

# RMS



This is legacy documentation for RenderMan 21 and prior.

## Introduction

The default configuration of RenderMan Studio distributes work via Tractor. Straight out of the box a user can go to the Render Options window and set things up such that any render that isn't a local internal render will use Tractor to make sure the job gets done right. That is, of course, assuming that your site happens to be using the default, straight-out-of-the-box configuration of Tractor, as well.

If, perchance, you think you're special and opted to change the default configuration, fear not. RenderMan Studio provides all the hooks you need, via preferences and other doodads, to maintain a harmonious symbiosis between your RenderMan Studio workstations and your Tractor work dispatching system.

## Engine Location and Tractor Users

The `RenderMan_for_Maya.ini` provides two critical preferences for sealing RMS and Tractor's bond:

```
SetPref TractorEngine {tractor-engine:80}
SetPref TractorUser {}
```

These preferences are used by the MEL script that generates job scripts and spools the jobs. As with all prefs, they can be set in a `RMS_SCRIPT_PATHS`-based ini override file.

By default, Tractor spools to a machine called *tractor-engine*, on port 80. For any of a number of reasons, this might not be your setup, but you can override this pref in your site-specific ini, for example, like so:

```
SetPref TractorEngine { myserver:8080 }
```

The format is: engine:port.

The default for `tractor-spool.py` (`tractor.py` in the RMS distribution) is to spool as the user spooling the job. You may want the jobs spooled as a specific user. The second preference enables locking down the jobs to a specific user, e.g.:

```
SetPref TractorUser { mwazowski }
```

## EnvKeys

The `RenderMan_for_Maya_Pro.ini` file now has an additional pref that specifies the default environment key that is passed to Tractor.

```
SetPref DefaultEnvKey "rms-[GetPlatform rms_version]-maya-[GetPlatform rfm_mayaversion]"
```

This *envkey* is handled by the Tractor *envhandler*, which parses the key and properly sets up the `$RMSTREE` and `$MAYA_LOCATION` on the tractor-blade. (It's important to note, however, that the *envhandler* for RenderMan Studio sets up the environment based upon "default" installation location, which may be different at different sites.)

The `DefaultEnvKey` preference sets the default for the Render Options (both *Current Frame* and *Batch*) in RMS. In this dialog, the user can completely change the environment settings that are sent to Tractor, allowing users to append other envkeys to the default. Note that the resulting envkey then gets saved into a Maya preference, like all the other values in the batch render dialog, and is propagated to future Maya scenes. Clicking on the *Default* button for the Environment Key in the Render Options window restores the `DefaultEnvKey` from the `.ini` file.

## tractor-blade Configuration

If you have changed the default install location for RenderMan Studio you will need to edit the blade configuration to change the install location, so that RMS jobs will be processed by Tractor correctly. In particular, there are three shared `*.envkeys` files that are sent from the tractor-engine to the tractor-blades. These envkey files contain environment dictionaries. The environment dictionary for the envkey "rms-\*-maya-\*" will need to be altered to point to the correct install location.

- **Important**

*As with .ini files, we recommend that users don't edit the factory config files. Instead, copy the files to a special config directory for your site and edit them there. Then, start the tractor-engine with the `--configdir=/path/to/config/files` option.*

## Additional envhandlers

Several other envhandlers are available, and probably most used is the *prman* envkey. The *prman* key is of the format:

```
"prman-XX.X"
```

The XX.X refers to the version (example '19.0') of prman that is to be launched with the batch rendering. When this key is added to the batch dialog, the *rm anhandler* is invoked. It will locate the intended prman release on disk, set RMANTREE in the launch environment, and add \$RMANTREE/bin to the launch environment path.

If RMS or RPS are installed in non-standard locations the *rmantree* and *rmstree* handlers can be used instead.

```
"rmantree=/path/to/my/rmantree/installation"
```

```
"rmstree=/path/to/my/rmstree/installation"
```

These handlers pull the path to the *rmantree* and *rmstree* out and add them into the environment as \$RMANTREE and \$RMSTREE. They also append \$RMANTREE/bin and \$RMSTREE/bin to the path of the launched commands.