

Enhanced Balancing Logic/Weighted Job Scheduling

An experimental Enhanced Balancing Logic option has been added for balanced/weighted scheduling options in the Job Scheduling settings. When this option is enabled, the Slaves will use the new SlaveJobState collection in the database to get a more accurate snapshot of all the rendering jobs in the farm, and use this information to make better decisions about which job they should be rendering.

Testing has shown that when this option is enabled, a proper distribution of Slaves among jobs is much more consistent, and Slaves no longer jump between jobs of the same priority. The result is more predictable behavior, and less wasted time due to the overhead of switching between jobs that are expensive to start up.

New Slave Metrics

Slaves now report their Network I/O, Disk I/O, and Swap usage, which can be viewed from the Monitor. This information is also stored in the statistics that are gathered for the Slaves.

In addition, Swap usage for the rendering process is stored with a job's task when it completes, and is also stored in the statistics for the job when it completes.

Improved Slaves Statistics Reports

The Slave Resource Usage farm report is now called the Slaves Overview farm report, and shows additional statistics. For example, the new Slaves Overview chart shows how many slaves were in each state (starting job, rendering, idle, offline, stalled, and disabled). In addition, the new Available/Active Slaves charts show the number of slaves that are available, and the number of available slaves that are active. Finally, the new Plugin Usage chart shows the overall usage of the render plugins.

Both the Slaves Over and Active Slaves Stats reports can also be shown for a given region. This allows you see statistics for slaves in a specific Cloud region, or in specific areas in the office (ie: render nodes versus workstations). Note that this requires you to set which regions your slaves belong to in their Slave Settings.

Improved Graphs in the Monitor

Line and Bar graphs in the Monitor now support panning and zooming, and a right-click option has been added to reset the zoom level. In addition, individual series in some Line graphs can be shown/hidden from the right-click menu. Finally, the axis labels in these graphs have been updated to properly represent integer and date/time values, which makes them easier to read.

Expanded Font Synchronization

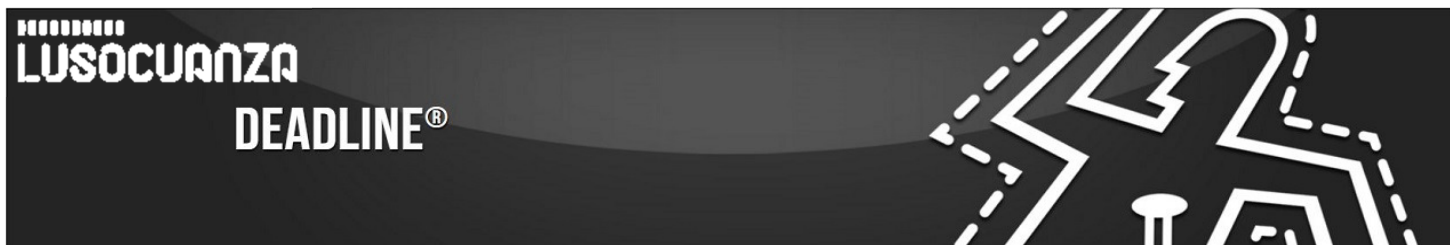
The new FontSync event plugin that ships with Deadline can be used to synchronize fonts on Mac OS X and Windows before the Slave application starts rendering any job, or when the Slave first starts up. This general FontSync event plugin replaces the font synchronization options in the After Effects plugin and now works for ALL plugin types in Deadline.

Improved Job Batch Display

Deadline introduced the ability to group jobs together in the Monitor by setting their Batch Name property. Now, all Deadline submitters automatically set the Batch Name if multiple related jobs are being submitted at the same time. For example, when submitting each render layer as a separate job in Maya, they will all be part of the same batch. Another example is submitting a Jigsaw render with a dependent assembly job.

In addition, the Batch Row in the job list in the Monitor now shows information for all columns, depending on the settings for the jobs in the batch. For numeric settings like priority or machine limit, the largest value for the jobs is shown. For settings like pool and group, the value will be shown if all jobs have the same value, and if they don't, <batch> is shown instead. For all other columns, <batch> is simply shown.





Finally, the counts above the job list in the job panel now show the number of batches in the list, and the selected count now ignores selected batches so that it properly represents the number of selected jobs.

Protected Jobs

Jobs now have a “Protected” property. When enabled, the job can only be deleted by the job’s user, a super user, or a user that belongs to a user group that has permissions to handle protected jobs. Other users will not be able to delete the job, and the job will also not be cleaned up by Deadline’s automatic house cleaning. This is useful if you have jobs you need to keep around for testing or benchmark purposes.

Flexible Image Viewer Configuration

The Monitor has always had the option to specify up to three viewers to use when viewing images from the Task list. Now, optional command line arguments can be set, which are then passed to the viewer when viewing images. There are also special tags, which are replaced automatically with information about the image being viewed.

The following tags are supported in the command line options:

- {FRAME}: This represents the task’s frame file. For example: /path/to/image0002.png
- {SEQ#}: This represents the task’s frame sequence files, using ‘#’ as the padding. For example: /path/to/image####.png
- {SEQ?}: This represents the task’s frame sequence files, using ‘?’ as the padding. For example: /path/to/image????.png
- {SEQ@}: This represents the task’s frame sequence files, using ‘@’ as the padding. For example: /path/to/image@@@@.png
- {SEQ%}: This represents the task’s frame sequence files, using ‘%d’ as the padding. For example: /path/to/image%04d.png

The arguments default to “{FRAME}”, which keeps the default behavior from previous versions of Deadline intact.

In addition, proper names can be given to the viewers, which are shown in their corresponding menu items. Finally, viewers can be configured to support chunked tasks (tasks which consist of more than one frame).

Standalone Web Service application

A standalone Web Service application is now shipped with Deadline, and is called deadlineWebService.exe. It works exactly the same as the Web Service feature that is built into Deadline Pulse, and both can be configured using the new Web Service page in the Repository Options.

Install Launcher as Daemon on Mac OS X

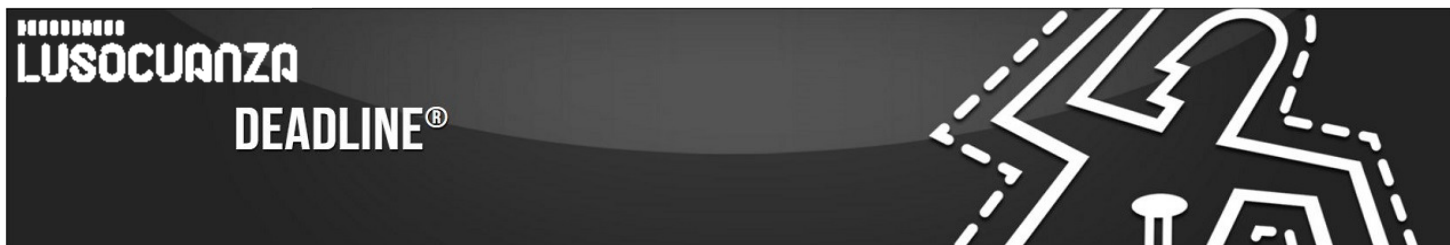
The Deadline Client installer now has an option to install the Launcher as a Daemon on Mac OS X. This feature lets you run the Launcher daemon as root, or as another user account.

Improved Submission Script Installers

The submission script installers now show what DEADLINE_PATH is set to (which is used by the submission scripts to determine where the Deadline Client’s bin folder is located). You then have the option to change it if it’s incorrect, or set it if it doesn’t exist. This is useful if you have multiple versions of Deadline installed on your system.

A side-effect of this improvement is that it allows you to update DEADLINE_PATH without having to reinstall the Deadline Client or manually changing your system’s environment. To do this, simply run any submission script installer, change the DEADLINE_PATH value, and uncheck all options listed in the Components list. The installer will then update the DEADLINE_PATH without installing the submission script files.





Draft Updated to Version 1.3.2.58232

This version of Draft requires a new Draft 1.3 license, and includes the following updates:

EXR Images:

- Added support for EXR data and display windows (previously data windows were set to the same size as the display windows).
- Updated to OpenEXR 2.2.0.

LUT support:

- Added ACES 1.0 LUTs to the included ocio-configs folder.
- Improved the robustness of the Draft ASCCDL Reader. The reader can now handle different syntax in its input file.

VideoEncoder:

- Fixed a bug when encoding an image with VideoEncoder. The VideoEncoder was applying a bit of scaling to the image.
- Fixed bug on Mac OS X where encoding with certain dimensions (ie: 640 x 480) was causing a memory error crash.

Draft Tile Assembler:

- Added support for assembling big images by exposing a new class in Python called TileAssembler. Most of the logic of an assembly job can now be handled internally

New Application Support

Support has been added for many AEC (Architecture Engineering Construction) and Product Visualization products, including AutoCAD, CSiBridge, CSiETABS, CSiSAFE, CSiSAP2000, EnergyPlus, MicroStation, VRED and VREDCluster.

Support has also been added for 3ds Max 2016, Anime Studio 11, Composite 2016, Corona distributed rendering, Maya 2016, Media Encoder, Houdini 14, LuxSlave distributed rendering, Natron, Nuke Studio Frame Server distributed rendering, Renderman RIS for Maya, VRay for modo, and Vue 2015.

VMX (Virtual Machine Extension)

With VMX (Virtual Machine eXtension) built in and pluggable cloud support, Deadline 7 can interact with private and public cloud solutions out-of-the-box, including Amazon EC2, Microsoft Azure and OpenStack, among others. The new Deadline Balancer application can start and shut down virtual instances on demand based on the jobs in the queue, the current budget settings, or other custom algorithms. Multiple cloud solutions can be used simultaneously, along with classic non-cloud rendernode and workstation rendering.

PYTHON Upgraded to 2.7.8

Deadline now ships with Python 2.7.8. Note that this shouldn't affect any existing scripts that you use with Deadline.

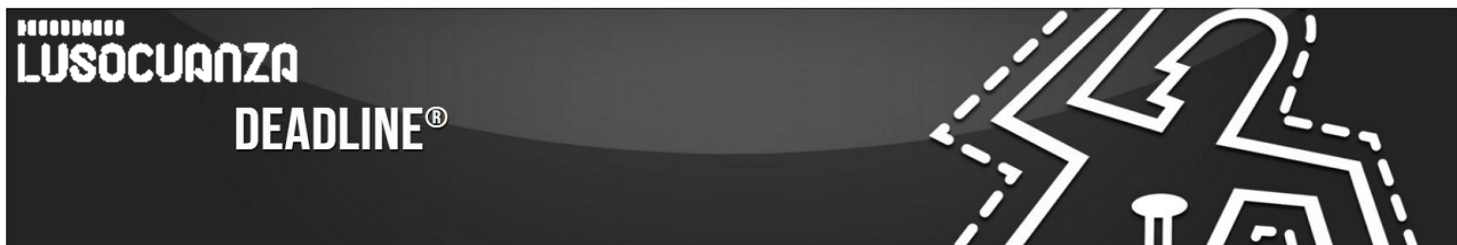
In addition, the Deadline applications no longer set the PYTHONHOME and PYTHONPATH environment variables for their current session. This means that any applications launched from a Deadline application will no longer inherit these modified variables, which should avoid compatibility issues if those other applications use a different version of Python.

MONO Upgraded to 3.8

Deadline now runs against Mono 3.8 on Linux and Mac OSX, which helps improve stability. In addition, the Mac OSX version of Mono is now 64-bit. This new version is bundled with the Linux and Mac OSX Client and Repository installers.

Mono is now installed automatically as part of the installation procedure on Linux. It is installed to the Deadline installation folder, and won't impact any existing Mono installations. Now Mono no longer needs to be installed manually on Linux prior to installing Deadline.





Updated Slave Licesing Model

When running multiple slaves on a single machine, they will now share a single license instead of needing one license per slave instance. In addition, the slaves will only hold onto their license while they are rendering. When they become idle, they will return their license.

Customizables Styles for Deadline Applications

The new Styles configuration panel in the Monitor options allows you to customize the color of the Deadline applications. Simply specify a palette color and the User Interface will automatically use lighter and darker variants of that color where necessary. In addition, the font style and size can be configured as well. Finally, you can export styles and share them with other users.

A new Batch property has been added to jobs that allows jobs to be grouped together in the Job List. All jobs with the same Batch name will be grouped under that Batch name, and the Batch name can be expanded or collapsed to show and hide all the jobs, respectively. Jobs in the same Batch will also be grouped together in the Job Dependency View. Finally, the properties for the jobs in the same Batch can be modified by simply right-clicking on the Batch item in the Job List or the Job Dependency View.

New graphs have been added to the Monitor. The Jobs panel can show pie charts based on the job pool, secondary pool, group, user, and plugin. The Tasks panel can show graphs representing the task render times, image sizes, cpu usage, and memory usage. The Slaves panel can now show bar charts that show how many slaves are in certain pools and groups. The Job Reports panel can now show a pie chart that shows the percentage of errors generated by each slave.

A default layout for panels in the Monitor can now be saved, and when a new panel is opened, it will use the saved default layout. So now you can set up your favourite default layouts for the Job list, Task list, etc and not have to worry about setting them up again when you open new panels.

In addition, you can now save the layout from a panel to disk and load it in again. This allows you to share a layout from your Monitor with someone else.

Job Dependency and Limit Improvements

Job dependencies are now more flexible than ever. Individual dependencies can have notes attached to them, and they can also have their own overrides for the Frame Offset and Resume On... settings.

The Job Dependency view in the Monitor has also been updated to show these per-dependency settings. In addition, there is now a new feature in the Dependency View that allows you to test the dependencies and see which ones pass and which ones do not. Finally, the look of the nodes in the Dependency View have been updated.

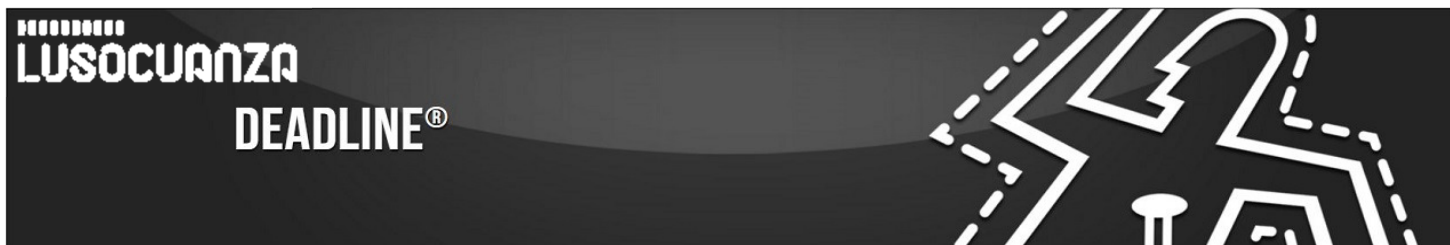
Limits are now much more flexible than they were before. Previously, one Limit Stub per Slave was used up when a Slave rendered a job that required that Limit. This is still supported, but now, a Limit can be configured so that one Limit Stub per Task is used up, or one Limit Stub per Machine is used up.

The per Task option is useful if you are rendering with an application that requires one license per instance, and you are rendering more than one concurrent task at a time. The per Machine option is useful if you are rendering with an application that only requires a single license per machine, regardless of how many instances are running on that machine.

Improvements to Pool and Group Management

The Slave list in the Pool and Group Management dialogs can now be filtered, and all columns in the list are now available. In addition, you can now right-click on specific slaves in the Slave list in the Monitor to modify Pools and Groups for the selected slaves only.





Deadline now supports the ability to suspend and resume individual tasks. This can be useful if you want to postpone or skip the rendering of specific tasks.

Deadline's Slave Scheduling feature has undergone a major overhaul. Previously it was part of Power Management and controlled by Pulse, but now it is a standalone feature that is controlled by the Launcher application that runs on every Client machine. This means that Pulse is no longer required to use the Slave Scheduling feature.

There are also new features that have been added to Slave Scheduling. If a slave is scheduled to start on a machine, a notification message will now pop up for 30 seconds indicating that the slave is scheduled to start. If someone is still using the machine, they can choose to delay the start of the slave for a certain amount of time. Another addition is the new option to enforce the slave schedule. If enabled, the Launcher will keep restarting the slave if it is shut down during a period of time that it is supposed to be running.

Finally, Slave Scheduling can now be configured to launch the slave if the machine has been idle for a certain amount of time ("idle" means no keyboard or mouse input). There is also additional criteria that can be checked before launching the slave, including the machine's current memory and CPU usage, the current logged in user, and the processes currently running on the machine. Finally, this system can stop the slave automatically when the machine is no longer idle.

Note that Idle Detection can be set in the Slave Scheduling settings, or on a per-slave basis in the Slave Settings dialog in the Monitor. It can also be set in the new Local Slave Control dialog so that users can configure if their local slave should launch when the machine becomes idle.

Job Dequeuing Mode

Slaves now have a new Job Dequeuing mode that controls which jobs a slave dequeues based on how the job was submitted. By default, a slave will dequeue any job, but it can be configured to only dequeue jobs submitted from the same machine that the slave is running on, or submitted by specific users.

The Job Dequeuing Mode can be configured in the Slave Settings dialog in the Monitor. It can also be set in the new Local Slave Control dialog so that users can configure if their local slave should only render their own jobs, or if they want to help another user render their jobs.

Local Slave Controls

The Monitor and Launcher applications now have a new dialog that can be used to control the slave running on the local machine. It can be used to start and stop the slave, or connect to the slave's log. This is useful if the slave is running as a service on the machine.

In addition, you can set up the slave to launch if the machine has been idle for a certain amount of time ("idle" means no keyboard or mouse input). It can also stop the slave automatically when the machine is no longer idle.

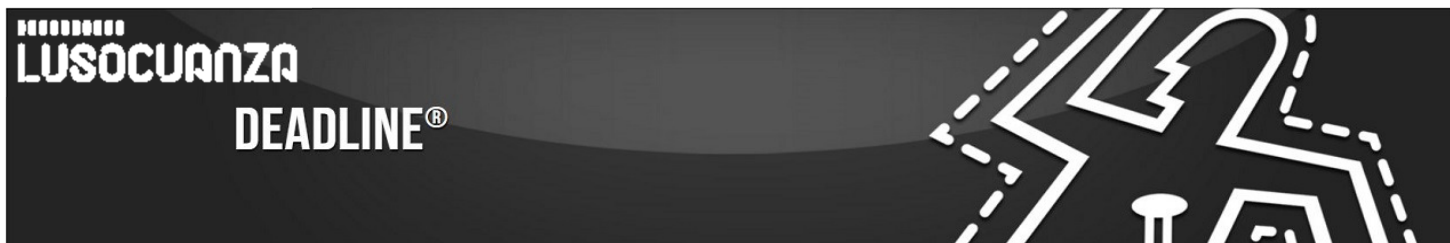
Finally, the slave's Job Dequeuing Mode can be configured here. By default, a slave will dequeue any job, but it can be configured to only dequeue jobs submitted from the same machine, or submitted by specific users. This is useful if a user wants their slave to only render their jobs, or they want to help another user render their jobs.

Note that the Idle Detection and Job Dequeuing Mode settings can also be changed by administrators for all slaves. In addition, the Local Slave Controls feature can be disabled by administrators if they don't want users to be able to control their local slaves.

Render as User

A new option has been added to Deadline to render jobs with the account that is associated with the job's user. The account information can be configured in the Deadline user settings. On Windows, the user's login name, domain, and password are required. On Linux and Mac OSX, just the user's login name is required, but the Slave must run as root so that the Slave has permission to launch the rendering process as another user.





Additional statistical information is now gathered for individual slaves, including the slave's running time, rendering time, and idle time. It also includes information about the number of tasks the slave has completed, the number of errors it has reported, and its average Memory and CPU usage. Like job statistics, Pulse does not need to be running to gather this information.

Pulse Redundancy

You can run now multiple instances of Pulse on separate machines as backups in case your Primary Pulse instance goes down. If the Primary Pulse goes offline or becomes stalled, Deadline's Repository Repair operation can elect another running instance of Pulse as the Primary, and the Slaves will automatically connect to the new Primary instance.

Note that when multiple Pulse instances are running, only the Primary Pulse is used by the Slaves for Throttling. In addition, only Primary Pulse is used to perform Housecleaning, Power Management, and Statistics Gathering. However, you can connect to any Pulse instance to use the Web Service.

New events and Asynchronous Job Events

New events have been added to the Event Plugin system. The first is the OnHouseCleaning event, which triggers whenever Deadline performs Housecleaning. This allows you to set up event plugins to do custom cron-job style operations within Deadline.

In addition, there are four new events that trigger when a slave changes state: OnSlaveStarted, OnSlaveStopped, OnSlaveRendering, and OnSlaveStartingJob. As an example, an event plugin could be written to have slaves automatically add themselves to Groups when they start up based on some custom criteria, or an event plugin could be written to have slaves perform maintenance checks when they become idle.

Finally, there is now an option to process many types of job events asynchronously. The benefit is that job events will no longer slow down batch operations in the Monitor (for example, deleting 1000 jobs will be much faster if you are using event plugins because those events will be processed later).

These job events are queued up in the Database and Deadline's Pending Job Scan will process them at regular intervals. Because they are placed in a queue, they will still be processed in the same order that they were triggered. Note that if this option is enabled, some events are still processed synchronously, like the OnJobSubmitted and OnJobStarted events.

Auto Configuration Overhaul

The Auto Configuration feature has undergone a couple of significant changes. The first is that all Deadline applications can now pull the Auto Configuration settings, instead of just the Slave. This means that Auto Configuration can now be used to automatically configure workstations, not just render nodes.

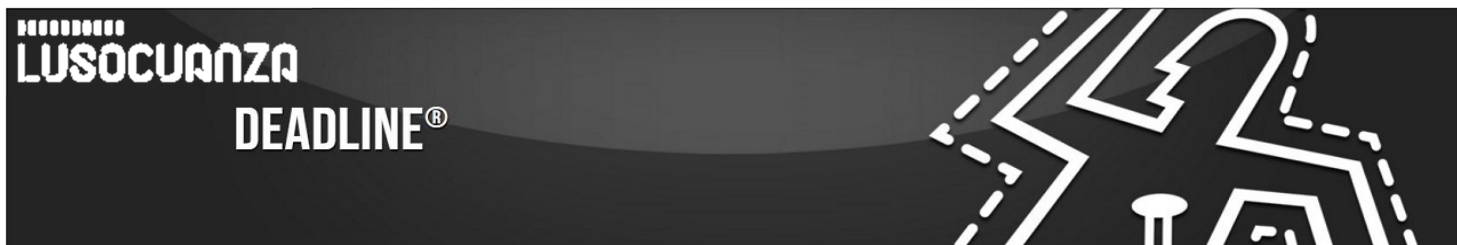
The second change is with how Auto Configuration works. Previously, all Auto Configuration settings were pulled from Pulse. Now, only the Repository Path is pulled from Pulse, and the other settings are pulled when the Deadline application connects to the Repository. The benefit to this is that most of the Auto Configuration settings will work without Pulse running.

Finally, Auto Configuration rule sets can now be enabled or disabled, so you no longer have to delete a rule set if you want to remove it temporarily.

Region Awareness

Regions can now be configured in Deadline, and users and slaves can be assigned to a specific region. Currently, this is useful for Path Mapping, and allows you to map paths differently based on the region that the users or slaves are in. Note that when VMX launches a slave, it will automatically be added to the region associated with the cloud provider settings.





Grid-Based Script Dialog

New grid-based functions have been added to the DeadlineScriptDialog class which makes it easier to create custom dialogs. Instead of setting the width and height when adding new controls to a row, you can instead add them to a grid and indicate which row and column the control should go in. Optionally, you can also indicate how many rows and columns the control should occupy. By being part of a grid, the controls will now grow and shrink dynamically based on the size of the dialog and the size of the font.

FTrack Integration

The Deadline/FTrack integration enables a seamless render and review data flow. When Deadline starts a render, an Asset Version is automatically created within FTrack using key metadata. When the render is complete, Deadline automatically updates the created Version appropriately – a thumbnail image is uploaded, components are created from the Job's output paths (taking advantage of FTrack's location plugins), and the Version is flagged for Review. In doing so, Deadline provides a seamless transition from Job Submission to Review process, without artists needing to monitor their renders.

JIGSAW for 3ds Max, Maya, MODO and Rhinoceros

Jigsaw, which was previously only available for 3ds Max, is now available for Maya, modo, and Rhino. It gives you more control over the tiles and/or regions that you are submitting to Deadline. This feature uses Thinkbox Software's Draft library to assemble the final image instead of the old TileAssembler.exe application. Note that Draft requires a license, so contact Thinkbox Sales if you don't already have a Draft license.

Submission Script Installers

Submission script installers can now be found in each application folder in the Submission folder in the Repository. These allow for most of the submission scripts to be installed automatically, instead of having to manually copy over files.

Support for Salt and Puppet

Application and Event plugins have been added to support the Salt and Puppet automation applications. Jobs can be submitted to the application plugin to update software and machine configurations on specific machines, while the event plugins can be used to update all of your machines when the slave running on them becomes idle.

Support Updated Application

Support has been added for After Effects CC 2014, Arnold for Houdini, Cinema 4D 16, Corona, Fusion 7, Nuke 9, Realflow 2014, and SketchUp 2015.

