

2023-05-01

May 1, 2023

Host: Carol Payne

Secretary: Carol Payne

Attendees:

- ☒ Rémi Achard (*TSC*) - DNEG
- ☐ Mark Boorer (*TSC*) - Industrial Light & Magic
- ☐ Mei Chu (*TSC*) - Sony Pictures Imageworks
- ☐ Sean Cooper (*TSC ACES TAC Rep*) - ARRI
- ☒ Michael Dolan (*TSC*) - Epic Games
- ☐ Patrick Hodoul (*TSC*) - Autodesk
- ☐ John Mertic - Academy Software Foundation / Linux Foundation
- ☒ Carol Payne (*TSC Chair*) - Netflix
- ☐ Mark Titchener (*TSC*) - Foundry
- ☐ Carl Rand (*TSC*) - Weta Digital
- ☒ Doug Walker (*TSC Chief Architect*) - Autodesk
- ☐ Kevin Wheatley (*TSC*) - Framestore
- ☒ Thomas Mansencal - WetaFx
- ☒ Zach Lewis - Method Studios

Apologies:

- Mark Titchener
- Kevin Wheatley

OCIO TSC Meeting Notes

- 2.3 Release Timeline
 - We had a discussion with the VFX Ref Platform. Last year, we got an extension to release in October.
 - This was a bit late for some vendors, so goal this year is to stick to the end of August release
 - 2.3.0 to lock API, end of year for a 2.3.1 patch/bug fix release
 - This should include a configs release as well, for built-in config updates
 - ACES 2.0 will wait for CY2025
 - Library features done earlier.
 - Main features: config merging, config editor, any others?
 - At next TSC: discussion on what folks would like to see for 2.3? Vulkan support, etc
- SIGGRAPH 2023 / Open Source Days Plans
 - Group agrees with plan: YES to virtual town hall, YES to SIGGRAPH BoF (meetup style).
 - Try to see when schedules work out to be able to do a happy hour after again 😊
- Lots of PRs stacking up
 - Lots of small PRs that need a review, one a bit more complicated
 - Please take some time to have a look as you have time
 - Planning one more 2.2 patch release before 2.3 (for processor cache PR)
- Config merging feature, continued
 - Doug has been updating the slide deck strawman, with examples and also the discussions, questions and concerns so far
 - If we go the #include approach: we'd need to basically throw out and re-write the yaml parsing functionality we currently have
 - If anyone has ideas on how to implement the include in a less destructive way, please let us know
 - Conclusion: merge is a lot cleaner, could be implemented as an app-helper approach as opposed to a core library
 - With merge - much easier to do work & overrides and comparison on two config objects vs. in the YAML
 - Michael: use case 1 is dependent on the configs, use case 2 is dependent on an application. Philosophically hard to compare. Are they really two features?
 - Yes, merge requires external action, where include can be implicit
 - Thomas: As a dev I certainly prefer include but I appreciate that it makes things much harder. Probably less visibility on what will happen and the end result. Merge seems more predictable all in all.
 - No objections from present TSC members on moving forward with the merge approach. Will send the deck to those not present for their review over the next two weeks, and at the next TSC will make the final decision. If we don't hear objections, we will move forward.