

Installation

Before installing, select a Tractor Engine host

Select a host to run the Tractor engine. This host will need to run on Linux or macOS; the supported versions are listed on the RenderMan Products download page. Blade hosts can be Linux, macOS, or Windows.

Hosts with faster CPUs, more cores, and faster storage will be able to schedule faster and handle more throughput.

Larger disks tend to be important as more completed jobs can be saved for analysis.

Modest commodity drives are sufficient for large queues; 1 GB will store approximately 200,000 tasks. However, larger drives may be desirable if jobs will be archived in perpetuity for historical analysis. Also, disk space is required to store log files under the default configuration.

Virtual machines are not usually recommended for running tractor-engine unless care is taken to allocate CPU, disk, and network resources with nearly "bare metal" performance.

For simplicity, the engine host should not be running a web server or other server using port 80. The engine can be configured to use a different port, but it **greatly** simplifies configuration and web-browser UI access to use the default 80.

Download and run the installer on the Tractor Engine host

To download the tractor installer(s) login to <https://renderman.pixar.com/forum/> and click the blue "Downloads" link at the top of the page, then select the Tractor entry from the list to see the available downloads.

You will need the **full installer in all cases** to install Tractor Engine, plus it includes all other tools too.

Note: For other roles there are also several much smaller packages containing the Python-only engine-client components that can be **downloaded separately** if desired. These include python "wheels" for tractor-blade and the scripting APIs, suitable for "pip" installing one or both of those components into a Python virtual environment (for example) on a given host or container. There is also a python "zipapp" that contains the collection of Python clients that connect to the engine. The zipapp can be used in stand-alone mode without changing an existing local stock Python. **See the README in the "clients" download archive for details** on these Python sub-packages. Using these lightweight python-only packages are optional for these other non-engine host types - the full installer can be used there too.

The full installer (rpm, msi, or pkg) must be run with system administrator privileges (e.g. using 'sudo' on Linux). The installation directory is:

- /opt/pixar/Tractor-VERSION on Linux,
- /Applications/Pixar/Tractor-VERSION on macOS, and
- C:\Program Files\Pixar\Tractor-VERSION on Windows.

This path will be referred to as INSTALL_DIR in this documentation.

NOTE:

- The full installer contains **all** Tractor components, but **only a subset** is needed on each type of machine: the engine host, render nodes, and user workstations. See the note about the much smaller Python-only client downloads that may be more appropriate in some cases.
- The (one) central tractor-engine service requires much more of the full install than any other component.
- It is possible to install the full package and the smaller python packages onto a fileserver where many hosts and users can access them. But as above, only a subset of the pieces are needed on each machine type.

Get a license.

Near the end of the full installation process, the LicenseApp program will be launched to retrieve your license file. The license is **only needed on the engine host**, and only if you have not installed one previously. If you already have a license, or if you are just installing on blade hosts, then you can simply **dismiss the license dialog** using the close button. If you do need to download your license, then you should enter your RenderMan forum login, password, and the serial number that you received at purchase time; then the application will connect to Pixar and retrieve the license.

If you need to download a license file after the engine has been installed, simply run the LicenseApp application, which is located in INSTALL_DIR/bin.

By default the license file pixar.license will be downloaded to the parent of the installation directory (/opt/pixar or /Applications/Pixar). Alternatively, the license location can be overridden by setting the PIXAR_LICENSE_FILE environment variable or by setting LicenseLocation in your tractor.config file. Because Tractor is typically run in combination with a floating license server, the pixar.license file will be a "stub" or "redirector" license, which simply points at the license server machine.

The engine manages the available Tractor seats; the total seat count represents the maximum number of tasks that can run concurrently across the farm. Blades do not require a license to start up, and do not consume seats while idle. The Dashboard does not consume a license.

Make "tractor-engine" a hostname alias

The dashboard, blade, and command line tools need to be able to find the engine in order to communicate with it. For this reason it is useful to create a hostname alias called "tractor-engine" - assigned to the real hostname of the engine host - especially in production environments. All of those tools will attempt to connect to "tractor-engine" by default, so it greatly simplifies day to day usage to use that name rather than applying overrides. The engine can optionally advertise its network address using a simple multicast protocol that is suitable for some **small** installations. (For more information, see [engine discovery](#).)

Using the "tractor-engine" alias also allows a new underlying host to be used as the engine host, just by switching the alias, rather than updating override options in all tool use cases. Please consult with your system administrator regarding the best method for creating the hostname alias *tractor-engine* to redirect connections to your selected engine host. This may involve:

- adding a DNS CNAME record to point "tractor-engine" to the engine hostname at your site, or
- adding a 'cn' entry to the engine hostname's LDAP entry, or
- adding a hostname alias to /etc/hosts (not scalable)

Several command-line tools can be used on user/blade hosts to verify that they can reach the engine host itself using the hostname alias:

```
host tractor-engine
nslookup tractor-engine
ping tractor-engine
```

After the engine is running (below) then web browsers can also test the tractor-engine alias by browsing to that hostname.

Command-line components such as tractor-blade, tractor-spool, and tq can point to a different engine by setting the TRACTOR_ENGINE environment variable or by adding the option `--engine=HOSTNAME:PORT` to match your engine installation. References to tractor-engine in the configuration files may also need to be changed.

Install blade services

Each machine that will execute Tractor tasks, typically render farm nodes, must be running the tractor-blade python program. See [Setting Up Services](#) for the specifics.