Qube 6.6-0 Release Notes

##

@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" whenthe 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 addtion 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 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 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 qbjoborder() 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 mystereous 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 "commiting 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"