

RenderPal V2

The professional Render Management System

Point Release 5

Release notes

RenderPal V2 Version: 2.5

RenderPal V2 Copyright © 2008-2011 Shoran Software
All rights reserved.



What's new in RenderPal V2 Version 2.5

Renderer license management

Most renderers and compositing applications use some form of license management, usually in the form of node-locked licenses, where licenses are given to specific machines in a render farm, or in the form of a floating license, which only limits the number of concurrent machines at a given time.

RenderPal V2 2.5 now offers a way to easily reflect your renderer licenses: For every renderer and its versions, you can now specify how many licenses are available and which pools and clients own a license. Using the new license management, you can avoid to have new jobs sent to clients even if there currently is no free license slot available, eliminating common license problems.

Licenses can be defined separately for specific versions or an entire renderer, making the license system extremely flexible. For each license, you can define if you have a limited license count, and you can also select specific pools and clients which have access to a license of that renderer. When dispatching its jobs, RenderPal will then take into account these license settings and will only send jobs to clients if there is a free license slot left and if the client actually has access to a license.

User groups

In RenderPal V2 2.5, it is no longer only possible to create individual user accounts, but to also create entire user groups. Every account can now belong to such a group, making creating and managing accounts faster and easier. A group defines the user rights, pool access and various options for its subordinate user accounts; by default, an account belonging to this group will then use these settings, but it is also possible to override them in the account.

We also made the entire user account management more comfortable and easier to use, which especially helps when dealing with many accounts (and groups).

Data queries

The console remote controller can now be used to query data from the server, including net jobs and their chunks, pools and clients. This data, which comes in the XML format, can then be used, for example, to put statistics about your render farm on a website or to evaluate rendering times to calculate the costs of a job. Previously, if you wanted to access data from the server, you had to directly query its database, which was rather cumbersome. This is no longer necessary - simply use the console remote controller to fetch what you need!

This feature is especially useful for companies offering rendering farm services and web integration.

Net Job tagging & coloring

While the view filters are already a very handy feature when dealing with net jobs, this new version introduces yet another helpful tool for quickly filtering net jobs: Net Job tags. Each net job can have several user-defined tags (like a project or shot name, version information and so on). The net job list can then be quickly filtered by those tags: Only net jobs which have one of the entered tags will be shown (view filters will be respected as well, of course). Tags make working with many net jobs even easier than before.

Beside these handy tags, it is now also possible to give each net job a custom color, which makes spotting specific net jobs in a large net job list much easier.

Restarting chunks on low CPU usage

Renderers tend to crash from time to time. When such crashes occur, it can happen that the rendering will hang for an indefinite time without doing anything. This can waste a lot of time, since the job never finishes, and RenderPal will think that it still is being rendered. To counteract such problems, RenderPal can now automatically restart chunks

if the CPU usage of the rendering client drops below a certain threshold over a period of time. The chunk will then be cancelled so it can be picked up and rendered again.

Net job event improvements

In version 2.5, you can now specify whether a post net job event should be executed if the chunk or job has rendered successfully or if it failed. This way, you can have different events for both cases, allowing you to react accordingly. You only want to run a program if the job went through fine? No problem! You want to execute a Python script if it failed instead? Also possible now!

Speed improvements

The net job chunk list and the client pool list in the main RenderPal view have received some dramatic speed improvements. In previous versions, large net jobs consisting of many chunks could slow down the chunk list drastically, especially when using the Remote Controller. This has now been remedied, and the chunk list is as fast as it should be; the client pool list received the same improvements.

Also, deleting net jobs could have taken a long time, which has been fixed as well.

Plenty of other additions and improvements

RenderPal V2 2.5 is full of other additions and improvements which make working with it more comfortable and easier. These include:

- Linux/MacOSX clients report their CPU and memory usage now
- Improved client logging
- Custom callbacks for output filters
- Support for multiple net job presets in the console RC
- Python improvements (eliminating conflicts with other Python versions)

To get a full overview of all changes, scroll down for the full changelog of RenderPal V2 2.5.

Full changelog for RenderPal V2 Version 2.5

```
+-- Legend -----+
| + New Feature   |
| # Modification/Improvement |
| ! Bugfix       |
| - Removed feature |
+-----+

[+ Global] Added a renderer license management
[+ Svr/Rc] Added user groups
[+ RmCtrl] The console remote controller can now query data from the server using -query
[+ Global] Net jobs now can have tags
[+ Svr/Rc] The net job list can now be quickly filtered by their tags
[+ Server] Added an option to restart chunks of a client if its CPU usage drops below a certain threshold over a period of time
[+ Svr/Rc] Net jobs can now be "colored"
[+ Client] The Linux and MacOSX clients now report the current CPU and memory statistics
[+ Global] The log directory can now be set to a user-defined path
[+ Svr/Rc] Added an option to the net job frame checking to disable frame checking for subsequent, automatically created jobs
[+ Global] Output filters now support custom script callback functions
[+ Global] Output filters now support regular expressions
[+ Svr/Rc] It's now possible to quickly reset all erroneous chunks of a net job
[+ Server] When a net job is deleted remotely, an entry is now added to the event log
[+ RmCtrl] When logging in using the graphical RC, the configuration file of the console RC is updated (if the corresponding entries are empty)
[+ Svr/Rc] Added 5 and 10 minutes to the "Idle clients shutdown" menu (client pool list)

[# Svr/Rc] Dramatically improved the performance of the net job chunk list and the client list
[# Global] It's now possible to execute net job post-events on success and on failure (of a chunk or the entire job)
[# RmCtrl] The console RC now supports multiple presets (using -nj_preset multiple times)
[# Client] The client now logs the output of jobs into separate log files
[# Server] Cancelling a net job chunk will now reduce its attempts count
[# Svr/Rc] Resetting all cancelled chunks of a net job will now also reset their render attempts count
[# Server] Cancelled net jobs no longer count as pending when sending email status reports
[# Svr/Rc] Made adding multiple user accounts easier
[# Svr/Rc] Improved the coloring in the frames list of the manual frame checker window
[# Svr/Rc] When dropping folders onto the scene list (render set tab), files will now be added recursively
[# Svr/Rc] The net job dialog now shows warnings if frame splitting/image slicing cannot be applied
[# Global] The "filter as you type" feature in the event log now has a small delay before updating
[# RmCtrl] The console remote controller will no longer request administrative system rights
[# Svr/Rc] When changing the sort column in the client pool list/net job chunk list, the expanded/collapsed states are now preserved
[# Global] Improved how the internal Python path is set
[# Server] Placeholder values in Python net job events are now escaped (\ becomes \\, for example)
[# Svr/Rc] The lists in the user management dialog now support multiple selections (for easy account/group removal)
[# RmCtrl] Made the console remote controller help output easier to read
[# Svr/Rc] The custom idle clients shutdown time dialog now remembers the last entered value
[# Server] Updated to SQLite 3.7.3
[# Global] Smaller changes and fixes here and there

[! Server] When deleting a net job, unnecessary chunk updates were sent, making deletion very slow
[! Global] Checking whether a directory exists didn't work for UNC paths
[! Svr/Rc] If a pool or client was deleted while the net job dialog was open, a crash could occur
[! Svr/Rc] Resetting all cancelled chunks of a net job did not update the net job status properly
[! Global] Reduced flickering when filtering an event log
[! RmCtrl] The toolbar for the client pool list didn't reflect the current user rights
[! Server] Fixed the "$(RenderSet.outfile_full)" net job event parameter
[! RmCtrl] Fixed a minor commandline parsing bug in the console remote controller
```

Shoran Software

Heidekamp 1
48268 Greven
Germany

Phone:

+49 (0)2571 - 953 353

Fax:

+49 (0)2571 - 953 443

E-Mail:

contact@shoran.de

<http://www.renderpal.com>

<http://www.shoran.de>