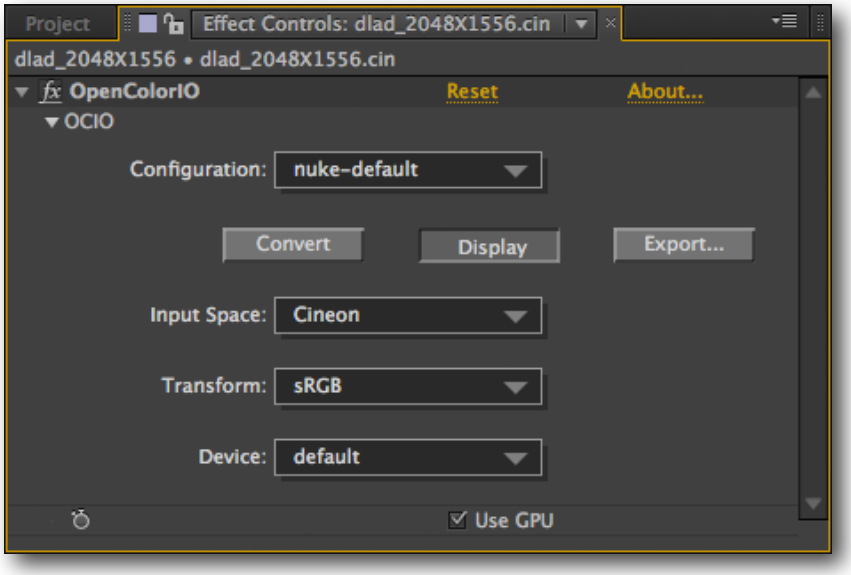
OpenColorIO

After Effects Manual

http://www.OpenColorIO.org



Overview

OpenColorIO ("OCIO") is an open source system for managing color in film production pipelines. It was originally created for internal use at Sony Pictures Imageworks, who then released it publicly in July 2010.

Visual effects artists typically work with a number of different color spaces. Your monitor functions in one color space (often sRGB), but film scans arrive in Cineon Log space, and CG renders are in linear space. Not to mention Panalog, Viper, RED, Rec709, etc.

OpenColorIO provides a a framework for converting between color spaces, and then leaves it up to the studio to customize the conversions as they see fit. For those of us who don't have a studio or an R&D department, Sony has provided several of their own configurations, which you can download separately from the OpenColorIO web site.

This After Effects plug-in includes most of the functionality found in OpenColorIO. You can use it to:

- Convert between color spaces in an .ocio configuration file
- Perform display transformations described in an .ocio file
- Load and apply LUT files in a number of supported formats
- Export color operations in LUT and ICC Profile formats

The After Effects plug-in code was written by Brendan Bolles of fnord software, who also wrote this manual (hi!). But I've contributed the code to the project, so this document and the included compiled plug-in should be considered a convenience and not an official part of of OpenColorIO.

System Requirements

The After Effects plug-in runs only in 64-bit After Effects CS5 and later. If you would like to see it run in earlier versions, feel free to port the UI drawing code to the old APIs and contribute your changes!

If anyone seriously wants to do that or just fix a bug, you can start with my Git fork here: https://github.com/fnordware/OpenColorIO

Quick Start

1. Copy the OpenColorIO plug-in to your After Effects plug-ins folder and restart AE.

2. Download configuration files from the OpenColorIO web site.

3. Copy configurations to the standard directory for your platform (you'll have to create it).
   Mac: `/Library/Application Support/OpenColorIO/`
   Windows 7: `C:\ProgramData\OpenColorIO\`
   Win XP: `C:\Documents and Settings\All Users\Application Data\OpenColorIO\`

4. Download sample images from that same web site.

5. Load the sample file marci_kodak_lad_lg10.dpx into AE. It's in the spi-vfx folder.

6. Create a new comp with the image and apply the OpenColorIO plug-in (under the Utility category).

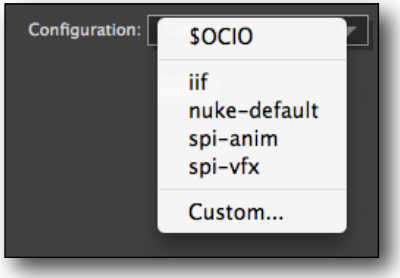
7. From the configuration menu, choose the "nuke-default" configuration.

8. Change the input space to Cineon.

9. Click the Display button. You are now seeing Marci with her color space converted to sRGB.

10. Now load some other files and experiment with other color configurations.

Configurations



OpenColorIO is designed around these config.ocio files that describe a network of color space operations. The name of the configuration comes from the folder containing the config.ocio file. Like footage, your project becomes dependent on these external files.

Using the Configuration menu, there are three ways to choose a configuration:

$OCIO

This first option requires the most technical knowledge to use. Mac and Windows operating systems define various environment variables that can be used to control how programs run. Environment variables can be system-wide, or they can also be custom-tailored for each launch of an application. The OpenColorIO library has standardized on the "OCIO" variable, the value of which should be a full path to a config.ocio file. If the variable is set up, you can use the $OCIO option to use that value. If the variable is not set up, the option will be grayed out.

Standard Configurations

Any configurations found in the shared OpenColorIO directory for your platform (see Quick Start above) will appear in the Configuration menu. This is the easiest and most reliable way to set your configuration, and the least likely to break as a project is moved between different drives or

platforms. Just make sure all the computers you use have the same configurations installed, just as you would for third-party plug-ins.

Custom

This option lets you specify the path for a configuration directly, wherever it may be. This method is the most prone to breaking if the project is moved, especially if it is opened on a different platform. The plug-in does its best to remember the location of the configuration relative to the project, so you should be able to move an entire directory structure around without problems. Should the configuration get lost, re-link the configuration like you would a piece of missing footage.

Use GPU

OpenColorIO has the ability to use the GPU for its processing. In general, it will speed up the OCIO plug-in significantly. However, this option will currently clip any overbright pixel values over 1.0. Therefore it is recommended that you only turn on the GPU when using the plug-in in Display mode and all the pixels are destined only for the screen or some other low dynamic range environment. When using Convert mode, leave the GPU off.

LUTs

Most operations in OCIO are performed using one or more LUTs (lookup tables). Since applying LUTs is a core part of what OpenColorIO does, you can also choose a LUT file directly if it's a format supported by OpenColorIO using the Custom configuration option.

When a LUT is in use, you'll notice an Invert button is present. Note that many LUTs can not be inverted (including all 3D LUTs).

Looks

OCIO supports a notion of per-shot "Looks" that can be configured inside the .ocio file. This feature is currently not implemented in the After Effects plug-in. If you are hoping to design a workflow around using Looks in After Effects, request this feature be added on the ocio-dev email list.

Exporting

Any color operation that OpenColorIO performs can be exported to a LUT, perhaps for use in programs that don't yet support OCIO.

Note that exporting LUTs from Display mode might not be as reliable as the LUTs made in Convert mode. Test your LUTs after you make them. Usually there is an equivalent Convert setting for whatever Display settings you're using.

ICC Profiles

Color operations can also be exported as ICC profiles. This may be of particular interest to users of Adobe applications, as ICC Profiles are the only way to set up a display transform in those programs.

When exporting an ICC Profile, you will be asked to specify your monitor's profile (it will be selected for you by default). This is because ICC Profiles are not LUTs per se. An ICC Profile describes a color space and then needs a destination profile to calculate the transformation. So if you have an operation working and looking good on the monitor you're using (and maybe its profile has been properly measured using a spectrophotometer), then choose your display. If the transform was approved on a different monitor, then maybe you should choose its profile instead.

The great thing about ICC profiles in that (in theory) you should be able to use them on any display, even if the monitors are calibrated very differently. As long as their profiles are set up properly, the application will tweak the display values so that the images look the same everywhere. In theory.

Workflow

After Effects features an ICC profile-based color workflow that has been well-tested. Profiles are assigned to each piece of input footage, output render module, and a project working space is set for the comps. When set up properly, this will allow After Effects to handle all the color transforms transparently for the user. Great, right?

The problem is that no other programs use the After Effects approach. In most production environments, you want all your applications to be performing color operations the same way, and you really want them showing the same on-screen preview. The way to do this is to using LUTs, which describe color operations in a program-independent way. (If you are reading this, you presumably are interested in a universal color management solution, because that's the whole point of OpenColorIO!)

Since After Effects doesn't support LUTs in its native color management system, we have to turn off that system and handle everything manually. While this involves more work on the user's end, it also offers complete control over the process so you can guarantee that After Effects behaves like your other applications. A sample project using this workflow can be found here.

Project Working Space

After Effects' color management activates when a project working space is set. Therefore, do *not* set a project working space because we are handling all the color conversions ourselves. You probably do want to put the project into 32-bit (float) mode, however.

Linear Inputs and Outputs

Even with color management off, After Effects will perform a few color operations if we don't intervene. For example, any 32-bit files like OpenEXR will get a linear to sRGB conversion. We do not want this, so go into the file's Interpret Footage dialog, Color Management tab, and click "Preserve RGB".

A similar checkbox can be found in the Color Management tab of 32-bit output modules.

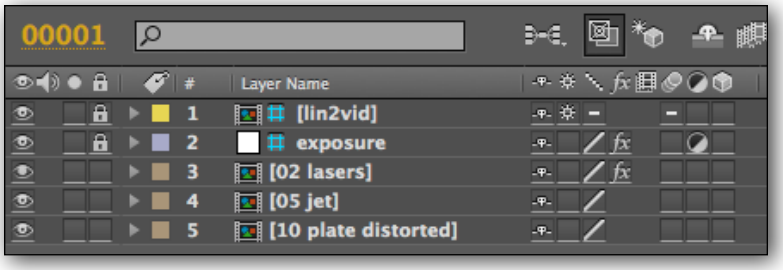
Other Inputs and Outputs

After Effects will not convert 8- and 16-bit files upon import because they are presumed to already be in video (sRGB) color space. Use OpenColorIO to convert these files from video space to linear space. In the sample project using Sony's configuration, video space is named "vd8" and linear space is "lnf". Other color spaces are available if your footage is in a different color space like Log ("lg10") or Rec. 709 ("hd10").

If you are rendering out to a non-linear file format, then the comp you are rendering should use OpenColorIO to convert from linear space to the output color space.

View LUTs

In every VFX workflow where image processing is happening in linear color space, the pixels need to be converted to video space when previewed on screen. That transformation is called a "View LUT". After Effects normally handles this behind the scenes, but in the OCIO workflow we need to make our own.



We make a View LUT in After Effects using an adjustment layer that sits at the top of our comp with OpenColorIO applied in Display mode. Because we want the transform to only be used for displaying the current comp and not pass downstream, we make this layer a Guide Layer in After Effects (under the Layer menu). We then lock the layer because we don't want to ever select it.

You'll want this LUT layer to live at the top of every comp in your project (with the exception of comps that are used to create mattes instead of image data). You will probably end up copying and pasting it into each new comp you make.

In the sample project, we avoid making multiple instances of the OpenColorIO plug-in by creating a comp holding just the View LUT adjustment layer (the comp is called "lin2vid"). We drag this comp into our regular comps and enable the Collapse Transformations switch, which causes the contents of that sub-comp to behave as if they're in the current comp. The advantage to this is that if we ever want to change our display settings, we can make global changes in just one place.

Exposure

The After Effects viewer has a handy exposure knob, but it is not of much use to us since the View LUT will generally crunch any overbrights. You can either temporarily switch off the View LUT or make your own exposure knob by creating another adjustment layer below the view LUT (make it a Guide Layer) and adding an exposure filter.

When using Exposure or any other filter that has a “Bypass Linear Light Conversion” checkbox, you probably want to keep that checked. It’s similar to the “Preserve RGB” checkbox mentioned above.

Sampling Colors

Your View LUT will also result in colors you sample either with the eyedropper or just when viewing the Info palette to be in video space instead of linear space. Temporarily turn off the View LUT to sample directly.

This is a Pain

Yeah, I know. Send email to aebugs@adobe.com and tell them you want color management to be more open, extensible to accommodate other workflows.

Alternatively, you could experiment with exporting ICC profiles from OpenColorIO and using those in AE’s color management.

Legalese

Plain English License Agreement

You are free to do anything you want with the software included in this package, but fnord software and Sony Pictures Imageworks can not be held responsible for any positive, neutral, or negative effects from doing so. So don’t try to sue us!

This plug-in includes code from OpenColorIO and several other libraries. Here’s the relevant legal mumbo jumbo:

OpenColorIO

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Sony Pictures Imageworks nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Pystring

Copyright (c) 2008-2010, Sony Pictures Imageworks Inc All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the organization Sony Pictures Imageworks nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS

FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

TinyXML

TinyXML is released under the zlib license:

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

yaml-cpp

Copyright (c) 2008 Jesse Beder.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Little CMS

Little CMS Copyright (c) 1998-2010 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Boost

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and

transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Fin