Installing and Licensing 3ds Max on a Worker Node

Components to install

You should **perform a full installation** of 3ds Max (including backburner) on all worker nodes that will be rendering Max jobs. **Do not install a trial version**.

Licensing

Because Qube uses 3ds Max in slave mode, it does not **consume** a Max license while rendering.

3ds Max requires a license number of some sort in order to properly install.

Install as a network license, providing a "bogus" serial number such as 111-1111111.

Initialize 3ds Max for stand-alone rendering immediately after the installation is complete

Start the BackBurner server in order to setup 3ds Max for "headless" rendering, then exit it. It should never have to be started again. Also ensure that the BackBurner service is **NOT** set to start automatically; it shouldn't be running at the same time as the Qube worker service. **This step must be performed by an adminstrator,** but you had to be an admin in order to install 3ds Max, so if you do this immediately after the install completes, you are an admin.

Initialize the 3ds Max user directories (prefs, etc)

In order to create the user prefs and directories necessary for 3ds Max to run, it must be started at least once as the user account who will actually be running the job on the worker. Which account this is depends on whether the worker is running in service or "Desktop User" mode. See Service Mode vs. Desktop User Mode for more information.

Service mode: you should log in as the account specified in proxy_account; if you're unsure what this is or you haven't set it, the default user is "qubeproxy", password: Pip3lin3P@$$wd - see How do I login to the local "qubeproxy" account on a Worker?

Desktop User mode: simply log in as the user who has been selected to run as the Desktop Worker.

Once logged in as the user, start 3ds Max, let the interface launch, then exit.

Testing the installation

Send the **absolutely simplest scene file** you can to the 3ds Max workers:

- 1 camera
- 1 light
- an unshaded sphere
- "render to file" is enabled, render directory must be on a network location

You're trying to test if Max opens, loads the scene, renders an image, and exits, not if it can render the most heinously complicated scene you've got in your arsenal.