaluated as tabs/newLines/etc. This is due to C4D needing to be run via "start" instead of "cmd.exe /C"

==== CL 10963 ====

@FIX: random worker crash issue on Windows

ZD: 8620

==== CL 10934 ====

@FIX: suppress printing of "Malformed env in parsing" and environment listing when environment values are other than simple strings and "Query SQL" is enabled in the WranglerView prefs

```
#####
```

@RELEASE: 6.4-4

#####

==== CL 10894 ====

@FIX: Updated qb.conf.wintemp (Windows template file for qb.conf) to be in sync with the Unix template.

Also added proper data paths for Windows Vista 7 and up, in the commented-out default values.

JIRA: QUBE-74

==== CL 10875 ====

@FIX: issue with stdout/err logs getting truncated and duplicate status being logged to .hst file when qbreportjob() is used to report intermediate status by a running instance.

JIRA: QUBE-46

==== CL 10867 ====

@FIX: data warehouse database creation failing on CentOS 6.2; mysql client is installed in /usr/bin instead of /bin, and must provide full paths to bash "." source statements

==== CL 10858 ====

@INTERNAL: qblock/qbunlock source consolidation on Windows

==== CL 10857 ====

@INTERNAL: consolidating qblock and qbunlock source files.

==== CL 10856 ====

@FIX: minor regex issue with previous check-in (CL10855)

==== CL 10855 ====

@FIX: qbhash command (on Windows only) allowed additional options when run as "qbhash.exe"

This was due to it sharing the same code as qblogin, and an internal regex not considering the .exe extension.

==== CL 10841 ====

@FIX: fixed issue where agenda item commands such as "retrywork" would incorrectly be applied to unspecified/undesired agenda items, if the list of agenda items contains items from more than 1 parent job (e.g. "1234:1 1235:1")

For example, "qbretry 1234:1 1235:1" would retry every work item in job 1235, despite the specification being just item 1.

JIRA: QUBE-61

==== CL 10839 ====

@FIX: minor logging fix: "resetting start/complete time of work" now prints the work 'name' instead of 'id' for consistency and readability.

==== CL 10833 ====

@FIX: supervisor msi installation fails during InnoDB cleanup operation, aborts the supervisor installation

==== CL 10830 ====

@FIX: python api qb.submit() fails silently when a job label is not unique within the pgrp

Now raises a ValueError exception, with an error msg when qb.submit() fails.

JIRA: QUBE-32

==== CL 10802 ====

@FIX: suppress warning about missing stderr job logs if stderr merged into stdout in job submission

==== CL 10798 ====

@CHANGE: pyCmd* jobtypes now properly mimic cmdline/cmdrange behavior: apply path conversion to entire cmdline string if convert_path flag is set, otherwise only apply path conversion to strings enclosed in a QB_CONVERT_PATH token block

@CHANGE: also support mixed use of QB_CONVERT_PATH tokens and convert_path job flag, apply translation to tokens first, then the rest of the cmdline

==== CL 10784 ====

@FIX: issue where grid jobs are doubly booted on the allocated nodes in certain cases.

ZD: 8686

==== CL 10744 ====

@NEW: add support for per-OS environment variables, allows for different envVar values depending on run-time OS. Currently only supported by pyCmdline, pyCmdrange, and appFinder jobtypes. Passed in as job['package']['env_runTimeOS']['Windows' | 'Linux' | 'Darwin'], keyed off platform.system()

==== CL 10724 ====

@FIX: possible crashes due to timing issue (between queue.listReady() and queue.getByld()) in startResources()

ZD: 8566

==== CL 10723 ====

@FIX: memory leak in startHost().

@FIX: possible crashes due to timing issue (between queue.listReady() and queue.getByld())

ZD: 8566

==== CL 10715 ====

@CHANGE: decrease default log rotation size from 256MB to 100MB on Windows and OS X, can be overridden by providing the '-s <size>' argument to logrotate.py in the Windows scheduled task or the OS X LaunchDaemon plist

==== CL 10705 ====

@FIX: fixed issue with -p_agenda option incorrectly picking frames.

==== CL 10695 ====

@FIX: Fixed issue with instances being dispatched to multiple workers when jobs were qbmodify-ed their "cpus" down and then up.

When the "cpus" parameter, i.e. the instance count, was qbmodify-ed down and then up, some instances would end up being dispatched and running on multiple workers at the same time. This was due to the fact that

Until now, when a job's cpus count is reduced, instances of higher ID numbers were always chosen to retire (i.e., if a 5-instance job was reduce to 3, then instances 3 and 4 were retired). Now, instead, the first instances that request a "requestwork" are retired.

Also, when a job's cpus count is increased, the supe will first revitalize any instances that are already in the "done" state, and then add more instances to the job if necessary. For example, say a 5-instance job was reduced to 3 instances, and instance 1 and 2 were retired in response (0,3,4 are running). If, later, the job was modified again to increase the instance count to 7, instances 1 and 2 are revitalized (i.e. moved back to "pending") AND 2 new instances, 5 and 6, are generated.

ZD: 8542

==== CL 10692 ====

@CHANGE: added more useful msg to print in workerVerifyAssignment()

==== CL 10685 ====

@FIX: fixed examples/cpp/ .sln and .vcproj files to build for x64 and under VS 2005

==== CL 10682 ====

@CHANGE: added a supervisor_preempt_policy of "mixed", to support mixed-mode preemption with custom algorithms (and potentially with built-in algorithms too, in the future).

Setting the preemption mode to "mixed" allows custom algorithms to aggressively preempt a job that's already been marked to be passively preempted.

ZD: 8556

==== CL 10642 ====

@NEW: move AppFinder jobs to their own jobtype

==== CL 10641 ====

@NEW: add QB_CONVERT_PATH() tokens to paths in simpleCmds to support runtime path conversion using the conventional qb.convertpath()

@NEW: imports new qb.utils module

@FIX: pattern matches in logs (output paths, highlights, etc) being stored multiple times

==== CL 10640 ====

@FIX: characters in application path string are being interpreted as escaped ctrl-characters

==== CL 10633 ====

@NEW: add QbTableVersion31

==== CL 10608 ====

@INTERNAL: changed MySQL MEMORY table creations to read "ENGINE=MEMORY" instead of "TYPE=MEMORY" which is obsolete as of MySQL 5.5

==== CL 10604 ====

@CHANGE: all obsolete "HEAP" type MySQL tables to the new "MEMORY" type, to conform to MySQL spec change as of version 4.1 (HEAP backward compatibility removed in 5.5)

@INTERNAL: added QbTableVersion31.cpp

@INTERNAL: upped QbVersion version to 6.4.3

BUGZID: 63769

==== CL 10594 ====

@FIX: issue where the automount flag was always set for jobs if client_job_flags was set to the empty string in qb.conf

==== CL 10589 ====

@FIX: job list not updating when switching supervisors, always show jobs from the default supervisor.

==== CL 9730 ====

@TWEAK: modified so that worker name and IP print when job is accepted by worker, in assignJob()

@RELEASE: 6.4-3
#####

(version skipped - never released)

@RELEASE: 6.4-2a
#####

==== CL 10591 ====

@FIX: fixed issue where the worker rejects jobs with the auto_mount flag turned on when run in desktop user mode and worker_cpus != 1 (which automatically turns off auto_mount in worker_flags)

The auto_mount settings of the job/worker should be irrelevant for workers running in desktop user mode.

@RELEASE: 6.4.2
#####

==== CL 10543 ====

@FIX: issue with worker_path_map not working when defined in qbrk.conf and containing backslashes.

==== CL 10537 ====

@FIX: issue where qbconvertpath() can return an empty string when worker_path_map is undefined.

@RELEASE: 6.4.1
#####

==== CL 10514 ====

@FIX: another patch for out-of-order issue. Fixed unexpected short-circuit evaluation that was happening in the startResources() routine

==== CL 10513 ====

@FIX: another patch for out-of-order issue. Fixed unexpected short-circuit evaluation that was happening in the startHost() routine

==== CL 10512 ====

@INTERNAL: QbJob object's _subjobswaiting data was not being initialized or copied correctly, causing some job comparisons based on subjobs waiting counts to unexpectedly fail.

==== CL 10504 ====

@INTERNAL: added more log output for debugging builds, added more comments while working on out-of-order issue.

ZD: 8198

==== CL 10477 ====

@FIX: Another out-of-order fix. Jobs at the same numerical and cluster priority should dispatch in the correct FIFO order now.

The FIFO enforcing should work most of the time, but there still will be occasional out-of-order behavior, due to the multi-threaded nature of the supervisor. ("qbshove"-ing the older job should correct it, when it's seen)

ZD: 8198

==== CL 10462 ====

@FIX: yet yet another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.
See also CL10440 10452

ZD: 8198

==== CL 10461 ====

@CHANGE: modified/compacted the multi-line "found a duty to replace" logging to be a single line.

==== CL 10452 ====

@FIX: yet another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.
See also CL10440

ZD: 8198

==== CL 10441 ====

@FIX: killing an already finished (complete, failed, killed) job leaves the job in the "dying" state.

==== CL 10440 ====

@FIX: another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.

ZD: 8198

==== CL 10429 ====

@FIX: out-of-order job dispatching issue with jobs using the "+" sign with the "host.processors" reservations.

ZD: 8198 8261 8229 8233 8228

==== CL 10389 ====

@NEW: add new appFinder submission for C4D

==== CL 10323 ====

@NEW: add support to pyCmd* jobtypes for new "auto-pathing" feature; can now send jobs to a mixed set of workers and find the 3rd-party executable on all OS's, not pre-defined in the job's package

==== CL 10271 ====

@CHANGE: desktop user mode worker to only allow automount when "worker_cpus = 1" is set explicitly.

==== CL 10264 ====

@NEW: add automount support for desktop user mode on Windows

@CHANGE: db table change (additional column to the assignment table) required-- adding QbTableVersion7 definition.

@FIX: unmounting of "subst" style local mounts was broken

@INTERNAL: added a bunch of comments, and renamed some methods in the QbMission class, for readability.

==== CL 10254 ====

@NEW: pyCmdline and pyCmdrange do run-time path translation

==== CL 10233 ====

@FIX: added qb::workerconfig() that was missing to the Perl API

==== CL 10228 ====

@FIX: missing "bin/qbhash" command on Linux

==== CL 10223 ====

@FIX: examples in the code to reflect previous change to the command line options/arg

==== CL 10216 ====

@NEW: Job cleanup script in utils directory. This script is designed to be run by a user or by a user-created scheduled task.

==== CL 10191 ====

@FIX: removed unneeded "install_worker" and "uninstall_worker" scripts from being installed on Mac OSX

==== CL 10189 ====

@FIX: timing issue where some worker resources (host.xyz) would disappear after the worker received a remote config.

@FIX: issue where supervisor tries to dispatch a subjob to a worker with insufficient resources (reduced the likeliness of that from happening)

@FIX: the above 2 fixes combined should now prevent some of the out-of-priority-order dispatch issues, especially in environments where worker resources are deployed.

ZD: 7885

==== CL 10149 ====

@CHANGE: modified so the worker_path_map mapping definition order is preserved when it is applied to paths via convertpath()

==== CL 10144 ====

@FIX: bug with handling lone backslash in the worker_path_map

@CHANGE: modifying QbConfig class to maintain order of option (config parameter) addition

==== CL 10125 ====

@NEW: add automatic runtime path conversion to cmdline and cmdrange jobtypes

@NEW: jobs may have the "convert_path" flag set to tell the jobtype to do runtime path conversion.

@NEW: qbsub now has a "-convertpath" option to set the flag.

@NEW: qubegui simpleCmd interface has a new "convert path" checkbox

==== CL 10118 ====

@FIX: fixed issue where agenda timeouts don't work properly on the first agenda item processed by a subjob, on Unix (Linux/OSX) workers

==== CL 10117 ====

@FIX: fixed issue where agenda items that fail because of timeout don't get automatically retried via retrywork

ZD: 7763

==== CL 10097 ====

@NEW: add Mac OS X 10.8 Mountain Lion support

==== CL 10095 ====

@FIX: fixed newly introduced issue with errors reading licenses in dev/main branch supe

==== CL 10074 ====

@INTEG: main -> rel-6.4

@FIX: data warehouse installation/upgrade scripts on linux/OSX now search /etc/qb.conf for database_user/_password/_port/_host values in order to support non-default values for these parameters

==== CL 10072 ====

@NEW: add activeperl 5.16 support for Windows

==== CL 10068 ====

@NEW: Add doc on QB_CONVERT_PATH(srcpath) in Use.doc and qbsub's online help

==== CL 10067 ====

@NEW: Add documentation on worker_path_map config parameter and the qbconvertpath() API routine.

==== CL 10062 ====

@FIX: fixed parsing code in QbConfigFile.cpp so that the "name" part of a name-value pair can contain special chars if double-quoted.

==== CL 10049 ====

@INTEG: main -> rel-6.4

@FIX: reduce the number of times qb.supervisorconfig() and qb.getusers() are called during GUI startup and normal operation, pre-populate the qbCache with this data at startup

==== CL 10048 ====

@FIX: reduce the number of times qb.supervisorconfig() and qb.getusers() are called during GUI startup and normal operation, pre-populate the qbCache with this data at startup

==== CL 10025 ====

@FIX: data warehouse installation/upgrade scripts on linux/OSX now search /etc/qb.conf for database_user/_password/_port/_host values in order to support non-default values for these parameters

==== CL 10022 ====

@FIX: modified the worker to only report to the supe of its host status when subjobs are completely done and removed, and NOT when they are only marked/scheduled for removal.

This was causing jobs to sometimes run out-of-order, especially when there are many subjobs to each job (such as one subjob per frame), since that situation tends to increase the chance of the supervisor dispatching the same subjob to the same worker. The subjob will be dispatched to the same worker, but rejected since the worker thinks it's a duplicate assignment of a subjob that's being removed (and consequently a lower priority job will get the worker's slot, causing out-of-order job execution)

ZD: 7601

```
#####  
@RELEASE: 6.4.0  
#####
```

==== CL 9973 ====

@FIX: bug where UNC paths with backslashes won't work in the new worker_path_map

@INTERNAL: Note: Backslashes are now NOT treated as special chars in QbConfigFile's tokenize() routine (called from parse())

==== CL 9966 ====

@NEW: pyCmdline - a python-based implementation of cmdline jobtype

==== CL 9963 ====

@FIX: add launchCondition so that worker and supervisor will not install if core is not present

@NEW: write a registry key upon installation in order to provide dependency checking for core removal (core will not uninstall if worker or supervisor is installed)

==== CL 9959 ====

@NEW: adding back-end run-time path conversion feature, and exposing in perl, python, and C++ APIs (qbconvertpath())

==== CL 9953 ====

@FIX: fixed config file (qb.conf) parsing code so that it properly parses the worker_path_map

Note: old code was corrupting qb.conf when upgrade_config tool was run.

==== CL 9937 ====

@NEW: houdini loadOnce jobtype finds the appropriate houdini installation at runtime, based off HFS and optionally pkg['houdiniVersion'], user no longer has to guess at python path on the remote worker

@NEW: add versionPicker controls to QubeGUI Houdini submission UI

@NEW: new multi-line syntax for application paths in the job.conf file

@NEW: added scanConfForPaths to backend utils module

==== CL 9930 ====

@NEW: added qbworkerpathmap() to the C++ API and qb.workerpathmap() to the python API.

The worker_path_map in qb.conf (or qbwrk.conf) must be defined like:

```
worker_path_map = {  
[direct]  
H: = /home  
X: = /proj/x  
}
```

Note, in particular, the "[direct]" keyword. That MUST be present.

qb.workerpathmap() called in the python backend will return a nested dict of the format:

```
{'directmap': {'X:': '/proj/x', 'H:': '/home'}}
```

@INTERNAL: fixed bugs in the config-file reader code, added a bunch of comments

==== CL 9918 ====

@UPDATE: update Use.doc with table of all job flags and their descriptions, including info on the new migrate_on_frame_retry job flag

==== CL 9915 ====

@NEW: added a new job flag "migrate_on_frame_retry", which, if set, forces a subjob to migrate to another worker if it fails a frame, and the frame is set to automatically retry (via retrywork).

==== CL 9909 ====

@FIX: fixed issue that was causing jobs to NOT be considered for dispatch immediately at submission.

Bug was introduced while attempting to fix a memory leak bug, in CL9592

==== CL 9903 ====

@FIX: better message from worker when it rejects a dispatched subjob because it's a duplicate (being preempted or migrated on the same worker)

==== CL 9893 ====

@NEW: add example qb.conf files for various-sized farms

@NEW: add example qbwrk.conf to the build

==== CL 9891 ====

@FIX: _highest_priority() routine to disregard priorities that are non-positive.

==== CL 9886 ====

@UPDATE: admin doc with info on supervisor_highest_user_priority

==== CL 9882 ====

@FIX: fixed pathmap bug where the object/data wasn't being properly transmitted over the network at all.

@CHANGE: also uncommented the line that prints out the pathmap to the workerlog on worker boot.

==== CL 9865 ====

@NEW: Added support for supervisor_highest_user_priority to the GUI's "Local Configuration" dialog.

@NEW: Added supervisor_highest_user_priority to the qb.conf.template file.

@CHANGE: Also modified the description of supervisor_max_priority in qb.conf.template to avoid confusion.

==== CL 9864 ====

@NEW: added qb.conf setting "supervisor_highest_user_priority", which sets the highest priority (i.e., smallest numerical value) at which an ordinary (non-admin) user can submit/modify jobs.

Users must be qube admin to be able to submit/modify at higher priority than this value.
It's default value is 1.

BUGZID: 63717

==== CL 9838 ====

@CHANGE: upped the default value for supervisor_max_threads to 100, and worker_max_threads to 32

==== CL 9837 ====

@CHANGE: update the qb.conf templates, supervisor_max_threads=96, leave it uncommented until such time as this matches the supervisor's default behavior

==== CL 9788 ====

@TWEAK: improved log message when worker goes into panic because of the lack of sufficient permissions

==== CL 9785 ====

@FIX: worker issue where desktop worker would randomly crash.

ZD: 6778

==== CL 9736 ====

@NEW: add support for MySQL passwords to qb.query.mysqlConnect

==== CL 9711 ====

@NEW: add Admin->Database Check/Repair functionality to the GUI

@TWEAK: add ability to print to logPane in realtime for long-running processes, no need to wait until operation is finished

@FIX: bugfix for Admin->Ping Supervisor raising KeyError when supervisor is down

==== CL 9698 ====

@FIX: fixed false-negative warning message pertaining to "select() in checkpoint()" seen in supelog.

Examples of these messages:

select() in checkpoint(): Operation timed out

select() in checkpoint(): Interrupted system call

==== CL 9694 ====

@FIX: fixed issue with the supe threads getting tied up on "subjob X seems to be already assigned" message.

On a farm with busy workers, the time between the supe dispatching a sub job to the worker via assignJob() and the worker reporting that the "subjob is running" can be several seconds to sometimes even several minutes, which was causing many supe threads to attempt dispatching the same subjob over and over. All of those threads end up hitting the "subjob X seems to be already assigned... retrying" message, and get tied up for 3 seconds while they retry.

BUGZID:

ZD: 6760 7125

==== CL 9689 ====

@FIX: fixed bug in clustering algorithm where it incorrectly gave more weight to a job when the only difference was the last letter in the cluster specification.

For example, if:

host cluster: /3D/projA
job1 cluster: /3D/projB
job2 cluster: /3D

job1 was getting more weight than job2, which is incorrect.

BUGZID: 63740
ZD: 7043

==== CL 9687 ====

@INTEG: rel-6.3 -> main CL 9686

@FIX: using deprecated "waitfor" attribute with Python api causes qb.submit() to raise a KeyError

@FIX: properly convert "waitfor" value (jobid integer) to proper "dependency" string of "link-done-job-<id>"

==== CL 9686 ====

@FIX: using deprecated "waitfor" attribute with Python api causes qb.submit() to raise a KeyError

@FIX: properly convert "waitfor" value (jobid integer) to proper "dependency" string of "link-done-job-<id>"

==== CL 9678 ====

@NEW: provide a "Studio Overrides Prefs" in the QubeGUI which will allow mandated studio-wide preferences, will override userPrefs, which already override the "Studio Defaults Prefs". Added support for --studioprefs cmdline option and QUBEGUI_STUDIOPREFS environment variable.

==== CL 9677 ====

@INTEG: rel-6.3->main CL 9676

@FIX: update documentation and GUI help text to show correct "||" syntax for job restrictions list.

==== CL 9676 ====

@FIX: update documentation and GUI help text to show correct "||" syntax for job restrictions list.

==== CL 9664 ====

@CHANGE: specify unix_socket when connecting to MySQL server on localhost on non-Windows platforms

==== CL 9663 ====

@INTEG: rel-6.3 -> main CL 9662

@FIX: supervisor install was failing postflight scripts on OSX Server, explicitly set the mysql socket to /tmp/mysql.sock in /etc/my.cnf and /etc/qb.conf to avoid conflicting with the factory-installed default of /var/lib/mysql/mysql.sock

==== CL 9662 ====

@FIX: supervisor was failing postflight upgrade scripts on OSX Server, explicitly set the mysql socket to /tmp/mysql.sock in /etc/my.cnf and /etc/qb.conf to avoid conflicting with the factory-installed default of /var/lib/mysql/mysql.sock

==== CL 9615 ====

@FIX: Added code to properly log frames (to supelog and job log) when they go back to "pending" after the processing subjob/worker is found dead.

@FIX: Added code in the supervisor to retry a failed worker connection after a random 5-10 sec sleep/delay, to alleviate network hiccups during network commands (kill, preempt, etc. of running subjobs).

ZD: 6760

==== CL 9614 ====

@INTERNAL: fixed a small cosmetic bug introduced in CL 9606

==== CL 9607 ====

@INTERNAL: added converseWorkerWithRetries() and also fixed small bug in the retry loop of converseSubSupervisorWithRetries()

==== CL 9592 ====

Fixed code that was causing memory leaks when supervisor threads handled job submissions.

==== CL 9585 ====

@FIX: issue where some jobs get stuck in the "dying" state when attempted to be killed

ZD: 6616

==== CL 9578 ====

@NEW: add another python example script which shows a 'block until' type of callback; a job can be submitted to run at a certain time of day, if the TOD is in the past, it's assumed to be tomorrow

==== CL 9570 ====

@FIX: improvements to the handling of GET_LOCK (aka "reserveJob()") timeout situations.

ZD: 6617

==== CL 9549 ====

@FIX: qbwrk.conf files that had any commented-lines before the first valid template was encountered would cause an exception to be raised, QubeGUI->worker->RMB->Configure would fail silently

==== CL 9535 ====

@NEW: add submit-agenda-timeout-job.py example python script, to demonstrate submission of a job with frame-level timeouts.

ZD: 6099

==== CL 9530 ====

@FIX: Submitting paths to shotgun no longer depends on the visibility of output paths to the supervisor.

@FIX: Shotgun submission script fails gracefully & logs a reason as to why it can't generate a thumbnail when thumbnail creation fails.

==== CL 9523 ====

@FIX: fixed issue where the supervisor fails to correctly track the host assignment for subjobs.

Symptom for this included seeing in the supelog, messages like "statusJob(): aberrant report from worker...", then followed by "subjob[xxxx] is assigned to worker[]" with mac address[00:00:00:00:00:00].

These subjobs would then be in the "running" state, but not assigned to a worker.

==== CL 9522 ====

@FIX: removed code that skipped code that made local decision on the supe to test for resource reservations, for jobs with host.processors set to > 1, delegating the decision-making to the workers and resulting in more network traffic and latency.

ZD: 6141

==== CL 9507 ====

@FIX: added more robust code that talks to the SMTP server when sending out email, to support some email servers with non-standard response behavior.

ZD: 6209

==== CL 9504 ====

@FIX: catch case where sg_path_to_frames is part of the Shotgun versionName, but the job has no outputPaths for the first frame; fallback to naming the version "job id: 123 jobName: ..."

==== CL 9500 ====

@FIX: Windows Vista/7/2008-R2 installer - don't error out when installing the worker or supervisor as an Admin-equivalent account during creation of scheduled tasks. Properly remove scheduled tasks during uninstall.

==== CL 9496 ====

@FIX: catch case when inserting in a new cluster into cluster_dim when more than 1 worker exists in the new cluster; occurs during run of regular_slotcount.sql, doesn't prevent new record from being added, just generates line noise and error emails from cron...

==== CL 9494 ====

@CHANGE: make explanation of "+ | *" in job/host restrictions less ambiguous

==== CL 9484 ====

@FIX: calculate cpu-seconds for agenda-based jobs by summing up work times, not subjobs. Better support for resetting of the start times for retried work.

==== CL 9467 ====

@NEW: add a random offset to the startup so that all workers don't report at the same time if they've started up at the same time.

@CHANGE: don't retrieve job name, it's extraneous and not reported; cuts down the query count by one.

@CHANGE: set workname for subjob to job.subid, not subid; easier to detect case where an agenda-based job falsely reports not having an agenda, so subjob id won't conflict with a frame number

==== CL 9463 ====

@FIX: don't report memory usage in the case where MySQL fails to return a valid agenda name, usually caused by timeouts or maxed out connections.

==== CL 9461 ====

@CHANGE: removing from VS solution: qbdeletevariable qbgetvariable qbsetvariable qbworkervar

==== CL 9460 ====

@CHANGE: removing legacy commands from sbin-- qbworkervar, qbdeletevariable, qbgetvariable, qbsetvariable

==== CL 9459 ====

@NEW: added ip address column ("address") to the banned DB table

@NEW: enabled "qbadmin w -unremove <worker>" to work with hostname and IP address, in addition to the mac address.

BUGZID: 63703

==== CL 9458 ====

@NEW: adding QbTableVersion30.cpp to upgrade_supervisor.vcproj

New DB table schema definition file for rel-6.4

See also the previous changelist, CL9451

==== CL 9456 ====

@FIX: moved the location of QbTableVersion29.cpp (rel-6.3) inside the upgrade_supervisor.vcproj file from the incorrect "Resources Files" folder to the proper "Source Files" folder.

It appeared as though the file was missing from the build.
(probably mostly only cosmetic, but was also was confusing).

==== CL 9455 ====

Back out changelist 9453, 9454

Changes were somehow not effectively made to the vcproj files, so trying again after backing off these CLs.

==== CL 9451 ====

@NEW: adding "name" column to the "banned" table

Note that this involves a DB table schema change. A new table definition, QbTableVersion 30, is added, and will be released with 6.4.0

BUGZID: 63681

ZD: 5271

==== CL 9449 ====

@FIX: fixed issue with removal of workers using the mac address (i.e. "qbadmin -worker remove <macaddr>") not working properly.

BUGZID: 63447

==== CL 9446 ====

@FIX: added "pgrp" modifying support to the supervisor code and the qbmodify() C++ API, qb.modify() Python API, and qb::modify() Perl API routines, and added a "-mpgrp <int>" option to the qbmodify command-line tool.

BUGZID: 63680

==== CL 9443 ====

@FIX: Added missing "qb.hostorder(id=JOBID)" routine to the python API.

==== CL 9442 ====

@FIX: modified to raise exception when parameter "fields" is not of type list.

BUGZID: 63627

ZD: 3998

==== CL 9440 ====

@FIX: variables such as \$qb::jobid not working in callbacks on Windows

BUGZID: 63686

ZD: 5240

==== CL 9438 ====

@FIX: minor fix to a perl example, [callback3.pl](#), so that the job cmdline works in Windows too.

==== CL 9427 ====

@FIX: added code to make sure all end-of-line in email data are CRLF (not just LF) in accordance to RFC2822.

This was causing notification emails to not work with some email servers, as they will not responding, and the communicating supe thread would just stall.

ZD: 5752

==== CL 9411 ====

@FIX: added code to chmod and open up the file permission of .out and .err files in the job log folder.

This was causing subjobs to fail on systems with "mounted" job log path, as the supervisor will initially create these files when when a subjob that previously never started is retried (the supe writes "qube! - retry/requeue on blahblah...") under the "root" user's ownership with mode 644, and the workers who get the subjobs can't write to it.

ZD: 5965

==== CL 9407 ====

@CHANGE: set upper limit for mysql user filehandles to 70,000; 'open-files-limit' setting in my.cnf is only a suggestion, mysql can auto-determine to a larger number, but it's internal max value in 65535. Setting ulimit upper bound larger than 64K should prevent mysql from ever running out of file handles.

==== CL 9402 ====

@FIX: adding "qbhash" command to windows.

==== CL 9395 ====

@FIX: fixed issue causing the supervisor to crash at initialization, right after "finding other supes..." was printed in the supelog.

The fix was in one of the base communication library routines QbConnection::receiveUdp().

Sometimes, unknown/malformed data would be received on the UDP socket, and was causing the code to attempt to access beyond the buffer array (index out-of-bounds error).

ZD: 5638

BUGZID: 63305

==== CL 9370 ====

@FIX: recreate the pfx_dw stored procedures and functions on Windows, as the MSI installer wipes them out during an upgrade.

==== CL 9342 ====

@FIX: fixed a supe thread crashing issue, when global_host or license_host resource tracking is used.

ZD: 5749

==== CL 9334 ====

@FIX: add error handler for MySQL error 1146 "Table 'x' doesn't exist" for work and cpu time calculations for job data collector script

@NEW: increment datawarehouse version to 10 to allow for installing this patch into existing databases

==== CL 9318 ====

@FIX: fixed crash bugs that were introduced when the "dying" state was implemented for 6.3.1.

ZD: 5794

==== CL 9312 ====

@FIX: add mail template for auto-wrangling emails to the installers

==== CL 9299 ====

@FIX: add mail template for auto-wrangling emails to the installers

==== CL 9277 ====

@NEW: increase file handle limit for mysql user on Linux installs to 64K

==== CL 9274 ====

@FIX: create global resource tables in data warehouse DB if they don't exist; creation was failing to happen in new DB installations.

==== CL 9265 ====

@FIX: fixed job-level history not being recorded into .hst file.

(Bug was introduced in CL9145, 9146)

ZD: 5609

==== CL 9261 ====

@CHANGE: cut down on the cmdline & cmdrange jobtypes' stdout; don't print 'LOG: ...' lines, make regex summaries much clearer, change printing or regex's to stderr to make it clearer that they're not actual errors, but rather things being searched for in the stderr stream.

==== CL 9252 ====

@FIX: properly find qb.conf on Windows versions Vista and later when unable to contact the supervisor directly.

==== CL 9245 ====

@FIX: GUI changes to be able to handle when supervisor host goes down, and both supervisor and MySQL server are unavailable. Also fix jobList not refreshing on down supervisor.

==== CL 9241 ====

@FIX: fix GUI crashbug in MySQLConnect when supervisor does not answer a qb.ping

==== CL 9239 ====

@FIX: global resource tables were not getting created in new instances of the datawarehouse db, only on upgrades.

==== CL 9232 ====

@FIX: fixed example python code (jobSubmit06.py) to work on Windows too.

==== CL 9211 ====

@FIX: added code to prevent the QbQueue::getSubjobReadyfindReady() routine from returning the same subjob to be dispatched over and over.

This was causing the findSubjobAndReserveJob() and startJob() routines to hit the "subjob [N] seems to be already assigned" situation, and cause threads to enter a long, sometimes semi-infinite, sleep-and-retry loop.

Fixed by adding code in the startJob() routine to quickly update the subjob

status when the the assignJob() returns QB_ASSIGN_OK (i.e., worker says it has accepted the subjob), instead of waiting until the worker later reports that the subjob is "running" via the STATUS_JOB message, which can take more than several seconds on a busy farm.

Also reduced the number of maximum retries to 3 (MAX_ATTEMPTS), in the situations where a subjob "seems to be already assigned" or when a worker host says it's busy (QB_ASSIGN_BUSY). This prevents the threads to get stuck for 10 or more seconds in a sleep-retry loop, and allow them to give up quickly and move on.

ZD: 5449

==== CL 9198 ====

@FIX: fixed issue with non-node-locked licenses ("FF:FF:...") not working (since 6.3.0)

==== CL 9174 ====

@INTEG: rel-6.3->main CL 9173

@FIX: ensure that mail sent by "qbadmin --emailtest" is RFC2822-compliant (no bare LF's, only CRLF)

==== CL 9173 ====

@FIX: ensure that mail sent by qbamdin --emailtest is RFC2822-compliant (no bare LF's, only CRLF)

==== CL 9161 ====

@NEW: add support for new 'dying' state into the GUI

==== CL 9150 ====

@INTERNAL: QbDebug::filename(QbString) took if statement out, so resetting _filename is allowed

==== CL 9145 ====

@FIX: disabled logging to /var/spool/qube/{host,user}, as it was creating large log files and causing sluggish performance.

An option to enable these logs may be made available in the future.

==== CL 9142 ====

@FIX: fixed issue where global resources tracking drift and more subjobs than can be accommodated by the actual global resource count is dispatched.

ZD: 5074

==== CL 9133 ====

@INTERNAL: CentOS support for "buildpyc" in rpm/quberpm.pm

==== CL 9105 ====

@NEW: A new transitional "dying" state for jobs that have been ordered to be "killed", but still being processed by the system

==== CL 9084 ====

@CHANGE: increase MySQL max_allowed_packet value from default of 1MB to 64MB to decrease frequency of "MySQL server has gone away (2006)" error messages.

==== CL 9083 ====

@CHANGE: increase MySQL wait_timeout value from default of 8 hours to 36 hours to decrease frequency of "MySQL server has gone away (2006)" error messages.

==== CL 9066 ====

@FIX: fixed "cpus" (subjob) count inaccuracy when a job's "cpus" was modified down and then up.

For example, if a job with initially 10 "cpus" was reduced to 5, then subsequently increased to 6, the system had inaccurately recomputed the subjob count to be 10.

==== CL 9058 ====

@FIX: renaming logs during rotation would fail on Windows

==== CL 9037 ====

@FIX: rename the globalResource_fact table to be all lower-case; causes issues stored procedure PFX_CREATE_DATASUBSET_TABLE() which errors out with "ERROR 1050 (42S01) at line 1: Table 'globalresource_fact_12h' already exists" (note lower-cased name)

==== CL 9016 ====

@NEW: adding license agreement for 3rd-party software

@NEW: also adding our own License.rtf to the docs dir.

==== CL 9013 ====

@NEW: added description of supervisor_job_flags in the qb.conf.template file

==== CL 9010 ====

@FIX: fixed memory bloat issue in supervisor threads on start up, on farms with many jobs.
In some cases, it had been reported that each supe thread was taking up 500+ MB.

==== CL 8939 ====

@FIX: fixed another small "hole" that could cause race-conditions to dispatch a single subjob more than once

ZD: 4783

BUGZID: 63657

==== CL 8937 ====

@FIX: supe issue where the same subjob can be dispatched more than once to worker(s).

ZD: 4783

BUGZID: 63657

Qube 6.3

@RELEASE: 6.3.6
#####

==== CL 10514 ====

@FIX: another patch for out-of-order issue. Fixed unexpected short-circuit evaluation that was happening in the startResources() routine

==== CL 10513 ====

@FIX: another patch for out-of-order issue. Fixed unexpected short-circuit evaluation that was happening in the startHost() routine

==== CL 10512 ====

@INTERNAL: QbJob object's _subjobswaiting data was not being initialized or copied correctly, causing some job comparisons based on subjobs waiting counts to unexpectedly fail.

==== CL 10504 ====

@INTERNAL: added more log output for debugging builds, added more comments while working on out-of-order issue.

ZD: 8198

==== CL 10477 ====

@FIX: Another out-of-order fix. Jobs at the same numerical and cluster priority should dispatch in the correct FIFO order now.

The FIFO enforcing should work most of the time, but there still will be occasional out-of-order behavior, due to the multi-threaded nature of the supervisor. ("qbshove"-ing the older job should correct it, when it's seen)

ZD: 8198

==== CL 10462 ====

@FIX: yet yet another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.

See also CL10440 10452

ZD: 8198

==== CL 10461 ====

@CHANGE: modified/compacted the multi-line "found a duty to replace" logging to be a single line.

==== CL 10452 ====

@FIX: yet another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.

See also CL10440

ZD: 8198

==== CL 10441 ====

@FIX: killing an already finished (complete, failed, killed) job leaves the job in the "dying" state.

==== CL 10440 ====

@FIX: another fix for out-of-order dispatch behavior-- eliminate race-condition that would allow lower priority jobs that were just preempted to get workers before higher-priority jobs.

ZD: 8198

==== CL 10429 ====

@FIX: out-of-order job dispatching issue with jobs using the "+" sign with the "host.processors" reservations.

ZD: 8198 8261 8229 8233 8228

==== CL 10189 ====

@FIX: timing issue where some worker resources (host.xyz) would disappear after the worker received a remote config.

@FIX: issue where supervisor tries to dispatch a subjob to a worker with insufficient resources (reduced the likeliness of that from happening)

@FIX: the above 2 fixes combined should now prevent some of the out-of-priority-order dispatch issues, especially in environments where worker resources are deployed.

ZD: 7885

==== CL 10118 ====

@FIX: fixed issue where agenda timeouts don't work properly on the first agenda item processed by a subjob, on Unix (Linux/OSX) workers

==== CL 10117 ====

@FIX: fixed issue where agenda items that fail because of timeout don't get automatically retried via retrywork

ZD: 7763

==== CL 10022 ====

@FIX: modified the worker to only report to the supe of its host status when subjobs are completely done and removed, and NOT when they are only marked/scheduled for removal.

This was causing jobs to sometimes run out-of-order, especially when there are many subjobs to each job (such as one subjob per frame), since that situation tends to increase the chance of the supervisor dispatching the same subjob to the same worker. The subjob will be dispatched to the same worker, but rejected since the worker thinks it's a duplicate assignment of a subjob that's being removed (and consequently a lower priority job will get the worker's slot, causing out-of-order job execution)

ZD: 7601

==== CL 9903 ====

@FIX: better message from worker when it rejects a dispatched subjob because it's a duplicate (being preempted or migrated on the same worker)

==== CL 9838 ====

@CHANGE: upped the default value for supervisor_max_threads to 100, and worker_max_threads to 32

#####

@RELEASE: 6.3.5

#####

==== CL 9785 ====

@FIX: worker issue where desktop worker would randomly crash.

ZD: 6778

==== CL 9730 ====

@TWEAK: modified so that worker name and IP print when job is accepted by worker, in assignJob()

==== CL 9729 ====

@INTERNAL: changed all calls to qbvcout to qbout in the QbDaemon, QbPreforkDaemon and QbDatabaseMysql code, so that the timestamp, hostname and pid, are always printed.

==== CL 9698 ====

@FIX: fixed false-negative warning message pertaining to "select() in checkpoint()" seen in supelog.

Examples of these messages:

select() in checkpoint(): Operation timed out
select() in checkpoint(): Interrupted system call

==== CL 9694 ====

@FIX: fixed issue with the supe threads getting tied up on "subjob X seems to be already assigned" message.

On a farm with busy workers, the time between the supe dispatching a sub job to the worker via assignJob() and the worker reporting that the "subjob is running" can be several seconds to sometimes even several minutes, which was causing many supe threads to attempt dispatching the same subjob over and over. All of those threads end up hitting the "subjob X seems to be already assigned... retrying" message, and get tied up for 3 seconds while they retry.

BUGZID:

ZD: 6760 7125

==== CL 9689 ====

@FIX: fixed bug in clustering algorithm where it incorrectly gave more weight to a job when the only difference was the last letter in the cluster specification.

For example, if:

host cluster: /3D/projA

job1 cluster: /3D/projB

job2 cluster: /3D

job1 was getting more weight than job2, which is incorrect.

BUGZID: 63740

ZD: 7043

==== CL 9686 ====

@FIX: using deprecated "waitfor" attribute with Python api causes qb.submit() to raise a KeyError

@FIX: properly convert "waitfor" value (jobid integer) to proper "dependency" string of "link-done-job-<id>"

==== CL 9676 ====

@FIX: update documentation and GUI help text to show correct "||" syntax for job restrictions list.

==== CL 9662 ====

@FIX: supervisor was failing postflight upgrade scripts on OSX Server, explicitly set the mysql socket to /tmp/mysql.sock in /etc/my.cnf and /etc/qb.conf to avoid conflicting with the factory-installed default of /var/lib/mysql/mysql.sock

==== CL 9615 ====

@FIX: Added code to properly log frames (to supelog and job log) when they go back to "pending" after the processing subjob/worker is found dead.

@FIX: Added code in the supervisor to retry a failed worker connection after a random 5-10 sec sleep/delay, to alleviate network hiccups during network commands (kill, preempt, etc. of running subjobs).

ZD: 6760

==== CL 9614 ====

@INTERNAL: fixed a small cosmetic bug introduced in CL 9606

==== CL 9607 ====

@INTERNAL: added converseWorkerWithRetries() and also fixed small bug in the retry loop of converseSubSupervisorWithRetries()

==== CL 9585 ====

@FIX: issue where some jobs get stuck in the "dying" state when attempted to be killed

ZD: 6616

==== CL 9570 ====

@FIX: improvements to the handling of GET_LOCK (aka "reserveJob()") timeout situations.

ZD: 6617

==== CL 9500 ====

@FIX: Windows Vista/7/2008-R2 installer - don't error out when installing the worker or supervisor as an Admin-equivalent account during creation of scheduled tasks. Properly remove scheduled tasks during uninstall.

@RELEASE: 6.3.4
#####

==== CL 9550 ====

@FIX: qbwrk.conf files that had any commented-lines before the first valid template was encountered would cause an exception to be raised, QubeGUI->worker->RMB->Configure (which uses qb.updateworkerconfig()) would fail silently

==== CL 9535 ====

@NEW: add submit-agenda-timeout-job.py example python script, to demonstrate submission of a job with frame-level timeouts.

ZD: 6099

==== CL 9530 ====

@FIX: Submitting paths to shotgun no longer depends on the visibility of output paths to the supervisor.

@FIX: Shotgun submission script fails gracefully & logs a reason as to why it can't generate a thumbnail when thumbnail creation fails.

==== CL 9523 ====

@FIX: fixed issue where the supervisor fails to correctly track the host assignment for subjobs.

Symptom for this included seeing in the supelog, messages like "statusJob(): aberrant report from worker...", then followed by "subjob[xxxx] is assigned to worker[]" with mac address[00:00:00:00:00:00]".

These subjobs would then be in the "running" state, but not assigned to a worker.

==== CL 9522 ====

@FIX: removed code that skipped code that made local decision on the supe to test for resource reservations, for jobs with host.processors set to > 1, delegating the decision-making to the workers and resulting in more network traffic and latency.

ZD: 6141

==== CL 9507 ====

@FIX: added more robust code that talks to the SMTP server when sending out email, to support some email servers with non-standard response behavior.

ZD: 6209

==== CL 9504 ====

@FIX: catch case where sg_path_to_frames is part of the Shotgun versionName, but the job has no outputPaths for the first frame; fallback to naming the version "job id: 123 jobName: ..."

==== CL 9496 ====

@FIX: catch case when inserting in a new cluster into cluster_dim when more than 1 worker exists in the new cluster; occurs during run of regular_slotcount.sql, doesn't prevent new record from being added, just generates line noise and error emails from cron...

==== CL 9494 ====

@CHANGE: make explanation of "+ | *" in job/host restrictions less ambiguous

==== CL 9484 ====

@FIX: calculate cpu-seconds for agenda-based jobs by summing up work times, not subjobs. Better support for resetting of the start times for retried work.

==== CL 9467 ====

@NEW: add a random offset to the startup so that all workers don't report at the same time if they've started up at the same time.

@CHANGE: don't retrieve job name, it's extraneous and not reported; cuts down the query count by one.

@CHANGE: set workname for subjob to job.subid, not subid; easier to detect case where an agenda-based job falsely reports not having an agenda, so subjob id won't conflict with a frame number

==== CL 9463 ====

@FIX: don't report memory usage in the case where MySQL fails to return a valid agenda name, usually caused by timeouts or maxed out connections.

==== CL 9456 ====

@FIX: moved the location of QbTableVersion29.cpp (rel-6.3) inside the upgrade_supervisor.vcproj file from the incorrect "Resources Files" folder to the proper "Source Files" folder.

It appeared as though the file was missing from the build.
(probably mostly only cosmetic, but was also was confusing).

==== CL 9449 ====

@FIX: fixed issue with removal of workers using the mac address (i.e. "qbadmin -worker remove <macaddr>") not working properly.

BUGZID: 63447

==== CL 9446 ====

@FIX: added "pgrp" modifying support to the supervisor code and the qbmodify() C++ API, qb.modify() Python API, and qb::modify() Perl API routines, and added a "-mpgrp <int>" option to the qbmodify command-line tool.

BUGZID: 63680

==== CL 9442 ====

@FIX: modified to raise exception when parameter "fields" is not of type list.

BUGZID: 63627

ZD: 3998

==== CL 9440 ====

@FIX: variables such as \$qb::jobid not working in callbacks on Windows

BUGZID: 63686

ZD: 5240

==== CL 9427 ====

@FIX: added code to make sure all end-of-line in email data are CRLF (not just LF) in accordance to RFC2822.

This was causing notification emails to not work with some email servers, as they will not responding, and the communicating supe thread would just stall.

ZD: 5752

==== CL 9411 ====

@FIX: added code to chmod and open up the file permission of .out and .err files in the job log folder.

This was causing subjobs to fail on systems with "mounted" job log path, as the supervisor will initially create these files when when a subjob that previously never started is retried (the supe writes "qube! - retry/requeue on blahblah...") under the "root" user's ownership with mode 644, and the workers who get the subjobs can't write to it.

ZD: 5965

==== CL 9402 ====

@FIX: adding "qbhash" command to windows.

==== CL 9395 ====

@FIX: fixed issue causing the supervisor to crash at initialization, right after "finding other supes..." was printed in the supelog.

The fix was in one of the base communication library routines QbConnection::receiveUdp().

Sometimes, unknown/malformed data would be received on the UDP socket, and was causing the code to attempt to access beyond the buffer array (index out-of-bounds error).

ZD: 5638

BUGZID: 63305

#####

@RELEASE: 6.3.3

#####

==== CL 9370 ====

@FIX: recreate the pfx_dw stored procedures and functions on Windows, as the MSI installer wipes them out during an upgrade.

==== CL 9342 ====

@FIX: fixed a supe thread crashing issue, when global_host or license_host resource tracking is used.

ZD: 5749

==== CL 9334 ====

@FIX: add error handler for MySQL error 1146 "Table 'x' doesn't exist" for work and cpu time calculations for job data collector script

@NEW: increment datawarehouse version to 10 to allow for installing this patch into existing databases

==== CL 9325 ====

@FIX: add qbhash program to be included in qube-core RPM package.

BUGZID: 63693

ZD: 5744

==== CL 9318 ====

@FIX: fixed crash bugs that were introduced when the "dying" state was implemented for 6.3.1.

ZD: 5794

==== CL 9311 ====

@FIX: add mail template for auto-wrangling emails to the installers

#####

@RELEASE: 6.3.2

#####

==== CL 9265 ====

@FIX: fixed job-level history not being recorded into .hst file.

(Bug was introduced in CL9145, 9146)

ZD: 5609

==== CL 9261 ====

@CHANGE: cut down on the cmdline & cmdrange jobtypes' stdout; don't print 'LOG: ...' lines, make regex summaries much clearer, change printing or regex's to stderr to make it clearer that they're not actual errors, but rather things being searched for in the stderr stream.

==== CL 9252 ====

@FIX: properly find qb.conf on Windows versions Vista and later when unable to contact the supervisor directly.

==== CL 9245 ====

@FIX: GUI changes to be able to handle when supervisor host goes down, and both supervisor and MySQL server are unavailable. Also fix jobList not refreshing on down supervisor.

==== CL 9241 ====

@FIX: fix GUI crashbug in MySQLConnect when supervisor does not answer a qb.ping

==== CL 9239 ====

@FIX: global resource tables were not getting created in new instances of the datawarehouse db, only on upgrades.

==== CL 9234 ====

@FIX: disable permission check of worker_logpath, as it was creating false-alarms and putting the worker to be in panic mode unnecessarily.

ZD: 5445 5236

BUGZID: 63683

==== CL 9232 ====

@FIX: fixed example python code (jobSubmit06.py) to work on Windows too.

==== CL 9211 ====

@FIX: added code to prevent the QbQueue::getSubjobReadyfindReady() routine from returning the same subjob to be dispatched over and over.

This was causing the findSubjobAndReserveJob() and startJob() routines to hit the "subjob [N] seems to be already assigned" situation, and cause threads to enter a long, sometimes semi-infinite, sleep-and-retry loop.

Fixed by adding code in the startJob() routine to quickly update the subjob status when the assignJob() returns QB_ASSIGN_OK (i.e., worker says it has accepted the subjob), instead of waiting until the worker later reports that the subjob is "running" via the STATUS_JOB message, which can take more than several seconds on a busy farm.

Also reduced the number of maximum retries to 3 (MAX_ATTEMPTS), in the situations where a subjob "seems to be already assigned" or when a worker host says it's busy (QB_ASSIGN_BUSY). This prevents the threads to get stuck for 10 or more seconds in a sleep-retry loop, and allow them to give up quickly and move on.

ZD: 5449

==== CL 9198 ====

@FIX: fixed issue with non-node-locked licenses ("FF:FF:...") not working (since 6.3.0)

==== CL 9173 ====

@FIX: ensure that mail sent by qbamdin --emailtest is RFC2822-compliant (no bare LF's, only CRLF)

```
#####  
@RELEASE: 6.3.1  
#####
```

=====

@NEW: Add CentOS/RHEL 6 x64 support

==== CL 9150 ====

@INTERNAL: QbDebug::filename(QbString) took if statement out, so resetting _filename is allowed

==== CL 9145 ====

@FIX: disabled logging to /var/spool/qube/{host,user}, as it was creating large log files and causing sluggish performance.

An option to enable these logs may be made available in the future.

==== CL 9142 ====

@FIX: fixed issue where global resources tracking drift sand more subjobs than can be accomodated by the actual global resource count is dispatched.

ZD: 5074

==== CL 9133 ====

@INTERNAL: CentOS support for "buildpyc" in rpm/quberpm.pm

==== CL 9105 ====

@NEW: A new transitional "dying" state for jobs that have been ordered to be "killed", but still being processed by the system

==== CL 9085 ====

@INTEG: main -> rel-6.[0,1,2,3] CL 9083, 9084

@CHANGE: increase MySQL wait_timeout value from default of 8 hours to 36 hours to decrease frequency of "MySQL server has gone away (2006)" error messages.

@CHANGE: increase MySQL max_allowed_packet value from default of 1MB to 64MB to decrease frequency of "MySQL server has gone away (2006)" error messages.

==== CL 9084 ====

@CHANGE: increase MySQL max_allowed_packet value from default of 1MB to 64MB to decrease frequency of "MySQL server has gone away (2006)" error messages.

==== CL 9066 ====

@FIX: fixed "cpus" (subjob) count inaccuracy when a job's "cpus" was modified down and then up.

For example, if a job with initially 10 "cpus" was reduced to 5, then subsequently increased to 6, the system had inaccurately recomputed the subjob count to be 10.

==== CL 9058 ====

@FIX: renaming logs during rotation would fail on Windows

==== CL 8939 ====

@FIX: fixed another small "hole" that could cause race-conditions to dispatch a single subjob more than once

ZD: 4783

BUGZID: 63657

==== CL 8937 ====

@FIX: supe issue where the same subjob can be dispatched more than once to worker(s).

ZD: 4783

BUGZID: 63657

```
#####  
@RELEASE: 6.3.0  
#####
```

==== CL 9013 ====

@NEW: added description of supervisor_job_flags in the qb.conf.template file

==== CL 9010 ====

@FIX: fixed memory bloat issue in supervisor threads on start up, on farms with many jobs.
In some cases, it had been reported that each supe thread was taking up 500+ MB.

==== CL 8975 ====

@NEW: add section (8.7) for "externally updatable worker resources and properties" to Administration.doc

==== CL 8957 ====

@NEW: add user name to print to supelog when a worker lock is updated

BUGZID: 63661

ZD: 4860

==== CL 8949 ====

@FIX: fix datawarehouse crontab so that 7-day tables are rebuilt twice a day

==== CL 8948 ====

@NEW: add global_resource tracking to the datawarehouse

==== CL 8935 ====

@FIX: update qb.conf templates to show the correct default value for supervisor_default_security
@INTERNAL: previous setting was an hex value, which seems to be unsupported now.

==== CL 8910 ====

@NEW: add C++ examples for using the qbupdateworkerresource(), qbupdateworkerproperties(), qbdeleteworkerresources(), and qbdeleteworkerproperties() routines

==== CL 8909 ====

@NEW: add Perl API routines for externally updated worker resources/properties

* add bindings to perl

** add qb::updateworkerresources() and updateworkerproperties() to perl api

qb::updateworkerresources("shinyamp.local", "host.ooga=2/3,host.exten=0/10")

qb::updateworkerrproperties("shinyamp.local", "host.oogaprop=3,host.oogaextprop2=11")

** add deleteworkerresources() and deleteworkerproperties() to perl

qb::deleteworkerresources(\$host, @resources);

qb::deleteworkerresources("shinyamp.local", "host.extenres", "host.ooga");

==== CL 8901 ====

@FIX: fixed bug where subjobs will be retried indefinitely when retrysubjob is set.

BUGZID: 63517

ZD: 2950 4661

==== CL 8889 ====

@FIX: fixed issue where the supervisor kept adding duplicate auto-wrangling and mail callbacks when jobs are resubmitted

BUGZID: 63655

ZD: 4661

==== CL 8886 ====

@INTEG: rel-6.2 -> main

@FIX: properly remove datawarehouse scheduled tasks for round-robin tables

==== CL 8885 ====

@FIX: properly remove datawarehouse scheduled tasks for round-robin tables

==== CL 8872 ====

@FIX: issue introduced in 6.2.1 that broke callbacks (not being triggered)

==== CL 8859 ====

@FIX: add bookmarks (TOC) to Admin docs, update section for qblock to refer to "Users guide" instead of non-existent "Command Reference"

==== CL 8857 ====

@NEW: add externally-updatable worker resources and properties

BUGZID:

==== CL 8847 ====

@CHANGE: upgrade_config tool no longer comments out some of the customized paths in qb.conf

ZD: 4470

==== CL 8846 ====

@FIX: supe and worker RPMs now correctly "require" specific qube-core version (like "6.2-1")

BUGZID: 63644
ZD: 4470

==== CL 8841 ====

@FIX: issue with supervisor threads stalling, waiting for NFS I/O on the "mounted" job logs, when NFS latency is large.

==== CL 8840 ====

@UPDATE: "Use" doc with p-agenda documentation

@UPDATE: also added/updated some qbsub examples

BUGZID: 63636

==== CL 8837 ====

@NEW: add example scripts to demonstrate submission of p-agenda jobs in perl and python

BUGZID: 63636

==== CL 8836 ====

@NEW: adding docs for retryworkdelay (qbsub option)

==== CL 8811 ====

@FIX: fixed worker installer to start the worker service iff the system has not already turned it OFF via chkconfig.

ZD: 4286

==== CL 8798 ====

@NEW: optimization when submitting big groups of jobs via qbsubmit() loaded with callbacks and dependencies
Fixed reported issue where submission performance will degrade linearly proportional to the number of jobs in the queue.

==== CL 8795 ====

@UPDATE: added descriptions of new/missing qb.conf parameters to the qb.conf.template file, which is used to build the default qb.conf.

- * added p-agenda params (supe and client)
- * added auto-wrangling params (supe)
- * added per-user/pgrp subjob limit params (supe)
- * added mail setup params (supe)
- * added database setup params (supe)

==== CL 8794 ====

@NEW: add p-agenda submission options to qbsub (p_agenda, p_priority, and p_cpus), and updated online help text.

==== CL 8790 ====

@CHANGE: Python API qb.reportjob() now takes a subjob object (dict). It can still take just the status (string).

This should enable the custom jobtype back-end programmer to pass back subjob-level "resultpackage" data to the supe, for example.

==== CL 8783 ====

@NEW: add supervisor_p_agenda_max qb.conf parameter, for the site-admin to control the maximum number of p-agenda any job can have.

==== CL 8782 ====

@NEW: add p_agenda_cpus to enable control of the number of "cpus" used for the p-agenda jobs. Defaults to number of p-agenda items.

@CHANGE: removed code that automatically makes a job to become a p-agenda job when p_agenda_priority() is set. The "p_agenda" list or the "p_agenda" job flag must be explicitly set for a job to be a p-agenda job.

==== CL 8781 ====

@CHANGE: if an agenda-based job specifies the p_agenda_priority, then automatically add the p_agenda flag.

@CHANGE: added code to check that the job being submitted is an agenda-based one, before doing the p-agenda magic

==== CL 8775 ====

@UPDATE: doc update w/ "qbcrypt" and encrypted DB password descriptions

@UPDATE: Added section for qbcrypt, and updated section for qblogin.

@UPDATE: section for database_password

BUGZID: 63383 63628 39741

==== CL 8769 ====

@NEW: add "qbcrypt" tool, used to generate/display encrypted passwords

@NEW: add "-password" option to qblogin, to specify password in a command-line option instead of on the stdin

BUGZD: 63383

==== CL 8767 ====

@FIX: install datawarehouse plists on OSX (missing from installer package)

==== CL 8764 ====

@NEW: add p-agenda (p-frames, "p" stands for Priority/Preview/Poster) support, where a select few agenda items of a job can be sent at a higher priority for quicker turn around for previewing purposes.

To use in API: set the "p_agenda" job flag when submitting an agenda-based job.

Optionally attach a list, job['p_agenda'] in python API, to the job on submission to explicitly specify the p-agenda items. If not set explicitly, the system will automatically choose the 1st, last, and middle items to be rendered at a higher priority.

The priority of the p-agenda items may also be specified on submission, by setting the job's p_agenda_priority parameter.

p-agenda job support for the standard submission tools (GUI, qbsub) coming shortly.

@NEW: qb.conf parameters: client_p_agenda_priority, supervisor_default_p_agenda_priority (default 1)

==== CL 8760 ====

@UPDATE: Administration.doc with details about the new worker_boot_diagnostic_retries and worker_boot_diagnostics_retry_interval parameters

BUGZID: 63600

==== CL 8755 ====

@FIX: Added worker_boot_diagnostics_retries and worker_boot_diagnostics_retry_interval

These new configuration parameters tell the worker to automatically retry the boot-time diagnostic routines for "worker_boot_diagnostics_retries" times, with "worker_boot_diagnostics_retry_interval" seconds of sleep time inbetween the retries. By default, they are set to 1 and 30 (seconds) respectively. These values may be set in the local qb.conf file, or in the qbrwk.conf file.

@FIX: issue where worker will "panic" when proxy settings are set in the remote qbrwk.conf file.

BUGZID: 63600 63422 63407

ZD: 3650 1638 2035

==== CL 8743 ====

@NEW: add qb.frontend package, will serve as base class for constructing jobs for new python jobtypes

==== CL 8727 ====

@CHANGE: database_password is now expected to be encrypted.

Plain text password still works, but if a password has been set up to access the MySQL db, site administrators are recommended, but not required, that they use "qblogin -display" to generate the encrypted password, and set database_password in qb.conf to the encrypted string for more security.

BUGZID: 63628

==== CL 8722 ====

@NEW: add optional artificial delay before auto-retry of agenda items via "retrywork"

When a failed frame is automatically retried via "retrywork", an artificial delay may be inserted before the subjob starts processing it. Requested by customers to work around issues with, for example, application license contentions.

Submission APIs (C++, Perl, Python) and clients (qbsub, QubeGUI) modified to allow specifying "retrywork_delay" when submitting jobs.

==== CL 8717 ====

@FIX: logs written into a "hidden" file, in "log/user/.hst", which grows very large

Actions initiated by the supe (as opposed to a particular user), such as "starting a subjob on worker", were logged into this hidden ".hst" file. Fixed it so the file has a special folder/name, "__QUBE_SYSTEM__/__QUBE_SYSTEM__.hst".

Also modified code so that if the "user" flag was omitted from the "supervisor_log_flags", then this user action logging is disabled altogether.

BUGZID: 62030

==== CL 8713 ====

@FIX: turned off worker debug-level logging that accidentally made it into the 6.2.0 release.

==== CL 8712 ====

@FIX: issue where worker processes will stall when a config field, such as "worker_description" has quotes in them.

==== CL 8704 ====

@FIX: support bash exported function definitions, which are saved as multi-line environment variable values

BUGZID: 63624

ZD: 4100

==== CL 8702 ====

@NEW: add perl 5.12 and 5.14 support for windows x64 and 32-bit.

BUGZID: 63631

==== CL 8695 ====

@FIX: export_environment now works properly with built-in cmd* jobtypes

@FIX: cmd* jobtype backends will run jobs in a non-login shell if export_environment flag is set on the job, to avoid overriding of environment variables set by the job's submission environment.

@NEW: QbApi::qbsystem() now optionally takes a boolean to specify commands to be run in a login shell.

@CHANGE: By default now, QbApi::qbsystem() will run the given command in a non-login shell.

@NEW: added optional "shell" parameter to QbEnv::setToEnv(user, [shell]) method, so the user environment for a non-default shell can be fetched. This new method is called from QbWorker::QbUnix.cpp now.

BUGZID: 63625

ZD: 4100

==== CL 8693 ====

@FIX: fix "ERROR 1290 (HY000) at line 31 in file: '.\create_stored_programs.sql'" on new Windows installations

@FIX: fix Windows 5.15-beta version specific SQL syntax error (does not exhibit in later versions of MySQL)

==== CL 8676 ====

@FIX: add code to license check routine to validate hostid against all mac addresses on the host, as opposed to just the primary one.

Note: this involves changes to the base library (utils/QbList, utils/QbServer)

BUGZID: 63621

--

@CHANGE: modify license verification code to only run when the license file had been changed, or a new day has arrived, or on boot.

The code still checks to modification time of the license file everytime that a license access is required but most of the logic is now short-circuited, if no mod was made to the file.

It turns out to be rather tricky to, say, add a "reread" option to "qbadmin" to only read the license on demand, since all supe thread must be told to read the file (for quick access, license data is kept in memory of each thread/proc), and such "broadcast" type of instruction to go out to all threads is not supported at the moment.

The optimization being checked in, however, should significantly reduce the overhead in license-checking nonetheless, especially with the new code where each license key's hostid is checked against all mac addresses for validation.

BUGZID: 63622

==== CL 8668 ====

@FIX: fix "/etc/rc.d/init.d/supervisor: line 139: [: /var/spool/qube/user/jburk/jburk.hst: binary operator expected" error message in supervisor startup

BUGZID:63618

==== CL 8662 ====

@UPDATE: update doc with mail_from parameter description.

BUGZID: 63591

==== CL 8654 ====

@FIX: Made the qube-core RPM "obsolete" the "qube" package, to accomodate the change in RPM package name.

BUGZID: 63611

ZD: 3950

==== CL 8641 ====

@FIX: added more details to default qb.conf template's description of proxy_nice_value, and also included explanation for Windows. Also corrected the commented-out default proxy_account to "qubeproxy" (from "proxyuser") in the same qb.conf.template.

@DOC: update proxy_nice_value doc accordingly.

==== CL 8610 ====

@FIX: issue where supe will install but not run, due to missing python25.dll file.

==== CL 8606 ====

@FIX: The "Start Time" parameter for SCHEDTASKS.EXE (/ST option) must be in hh:mm:ss format for earlier versions of Windows (notably winxp 32).

==== CL 8598 ====

@NEW: add sample perl-based submit script that submits jobs with per-work email notification callbacks.

ZD: 3854

==== CL 8550 ====

@FIX: rolling back to linking supe against python 2.5 for its embedded interpreter instead of 2.7 to avoid runtime linkage issues with 2.7

BUGZID:

==== CL 8547 ====

@FIX: added "post" as possible supervisor_language_flags

@FIX: default for supervisor_manifest_flags should be empty

==== CL 8535 ====

@CHANGE: Enhanced shotgun integration in job submission.

==== CL 8503 ====

@CHANGE: grant access to the PFX_*QBTIME* functions to MySQL user "qube_readonly"

@CHANGE: grant the pfx_dw user all rights to the pfx_stats DB

==== CL 8483 ====

@CHANGE: add support for non-cmdrange type backends, don't require qbTokens

==== CL 8482 ====

@NEW: a framework for python-based jobtype backends, as well as a base class for jobtypes which use an application's embedded python terminal prompt

* for use by the Nuke python jobtype (dynamic allocation)

* used by the intra-frame progress in pycmdrange

* can be used for Houdini jobtype dynamic allocation (not mantra cmd-line renderer though)

==== CL 8480 ====

@FIX: fixed issue with perl API where the system won't respect the "retrywork" specified in jobs processed with a perl-based custom jobtype back-end.

@CHANGE: added some useful logging message to print to supelog when retrywork is being considered

==== CL 8472 ====

@FIX: issue where perl-based custom policy didn't work on some systems.

Embedded perl interpreter had to be initialized much earlier than it was, before the supervisor goes into multi-proc, and before initializing customizable modules (algorithm, policy) that rely on it.

ZD: 3718

BUGZID: 63603

==== CL 8468 ====

@FIX: (Windows) modified worker memory tracking to store values in KB instead of bytes, to avoid buffer overflow.

ZD: 3308

==== CL 8466 ====

@FIX: (OSX) modified worker memory tracking to store values in KB instead of bytes, to avoid buffer overflow.

ZD: 3308

==== CL 8465 ====

@FIX: (Linux) modified worker memory tracking to store values in KB instead of bytes, to avoid buffer overflow.

BUGZID: 3308

ZD:

==== CL 8458 ====

@NEW: add documentation for the "Get Next n Jobs" jobList pagination, document the new behavior of the User filterCtrl, since it now serves as both a display and request filter.

==== CL 8453 ====

@NEW: implement the ability to parse the logs on the fly to determine intra-chunk progress

@INTERNAL: clean up backend base class and backendUtils in preparation of more wide-spread use

==== CL 8449 ====

@NEW: a pure-python implementation of the cmdrange jobtype; it implements intra-chunk progress by parsing the output stream from the command as it's being written to disk during the course of the job, not after the job completes. Progress calculation works on both single- and multiple-item agenda jobs.

==== CL 8436 ====

@NEW: add doc for per-user/pgrp subjobs limits