

Managing Jobs with QubeArtistView

Now that you have submitted some jobs, you will want to monitor them in the Qube UI. In fact, there are now two Qube UIs - the original, now called Wrangler View (because it contains features that an administrator or render wrangler would use) and qubeArtistView, intended for artists to monitor and interact with their jobs.

If you have been working through the Qube 101 documentation in order, you will probably be running the Wrangler View, which has a white background. In this section we are going to skim across the surface of the qubeArtistView UI.



Important

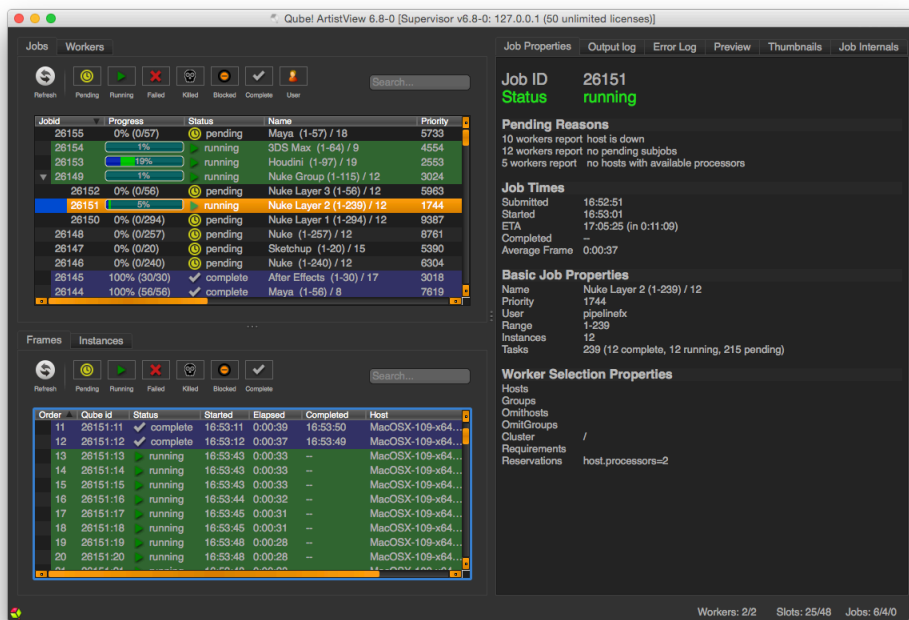
The Qube Client package must be installed on your machine in order for you to find and run the Qube Artist View. If you cannot find the Artist View UI on your machine, review the instructions for [Installing Qube!](#)

Launch the Qube Artist View UI

To launch the Artist View, do this:

- **Windows:** Click on the black and white Qube ArtistView desktop icon, or locate it in the Program Files\pfx\qube\qubeArtistView folder (Note: **not** the "Program Files (x86)" folder)
- **OS X:** Click on the black and white Qube ArtistView desktop icon, or locate it in the Applications/pfx/qube folder
- **Linux:** Run 'qubeArtistView' at the shell prompt (no quotes). By default, qubeArtistView will be located in /usr/local/pfx/qube/qubeArtistView/.

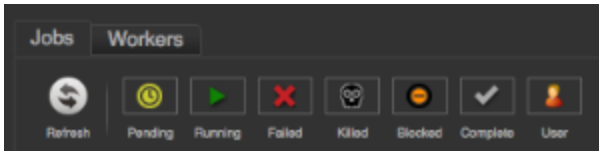
You should see this:



You should see the test jobs or other jobs that you have submitted. Notice that they are color coded - if you are lucky, they will all be blue, which is the color of a successfully completed job. Jobs which failed are colored red. The jobs also have a status, such as "complete", "failed" or "running." Running jobs are shown in green. There are other colors for other statuses, which you will become familiar with as you submit and monitor jobs.

Overview

The row of buttons just under the Jobs tab is a set of filters which will show or hide jobs in the Jobs Panel. An important one to know about is the User Filter - when set, you will only see your own jobs. When unset, you will see all jobs. Click and unclick this button to see the difference. Note that if you are on Linux, you may have submitted jobs as the root or qubeproxy user which will be hidden if you choose this filter.



Let's have a look at some simple tasks you can do from this UI. The Jobs pane shows your job(s). By default, they are sorted by Jobid, but you can sort them by any other column by clicking on the column header. For example, you can sort them by Name by clicking on the "Name" label at the top of that column. Clicking once sorts them in ascending order, clicking again sorts them in the other direction. Now, click on one of your test jobs in the Jobs pane.

Jobid	Progress	Status	Name	Priority	User	Eta remaining	Submitted
273	21%	▶ running	maya render ...	5000	andrew	0:03:55	11:44:13
272	0% (0/100)	✖ failed	maya render ...	5000	andrew	--	11:40:47
271	0% (0/10)	⦿ blocked		9999	andrew	--	2014-10
270	100% (2/2)	✔ complete	maya render ...	9999	andrew	--	2014-10
269	100% (1/1)	✔ complete	Qube Test (Is...	9999	andrew	--	2014-10
268	0% (0/4)	✖ failed	maya render ...	9999	andrew	--	2014-10
267	0% (0/4)	✖ failed	maya render ...	9999	andrew	--	2014-10
266	100% (2/2)	✔ complete	maya render ...	9999	andrew	--	2014-10

Notice that in the panel below the Jobs panel, you can look at the Job Properties on one tab, and the Output on the next two tabs (Output Log & Error Log). This is the Job Details panel - we'll cover that in a second.

Job Properties

Output log

Error Log

Preview

Thumbnails

Job Internals

Job ID 26151

Status running

Pending Reasons

- 10 workers report host is down
- 12 workers report no pending subjobs
- 5 workers report no hosts with available processors

Job Times

- Submitted 16:52:51
- Started 16:53:01
- ETA 17:05:25 (in 0:11:09)
- Completed --
- Average Frame 0:00:37

Basic Job Properties

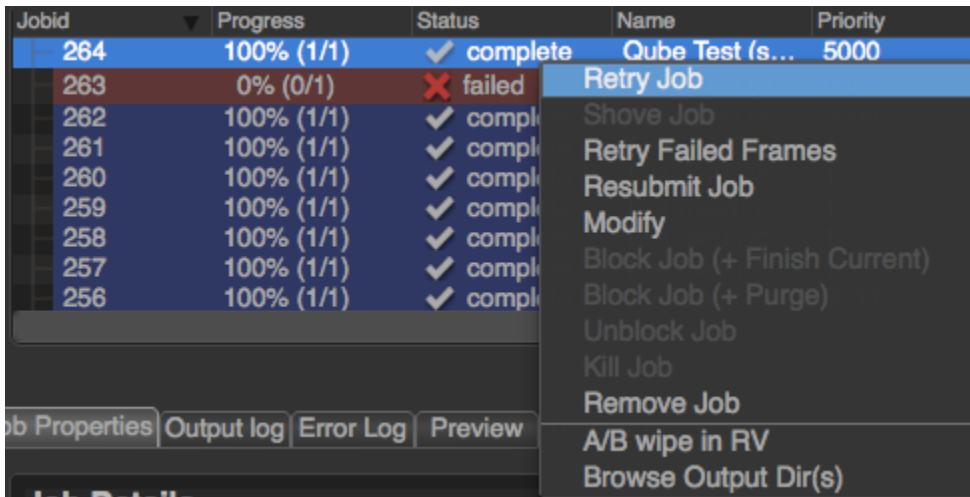
- Name Nuke Layer 2 (1-239) / 12
- Priority 1744
- User pipelinefx
- Range 1-239
- Instances 12
- Tasks 239 (12 complete, 12 running, 215 pending)

Worker Selection Properties

- Hosts
- Groups
- OmitHosts
- OmitGroups
- Cluster /
- Requirements
- Reservations host.processors=2

Interact with a Job

Right-clicking on a job (or a frame, worker, or instance, for that matter), will bring up a context menu:



The contents of that menu depend on the entity you clicked. For example, jobs will have a different list of menu items than will frames or workers. The menu items are dictated by plugins which can be created or modified. A more detailed description of the creation or modification of plugins can be found in [Adding Custom Plugins](#).

As of version 6.8, the list of available actions are as follows:

- **Job**

- **Retry Job:** Start this job over, using the same job id. All frames will be re-rendered, regardless of whether or not they had previously completed.
- **Shove:** Force the supervisor to reevaluate this job for dispatch. This is a good way to start a job that is stuck in the "pending" state, even though all of its requirements can be met.
- **Retry Failed Frames:** Rerender all frames marked as failed
- **Modify:** Change modifiable job properties. When modifying multiple jobs at the same time, you may see `<varies...>` in the field. If you wish to append a value, simply add the new value after `<varies...>`, using the correct delimiter, e.g. `<varies...>, global.resource_name=5`. If you wish to overwrite all values, simply remove `<varies...>` and replace it with the new value.
- **Interrupt:** Stops the job immediately, and sets the status back to pending. This is typically used when a high priority job is starved for resources - manually interrupting a lower priority job will free up those resources to allow the high priority job to start. The lower priority job to start up again as soon as workers become available.
- **Block (Purge):** Stops this job immediately, but sets the job state to "blocked". The job will not run again until it is manually unblocked.
- **Block (Finish Current):** Stops this job after the currently rendering frame(s) complete. The job will remain in a "blocked" state until it is manually unblocked.
- **Unblock:** Remove the "blocked" state from the job. All frames that have not completed will go back to a "pending" state. Those that have completed or failed will remain completed or failed.
- **Kill:** Stops this job immediately and sets the state to "killed". A job that is killed cannot be continued - it can only be retried, resubmitted, or removed.
- **Remove:** Stops the job immediately (if running), and permanently removes the job from the qube database. This cannot be undone.
- **A/B Wipe in RV:** An example of using ArtistView's plugin architecture to perform actions outside of Qube. When multiple jobs are selected, an RV session will be opened that does a wipe between each job.
- **Browse Output Dirs:** Open a file browser for each output dir in this job.

- **Worker**

- **Lock Host (Purge):** Stop all jobs running on this worker immediately, then leave the worker in a "locked" state which will prevent it from accepting any more work until it is manually unlocked.
- **Lock Host (Finish Current):** Same as above, but the currently rendering frames are allowed to finish.
- **Unlock Host:** Remove a worker lock.

- **Frame**

- **Retry:** Rerender the selected frame(s)
- **Block:** Set the frame's state to "blocked". This will prevent the frame from being rendering until it (or the job) is manually unblocked.
- **Unblock:** Remove the block from the frame, setting it back to a pending state.
- **Kill:** Stop this from immediately and set the state to "killed".
- **Complete:** Change the state of this frame to "complete," even if it has not yet run.

- **Instance**

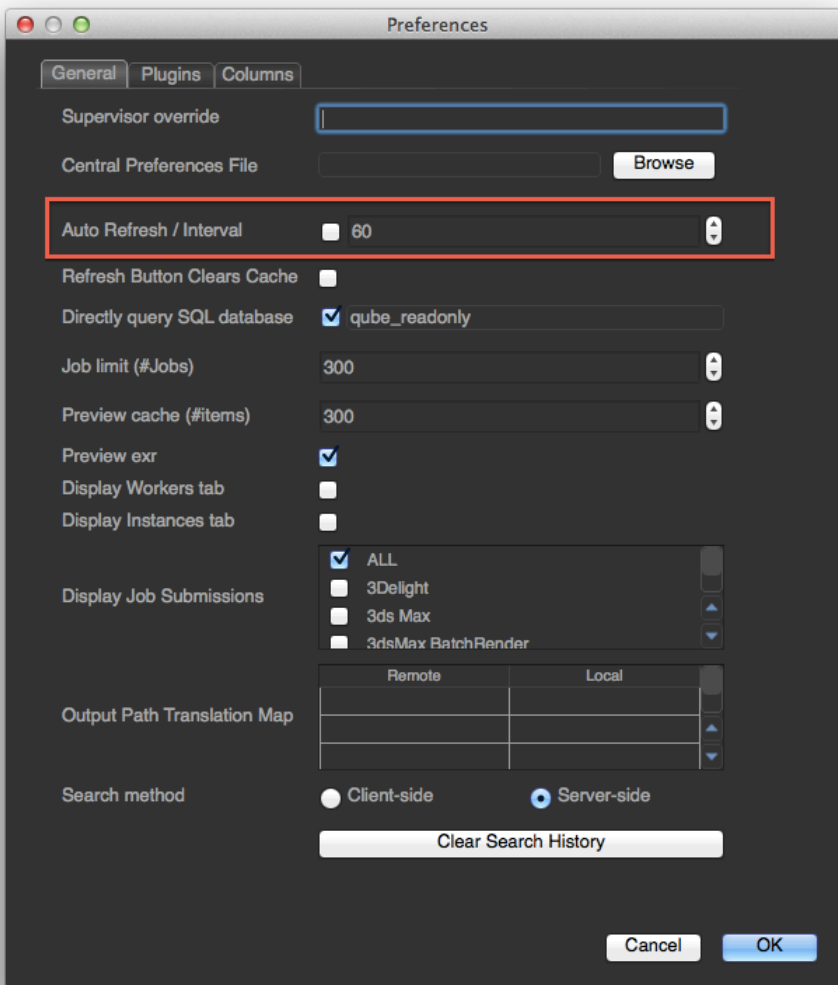
- **Migrate:** Stop this instance and set its state back to "pending". Default supervisor settings dictate that the next time this instance starts, it will start on a different machine.
- **Retry:** Restart this instance, regardless of its current state.
- **Complete:** Mark this instance as complete.

- **Interrupt:** Stop this instance from processing any more work, but set its state back to "pending". As with a job's "interrupt" this is typically done to manually free up resources that could be used by higher priority jobs.
- **Block:** Stop this instance from processing work and set its state to "blocked". This instance will not process any more work until it is manually unblocked or the job is retried.
- **Unblock:** Remove a block from an instance.

Watching Progress

Once a job is running, if it takes more than a few seconds (and most do) then you will want to keep an eye on progress. By default, the UI does not refresh itself, meaning that as the job progresses, nothing will change in the interface. You can force an update by simply clicking on the job. You can also refresh all jobs or frames or instances or hosts by clicking the "Refresh" button associated with their list.

You can also configure Qube to automatically update the UI every so often. To do this, go to Edit in the top menu and choose Preferences. Click on Auto Refresh / Interval. You can leave the refresh interval at 60 seconds, or make it shorter or longer.



Note that asking the Supervisor for an update every few seconds puts more load on it, and if there are many artists all doing the same thing, you could be slowing down the overall farm just for the sake of convenience. Think twice before asking for lots of updates you aren't going to be noticing anyway.

Looking at Job Details

The Job Details panel has a number of tabs. The first one, Job Properties, lists a number of attributes of the job, including the frame range and the command and/or scene file used.

The next two are useful for finding out what went wrong with jobs that fail (or getting logging information from jobs that succeed). Most output will go to the first tab, Output Log. Some programs will send error messages and other information to a separate stream, which goes to the Error Log. Scroll to the bottom of either of these - error messages are coded in red, warnings in yellow and info (as determined by the program being run) in green. All of the tabs and their uses are explained in more detail in the [User's Guide](#).

Submitting New Jobs

The Artist View UI can be used to submit jobs to Qube, through the Submit menu item at the top. However, in most cases you will be setting Qube up to allow submission directly from your application, such as Maya or 3dsMax. Submitting directly from applications was covered in the previous section. However, you can still use the UI to submit test jobs, to make sure that things are still working as expected, and to submit jobs for applications where you haven't installed the inApp interface.

The Frames Panel

The right side of the Artist View UI contains the Frames panel, which lists information about all the frames in the selected job. Notice that it has a separate refresh button, which allows you to query the status of just those frames. You can also right-mouse on any of the frames and kill them, retry them, suspend (block) them (and then unblock them), as well as marking them complete.