



Nuke Studio - Timeline and compositing in one package. And the Project? Cat & Moth, which we already know from the DP issues 22:01 and 03.

Timeline Tools von The Foundry – Teil 2: Color, Comp und Conform

Nuke and Nuke X are the de facto standard in node-based compositing. But even seasoned Nuke artists have often had little or no contact with the closely related reviewing and finishing products. That's what we want to change that a bit with this overview and in this part two we will take a look at what we can do with tags and Colors and how to get to the nodes... Here we go!

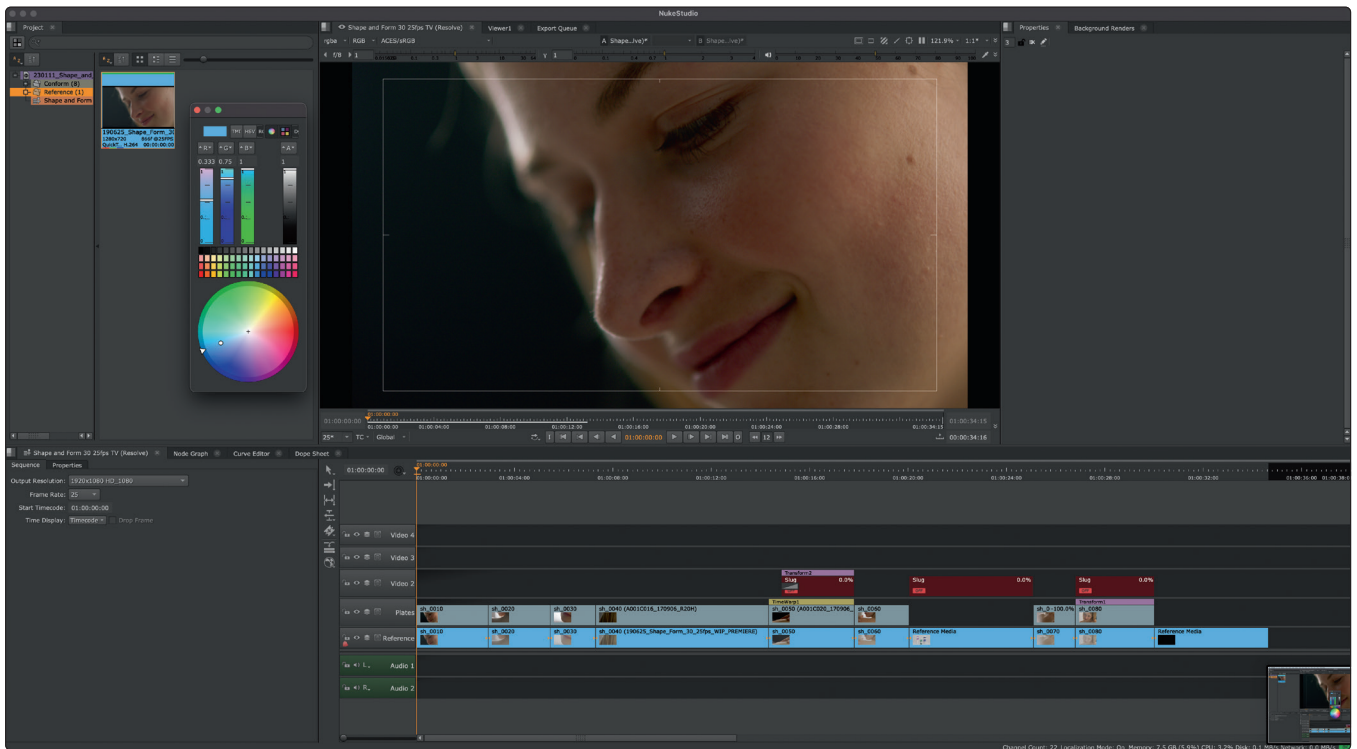
by Christoph Zapletal

Colors & Tagging

We already know this from Nuke's Node Tree: The colors of nodes always have a meaning, they assign nodes to a certain category and generally serve for better orientation. But it can also be very helpful to color nodes individually, just to facilitate this orientation. It is no different in the Timeline Environment of Nuke Studio. If we take a look at the Project Bin, we can already see that the different items have different colors. By default, the Project itself is Purple, Bins are Olive, and Timelines have an orange tint. If we look to the right to the proxies, they are all colored pale blue. If you want to have different default colors for your projects, you can change this in the Project Settings (like in Nuke by hitting „Shift-S“) under Panels – Project Items. Here you can also assign different colors to different file types. But also individually you can assign a color to an item – for example to be able to distinguish the different timelines in a project with several edits. To do this, you can right-click on any item and change the color using the „Color“ option.

What is interesting here – if an item is used in one or more timelines, whether single or multiple, this color will also be updated in the timeline. So coloring it accordingly can really help you find a particular clip again. To make this even easier, you can sort by color within the thumbnail view. In our conforming workspace, to the right of the „Project Bin“ tab, we find the „Tags“ tab. The tagging system is really powerful, especially since it works closely with all kinds of asset managers like Shotgrid, FTrack, NIM, Cerebro and so on. But even within Nuke Studio, tags can really help us out, especially when the status of different comps, versions, etc. changes.

If we take a closer look at the tagging window, we will find quite a selection of tags, which Foundry already provides. We can drag and drop the appropriate tags from the tags panel onto the timeline. This also works with multi-selections and of course each clip can have multiple tags. So we can distribute descriptive tags, like whether an element is for foreground or background, or whether it's a green screen shot.

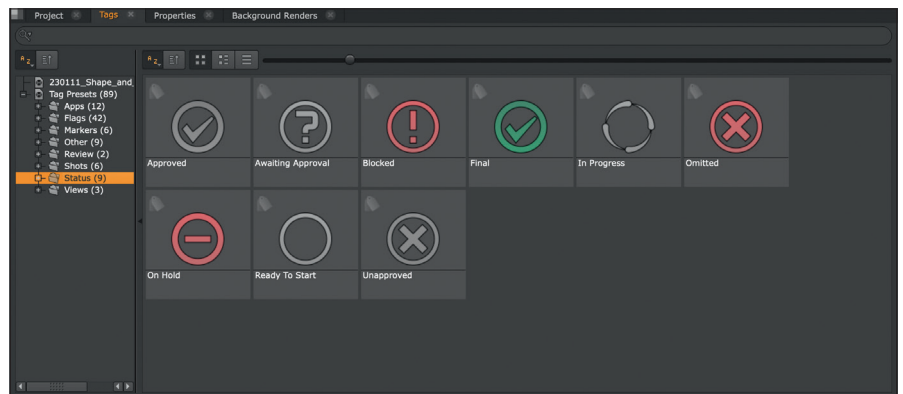


In the same color Space – Project Bin and Timeline.

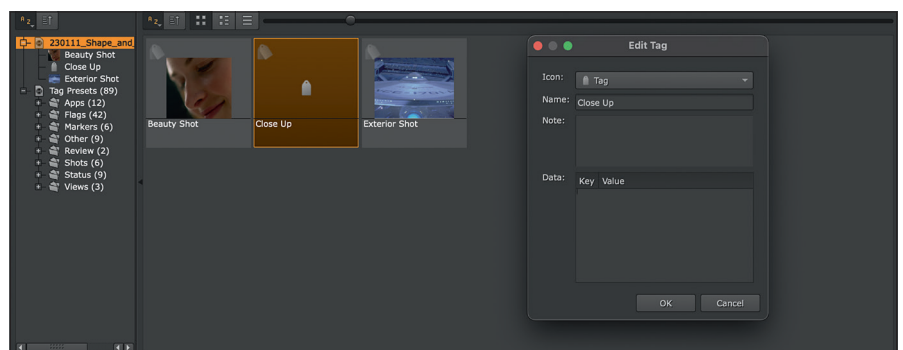
Even the Color tagging is dynamically linked between Project Bin and Timeline.

However, we can then also assign a status tag such as „Ready to Start“ or „Omitted“ to the same shot. If you want, you can also distribute the tags in the spread sheet by right-clicking, but then things quickly become confusing, because we don't find the pre-division into folders here, but a gigantic pull-down with all existing tags. By the way, not only clips, but also tracks and entire sequences can be tagged. It is also possible to create your own tags. Via the Presets tag we see our Project Bin displayed. By right-clicking on „New“ we can generate tags as well as bins for sorting the tags. PNGs are supported as thumbnails for the tags. To add your own PNG, click on the „Icon“ pull-down and scroll all the way down. There is the „Custom“ option and we can give our tag its very own flair.

Another very nice way to distribute tags is to simply drag them into the viewer. A HUD appears on the viewer and we now have the option to either assign a tag to the sequence, tag an individual frame, tag the clip within the timeline („Tag Track Item“) or tag the source clip („Tag Clip“). And depending on where we drop our tags, they will appear above the Viewer (For Individual Frames), In the Project Bin (For Clips and Sequences) or in the Timeline itself (For Track Items). With a double click on the tag we can open an Info Window, where we can assign Notes or Metadata to the tags. These are individual, meaning they are not shared



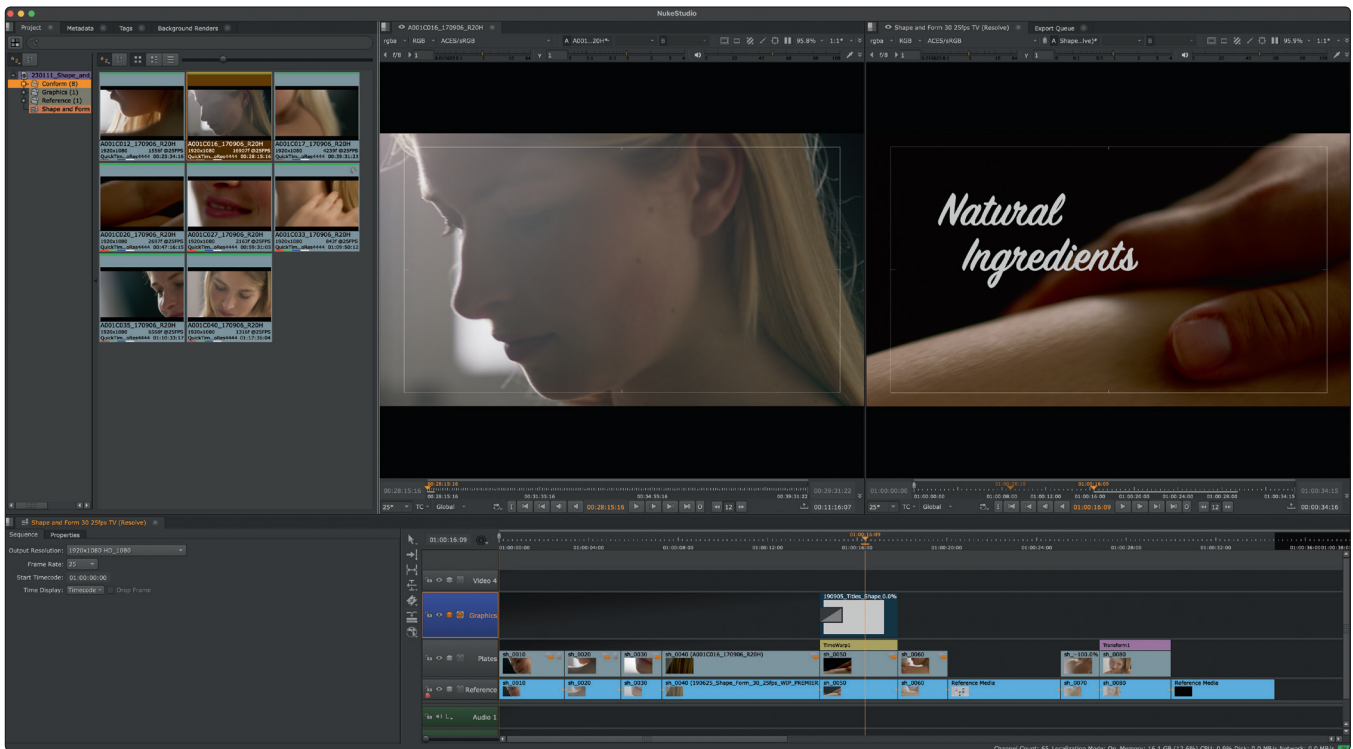
It will be a while before we are allowed to hand out green check marks.



With your own tags, the project can be made even more individual.

with other tags of the same name in the project, and can help share additional info. However, only if the Notes are written after dropping the tag: Notes that are written during creation or within the tag panel are then also consequently distributed. Especially

with longer timelines it makes sense to be able to search and filter for tags. And this can be done quite easily by drag and drop. The tag you are searching for can be dragged from the tag window into the search bar of the spreadsheet in our conforming work-



It's really starting to look like editing here.

space. The Search Bar of the Spread Sheet has the same options under the magnifying glass as the search in the Project Bin. That means and-or- search, flagging, etc. are available, but unfortunately you cannot combine the tags search with a text search. But of course you can search for tags directly in the Project Bin.

Editing

As mentioned before, Nuke Studio is not a dedicated NLE, but rather a finishing system, so the features are rather limited in regards to editing. Nevertheless, we should now switch from the conforming workspace to the editing workspace. For this we can either use the workspace tab in the menu bar or the shortcut Shift-F2. What we get already looks a lot more like editing. Now we have a Source Player in the middle of our interface and every clip we double-click in the Project Bin will open inside of that Source Player. If you know the Nuke player, you'll feel right at home here, even putting ins and outs works as you're used to. But if Nuke Studio is not the measure of all things when it comes to editing, then let's take an example that fits more into the finishing environment for which Nuke Studio is intended. We import an animated graphic with an alpha channel and want to apply it to our timeline.

To get our graphic into Nuke Studio, we can either drag and drop it into our Project Bin, we can right-click in the Project Bin and choose „Import Files“ or „Import Folders“, or

we can choose one of these two options from the „File“ tab in the Menu Bar. And as mentioned above, we can now bring this graphic into our Source Player with a double click. From here on we have several options: Drag and drop onto the timeline would be one option. Another would be to set an In point with „I“ and an Out point with „O“ in the Source Player and then execute an Overwrite Edit with the shortcut „M“. But beware: We have to make sure that the right track is selected in the timeline beforehand, otherwise we can mess up our nice conform pretty quickly. So, either select an empty track (by default, we have four tracks to choose from) or add a new one by right-clicking below the timecode window on the timeline. The selected track is now marked blue and gets our graphic with the „M“. If we move over it with the playhead, our graphic is not yet displayed with the correct alpha blending. For this we have to switch on the blending for the track. To do this, we press the icon that looks like a stack and is located right next to the eye for the visibility of the track. By the way, if we click and hold here, we get various blend modes displayed, all of which should look familiar to us from Nuke. However, the situation of our graphics has not yet improved much, the alpha channel still seems to be nonexistent for Nuke Studio. The remedy for this is the icon right next to the one for track blending, and that is the „Mask Blend by Alpha“. And there you go, our graphic is sitting right atop the image as it should. As is so often the case, many roads

lead to Rome and there are certainly a few more scenarios we could go through here, but anyone who has ever sat in front of a program with a timeline will quickly find their way around here. So you can start with the „arrow up“ and „arrow down“ keys to jump between different edits, for an insert edit you take the „N“ instead of the „M“ for overwrite, and with „J“, „K“ and „L“ plays the clip backwards, stops it or plays it forwards – while repeatedly pressing „J“ or „L“ doubles the playback speed again and again, up to a maximum of 32 times. And then, of course, there are the Editing Tools, which we find in the Timeline panel between the Sequence/Properties panel and the actual Timeline. The five icons up to the small dash belong to the Editing Tools, with several tools under each icon. At the top is the „Multi“ tool, which we've actually used all the time so far and which – depending on the workflow – is also sufficient for the vast majority of tasks. If we click on the arrow at the bottom right of the „Multi“ tool, three more options appear. Below that is the Select Tool, and under the arrow are options for quickly selecting multiple edits in a specific area of the timeline.

Next up is the „slip“ tool and under the arrow his best friend, the „Slide.“ Below that you will „Roll“, „Ripple“ and „Retime“ and last but not least the universally loved razor blade with a separate option to cut all tracks at once and, how conciliatory, also a „Join“ function to join clips back together. If the arrow-clicking is too fiddly for you, you'll be

glad to know that there are shortcuts. The entire „QWERT“ block at the top left of the keyboard is dedicated to editing tools: „Q“ for the „Multi“ tool, „W“ for Select, „E“ for Slip and Slide, „R“ for Ripple and Roll, and „T“ for the Razor. And, you guessed it, with a consecutive press on the hotkey we toggle through the respective tool. If you prefer to work with a mouse or pen and still don't like to click through all the drop-downs, you should spend a little time with the „Multi“ tool. Depending on whether the cursor is at the bottom of the clip (Slip) or at the top (Slide), left or right (Roll) or on an edit (Ripple), you can execute other functions – without any shortcuts or menu clicks.

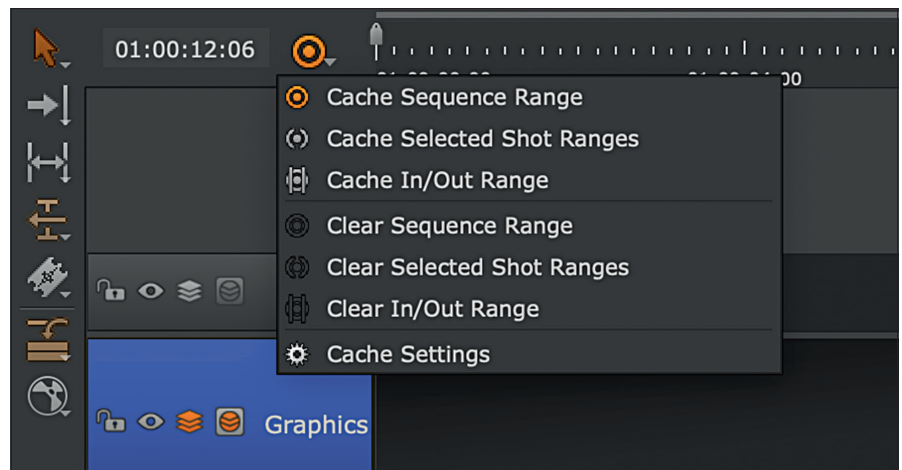
Timeline Effects

For our next part, it is best to switch to the „Finishing“ workspace. This will make room for a larger viewer, since we no longer need the source player. Below our Timeline Tools are two more tools: Add Soft Effects and Create Nuke Comp. Let's deal with the Soft Effects first. These are considerably less than we have nodes in Nuke or Nuke X, but they should all be familiar to us in some way: Transform, Crop, Grade, ColorCorrect – we all know them. And when we apply a grade to a clip in our timeline, for example, we get exactly the same parameters in the Properties panel as with the classic grade node in Nuke. So we won't dwell on the function of the individual tools here. But why is the selection of timeline Effects, or Soft Effects as they are called in Nuke Studio, are so much smaller than in the compositing app. Well, for one thing, the type of effects you can apply on a timeline is already limited by the architecture, because after all, on the timeline we don't have the ability to pipe matte inputs or geometry or anything else into an effect. On the other hand, all the effects that we can apply on the Timeline are GPU accelerated and thus real-time. A Defocus, for example, would not be able to do that. But that is not a concern, as the Timeline is not the place to build advanced effects in Nuke Studio, for that we will move to Nuke Studio's integrated Nuke X later. The Timeline effects are meant for everything that belongs to finishing, i.e. transforms, resizes, last-minute color corrections or burn-ins. Speaking of realtime: Nuke Studio does not have a dedicated frame store, just like Nuke, but just links to the files' actual location. If this is on a server, the resolution is high and the connection is slow, this can be tough. By default, Nuke will try to cache the timeline into RAM, but even that fills up pretty quickly. That's why it's worth having a fast SSD as a cache drive for Nuke Studio – it behaves quite similar to Nuke. But beware: The place

where Nuke Studio stores the time-line caches is different from the place where Nuke locates its files. So better check the Project Settings under Performance – Caching that the cache files go to the right place – and do not fill up the system disk. If all this is set correctly, we have the possibility to cache our timeline in parts or in its entirety with the small bullseye, which is hidden on the timeline to the right of the timecode display – including any soft effects.

But back to our Soft Effects. Of course, they can be copied with the familiar shortcuts Ctrl+C and Ctrl+V (Cmd+C and Cmd+V on the Mac) or by right-clicking. But a nice example of how features come from the node tree into the timeline and really make

effect in the stack is rendered, then it goes to the top. Since we don't want to build complex comps in the timeline anyway, it will rarely be relevant, but it's good to keep this behavior of Nuke Studio in mind. But Soft Effects can not only sit on top of a single clip, but also like an Adjustment Layer on top of our entire timeline. To do this, we need an empty track, which we then select. Right-click or click on our Soft Effect icon and we can place any Soft Effect we want here. But the one that will be really important, especially if we want to share a work-in-progress version, is the Burn-in Effect: In six areas of the image we can distribute important metadata info, ranging from clip and shot names to our project and timeline names to



Particularly indispensable for larger projects: A cache for the the timeline.

sense there is the possibility to clone Soft Effects. This works wonderfully via the right-click menu or via the shortcut Ctrl+K/Cmd+K known from Nuke.

Now we not only have two identical color corrections, but both are also dynamically linked to each other, which means that if we change the value in a soft effect, this change is applied to all clones. If we now also realize that we can apply such a clone to all clips with a certain tag, for example via the search in the spread sheet, then it becomes clear how much this feature, together with good prep work, can make finishing easier for us.

Likewise, just as with the nodes in Nuke, we can save presets via the small wrench in the upper right-hand corner – and then distribute them across our timelines. Presets can be saved globally or on a project basis. And, unlike in many other apps, the same type of soft effect can be stacked multiple times. Speaking of stacking: The order in which soft effects are stacked is crucial for the order in which they are calculated. The order is exactly the other way around than we are used to from Nuke: First, the bottom

track and timeline info. You should definitely play around with this.

There is one more layer on which we can apply Soft Effects: on the version of a clip. To do this, we need to get a little ahead of ourselves: Later, when our renderings from Nuke or another application come back to our timeline, we can treat them as a new version of a clip in the timeline, which makes updating really easy. We can then assign Soft Effects to these individual versions. What can this be useful for? Well, for example, let's assume that our comp artist or we ourselves are doing the rendering in the wrong Colorspace.

No problem, he or she has already adapted the script for the next version, but the rendering capacities are limited and the client is already getting itchy because he finally wants to see the current state of things. So a soft effect on a version is a pretty useful thing. To accomplish this, we right-click on the corresponding clip and select „Open in – Timeline View“. A new tab opens with a timeline containing only our one clip. To make sure that the soft effect is not passed on



The properties, the colors ... the soft effects are very reminiscent of Nuke.

to other versions, we create a new video track and put the desired soft effect on top of it.

In our scenario it is best to use an OCIO with the „+“ in the toolbar. If we have chosen „Sequence“ or „Clip“, we can select several ranges in a pulldown, from the whole length of the clip or sequence over the „Current Item“ (i.e. the clip on which the cursor is currently positioned) and individual ranges up to „Current Frame“ are all possible. We can then confirm this selection with „New“ and we have already created our new annotation on which we can now paint and write.

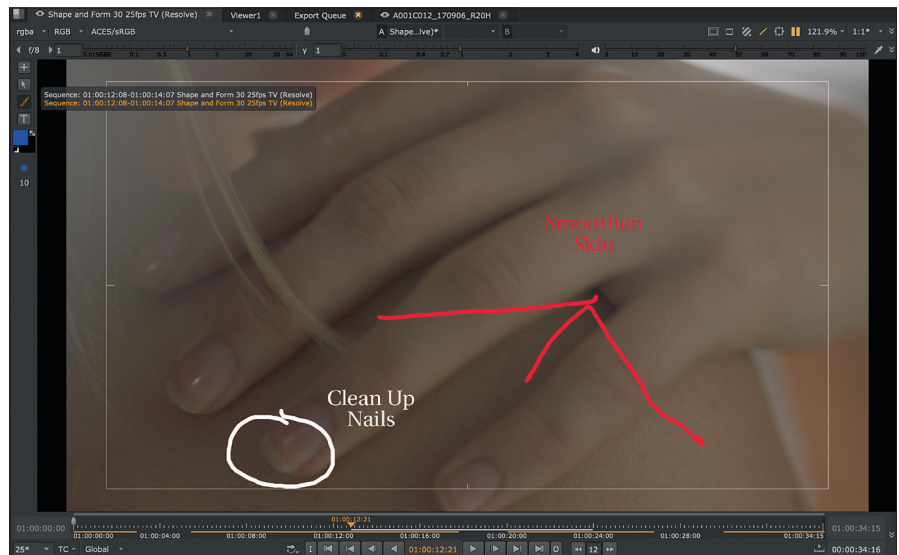
Annotations

The clumsy drawing of work instructions and comments by other departments not only satisfies the desire for childish scribbling, but also helps massively with forwarding feedback, recording briefings and keeping to briefings. In order to put annotations onto our timeline, we click on the small pencil icon above our viewer, directly between the clipping warning and the region of interest. A small toolbar appears to the left of the viewer, where we can set our first annotation using the „+“.

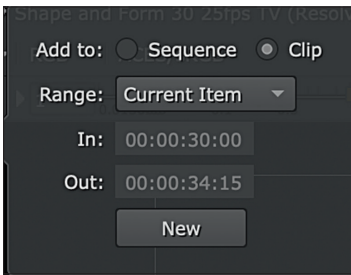
Like Timeline Effects, Annotations can be set at three different levels: within the sequence, within a clip, or on a specific version. All three levels have their own characteristics: If you place annotations within the sequence, they are placed on their own annotation track, where they can be edited independently of the underlying clips. If you drop it on a clip, it will be copied along with the clip within the sequence. And on the individual version, you make sure that the feedback applies only to the specific version. Usually, however, the first two variants are

used more frequently, which is why these are the two that are offered to us in a pop-up window when we add our first annotation with the „+“ in the toolbar. If we have chosen „Sequence“ or „Clip“, we can select several ranges in a pulldown, from the whole length of the clip or sequence over the „Current Item“ (i.e. the clip on which the cursor is currently positioned) and individual ranges up to „Current Frame“ are all possible. We can then confirm this selection with „New“ and we have already created our new annotation on which we can now paint and write.

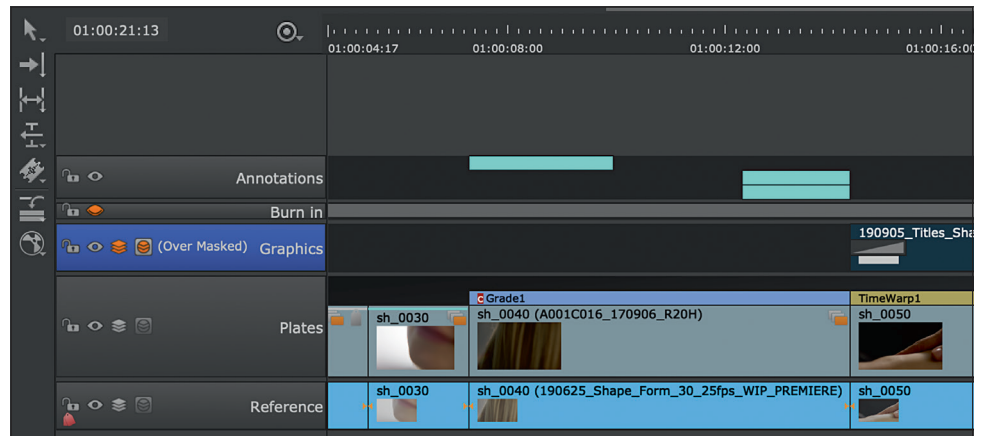
The tools are really self-explanatory, a brush and a text tool, plus a color picker and a radius for the brush. But that's all there is. After all, the whole thing is intended as a container for feedback, not as a retouching tool. And the simplicity ensures that the annotations can be used via a Python Hook, which of course makes the pipeline t.d. a happy camper. And if you don't want to see the annotations anymore, just click again on the pencil in the upper right corner above the viewer. But even without a big pipeline, these annotations really make sense, because their really big benefit, also for Nuke Indie



Finally we can scribble around in the viewer ...



Sequence or Clip, Range or Single Frame, we also want our feedback to sit exactly where it belongs.



On both the clip and the added annotations track, our annotations are displayed in light blue.

users for example, is the fact that they can be exported as annotations nodes into Nuke scripts. So you can then see the annotations directly in the comp and they can also be updated from Nuke Studio at any time.

Export

Probably no other tool offers such a complex export manager as Nuke Studio. In addition to the quick work-in-progress Quicktime, we can do everything from creating a complete backend folder structure or integrating our exports into an existing one, to transcodes and XML or AAF roundtrips, to automatically creating Nuke scripts for every single shot in our project. We'll try to go as deep as possible here, but in the end this is the part to test extensively before throwing yourself into a real production with Nuke Studio.

Of course, any item that we have in our bin can be exported. Let's just start and say that we want to export a single full length clip from our conform. We want the whole thing to be sent to the director as a small file so he can go back and look for another take. So we want to export an H.264 in half size. With a right click on the corresponding clip in our bin we can select „Export“ and the Export Manager appears.

The first choice we have to make in the Export Manager is the type of export. Since we want to export a single clip, the options „Process as Shots“ and „Export as Shots“ are grayed out. If this is our first time in a freshly installed Nuke Studio, we should see seven presets here in the „Using Local Export Presets“ column. Unfortunately, what we need is not here as a ready-to-use preset. But before we build a new preset from scratch, let's use the „Transcode Clips Apple ProRes 422 MOV“ preset as a base and copy this preset using the „Duplicate“ button – the one right next to the „Delete“ button with the minus sign.

Below we see the „Export to“ path with a variable or, to stay in Nuke Studio-Lingo, a token. We could now simply specify a path via the „Choose“ button, but there is a more elegant way. This token – the term between the two curved brackets – points to the project root folder, which we did

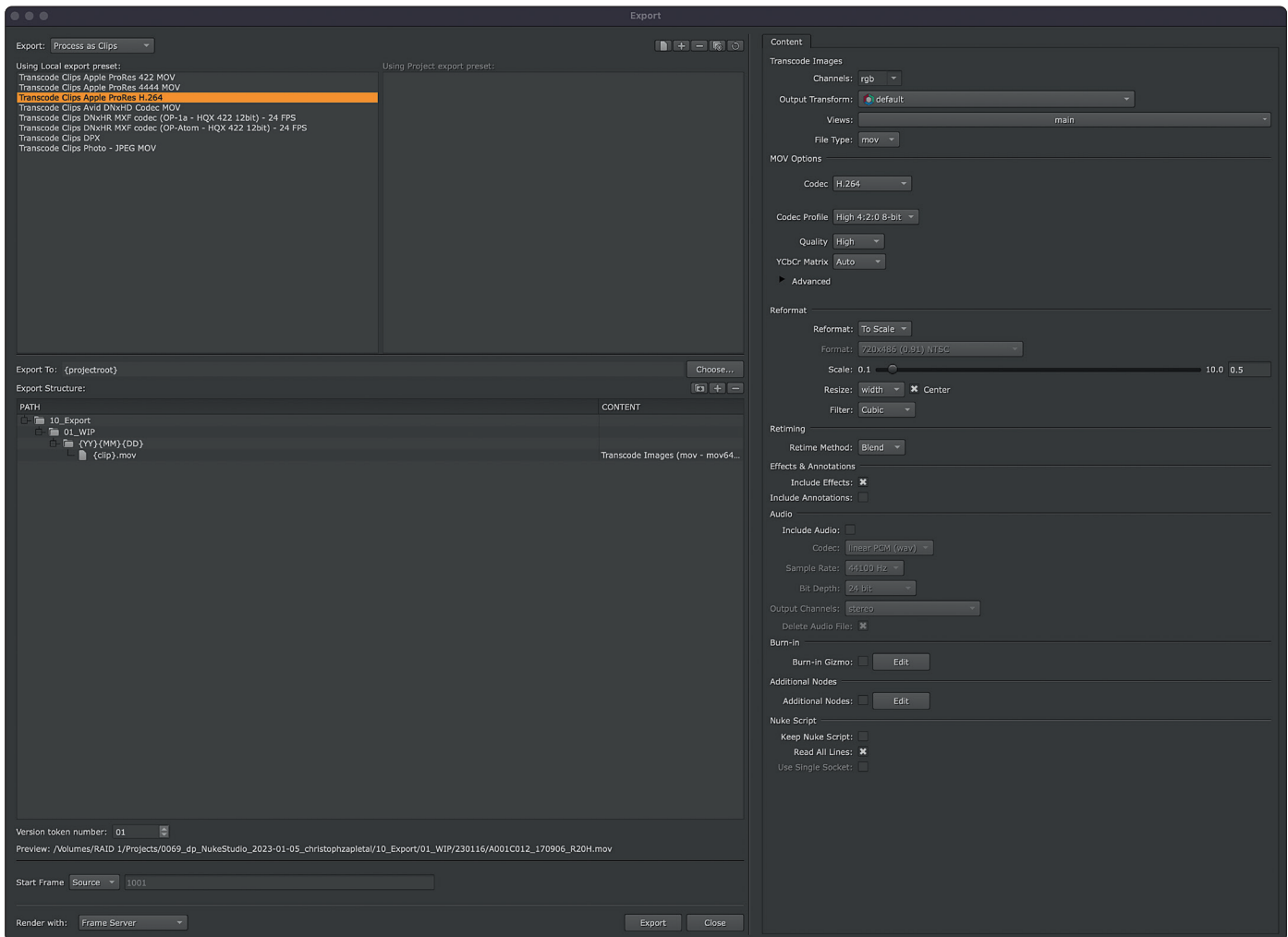
define at the very beginning of this workshop. If you haven't done this yet, just press „Cancel“, save the changes you made so far and set the folder under „Project – Project Settings“. The way it is set up now, the export would end up in the right in root folders of our project. We can do better than that.

Surely everyone has a dedicated place for exports, maybe even for intermediate exports, and has this place in his or hers project file structure. We can now add this exact folder to our export path. But instead of putting the path directly behind our Project Root Token, we create the folder in the Export Structure view directly below it. This is not only more visible, it also makes customization and debugging easier in case of doubt. For this we use the „Add a new Directory to your Export Structure“ button, directly below the „Choose“ button. Since we know our customers and know that we will be exporting things for them time and over again, we can of course create a date folder. Again, we could just do this manually, but if we are smart about our preset, Nuke Studio can do it for us in the future. Three simple tokens help us here. Namely, written directly one after the other, {YY}{MM}{DD}. Almost self-explanatory, the three of them will export the date in a format suitable for our folder structure. While we are on the topic: At the bottom of the Export Manager we always get a preview of how the path will look like. If we have the actual file (in our example the entry with the name {clip}. {ext}) is selected, we can still set everything possible for our transcode on the right. So we set the codec to H.264 and set a „To

Scale“ under Reformat and give it a value

of 0.5 We then have several options, such as whether to render on Nuke's own frame server, in a foreground process or with the help of an external render manager such as Deadline – but of course this must also be installed accordingly.

In the next case things get a bit more complex – and hopefully show why these presets are so powerful. Let's take a very common task: Another department would like to have the current status of our project as XML with the corresponding reference file as H.264 – with all annotations. Unlike the single shot, we're going to build this preset from the ground up to also get a better understanding of what tools are actually all available to us in the Export Manager. So we select our sequence in the Project Bin and right-click again on „Export“. This time, however, we select „Process as Sequence“. The layout is almost identical to the last time, however, we now have the option at the bottom right to select the individual tracks to be exported. The presets above are also a little different, but they don't bother us at all this time, since we're working „from scratch“ anyhow. With the „+“ icon at the top center we create a new preset, which we name accordingly, for example „XML + H.264 REF“. Since we have set our project root in the project settings, we can leave the token {projectroot} under „Export To“. However, the „Export Structure“ underneath is still completely empty. Now we can fill it as we like. And if you use tokens, you can adapt the whole thing to your needs. So let's start creating our export structure. In the screenshot we see the first two as folders with a fixed value, i.e. folders that will be called the same in every project and are always there. This is followed by the already known date folder from the first case. Then we create again a folder with a fixed name, namely „XML“, because we want to create separate subfolders for the XML and the MOV. And now we create with the „+“ icon, which is not accompa-



Our first case study: export a single clip as H.264.

nied by a folder icon, a new content, i.e. a file for this folder, which Nuke Studio should generate. But before we select the XML format, we enter the file name. In our example: `{YY}{MM}{DD}_{hour24}h{minute}_{sequence}.ext`

The tokens used should by now be known to all of us, but apart from that there is really no limit to our own requirements and wishes. Now we have to tell Nuke Studio

what the file we named should contain. A click into the empty „Content“ column brings us to a pop-up where we select „XML Exporter“. The options we have on the right under „Content“ are very manageable. If you like markers, you can add them, but you don't have to.

Let's jump with the cursor back to our date folder (the one with the token string `{YY}{MM}{DD}`) and create a subfolder

with the name „MOV“. Inside this folder we create again a new content, but this time we select „Transcode Images“. We can set the

Parameters like with the first case, but make sure that „Include Annotations“ is checked. And that's it!

Let's move on to the third case: Let's move away from small exports of individual Quicktimes, we now want to transcode our shots properly and maintain them in our existing file structure. Let's say: Everyone has a different folder structure and a different workflow, some are task-based,

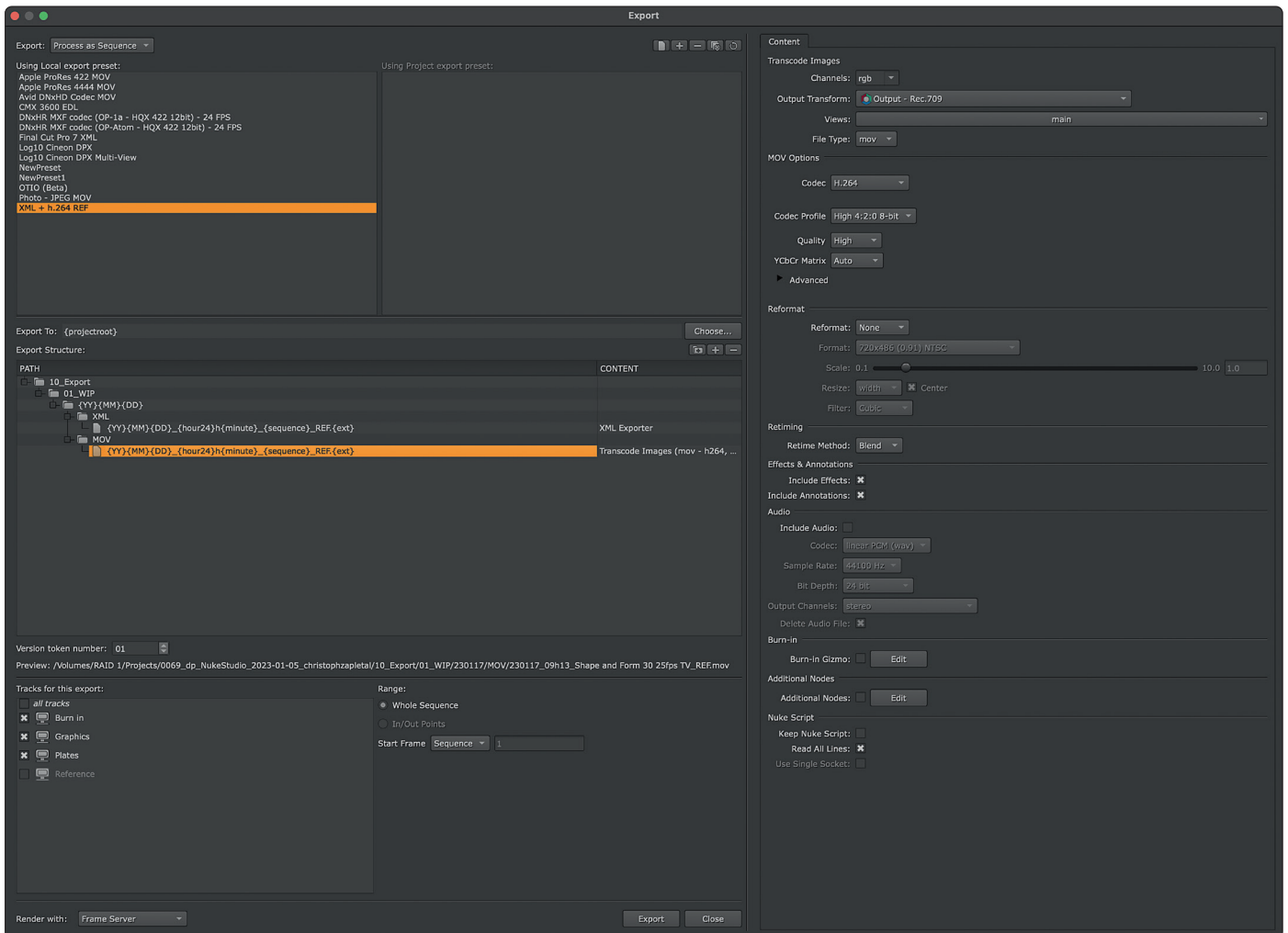
others are shot-based. The Export Manager is remarkably solid and flexible. However, as mentioned before, you should always test-run any export preset before using it on an actual production.

In our example we want to export all shots in single folders as OpenEXR with 10 frames heads and tails. Any additional tracks (e.g. for screen inserts, splitscreens, green-screens, etc.) should be taken into account. Timewarps should not be applied here yet, so that they can be applied later, either on the transcoded material in the Nuke Studio Timeline or in Nuke Comp. The procedure is

This field will replace the following keywords appropriately when exporting:

- {DD} - Day of the month as a decimal number, [01,31]
- {MM} - Month as a decimal number, [01,12]
- {YYYY} - Year with century as a decimal number
- {YY} - Year without century as a decimal number [00,99]
- {ampm} - Locale's equivalent of either AM or PM.
- {binpath} - Path within the bin, up to the selected BinItem. If no BinItem selected, path up to project
- {clip} - Name of the clip used in the shot being processed
- {day} - Locale's abbreviated weekday name, [Mon-Sun]
- {ext} - Extension of the file to be output
- {filebase} - Base part of the file name for the media being processed. Includes everything preceding the period before the extension. For example, {filebase} for 'myshot.%04d.dpx' would resolve to 'myshot.%04d'.
- {fileext} - Filename extension part of the media being processed
- {filehead} - Head part of the file name for the media being processed. For a file sequence, includes everything preceding the frame numbers, for single files behaves the same as {filebase}. For example, {filehead} for 'myshot.%04d.dpx' would resolve to 'myshot'. 'myshot.mov' would also resolve to 'myshot'.
- {filename} - Filename of the media being processed
- {filepadding} - Source Filename padding for formatting frame indices
- {filepath} - Path to the file name for the media being processed.
- {fullbinpath} - Path within the bin, up to the project
- {fullidday} - Locale's full weekday name
- {fullmonth} - Locale's full month name
- {hierotemp} - Temp directory as specified in the Application preferences
- {hour12} - Export start time hour (12-hour clock)
- {hour24} - Export start time hour (24-hour clock)
- {minute} - Export start time minute [00,59]
- {month} - Locale's abbreviated month name, [Jan-Dec]
- {projectroot} - Project root path specified in the Project Settings
- {project} - Name of the parent project of the item being processed
- {second} - Export start time second [00,61] - '61' accounts for leap/double-leap seconds
- {timestamp} - Export start time in 24-hour clock time (HHMM)
- {user} - Current username
- {version} - Version string 'v#', defined by the number (#) set in the Version section of the export dialog

The list of tokens – essential for automating exports.



Case number 2: Export XML with associated reference in one setting.

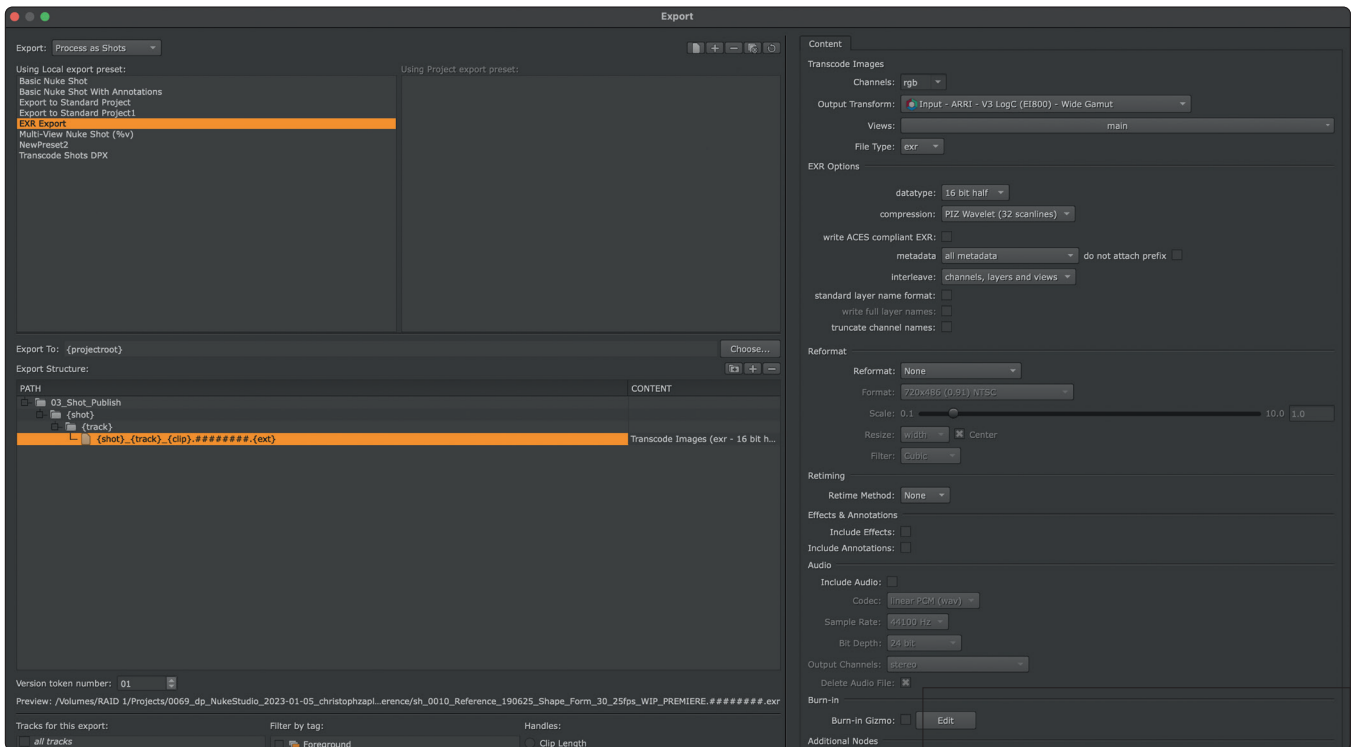
very similar to the second example. Right-click on the sequence in the Project Bin to open the Export Manager and select „Process as Shots“ in the upper left corner. Next we create a new preset with the name „EXR Export“. Then, starting from the root folder of the project, we set the path to the folder where we want the transcode to take place in each shot folder. With the tokens {shot} and {track} we can retrieve the shot name with index and the track name we assigned at the beginning of the workshop and use it as the folder name. For the actual filename we use here the structure {shot}_{track}_{clip}.#####.ext).By the way, {clip} provides the original clip name and it makes sense to include it here, so that an artist who then takes over the shot in the comp is always in the know where this shot comes from. Once again, we select „Transcode Images“ as the content. In the lower third of the export manager, we can now set various filters and settings, starting from left to right. On the far left, we select the tracks that are relevant to us – after all, a comp artist who is doing retouches does not necessarily have to be provided with graphics

such as lower thirds. If our sequence has been tagged properly, we can filter our export accordingly and, for example, export only shots that have been tagged with the tag „VFX“ or „Retouch“ or „Matchmove“. The concept of tags can definitely help to export only what you really need. Last but not least, we can specify handles, set a custom start frame (very handy to avoid untidy comps) and of course apply any retimes – which we don't want. On the right side is again our „Content“ column. Of course, everyone should set this up according to their individual needs, but here are a few tips: We should definitely consider the Output Transform so that it matches our material or our Color Pipeline. If you work in classic Nuke Color

Management, for example, you can transcode to „scene linear“, if you work in ACES, you might prefer to preserve the color space of the source material. By default, all channels are selected for EXR transcodes, which can be set to „rgb“ for actual footage. The rest is pretty self-explanatory or will be either later on or become only relevant in other, specific workflows. With that in mind,



And if everything worked out, our export should look like this.



Almost ready for our first comp. . . our transcode is done.

we can now press „Export“ and hopefully see our folder structure filling up.

Building Tracks

We've got all the footage exported now, but since we want to see what ends up in the comp, our first step is to integrate the transcodes back into our timeline. Now in other apps this could be a tedious detour via a reconform, but in Nuke there is a very powerful option called „Build Track“. The whole operation can be thought of as an „invert“ for the Export Manager. In other words: Nuke Studio looks at an export preset from the Export Manager, searches the path specified there for files and places the shots in the timeline accordingly. Of course, it helps that the names of our transcodes match those in our timeline. So, right-click again on our timeline, select „Build Track – From Export Structure“ and there is a new pop-up: First of all, we can give our new track a name – Transcodes, for example. In the „Export Preset“ list we find all created export presets – unfortunately not sorted as clearly as in the Export Manager. We select our „EXR Export“ preset (We really don't need to have the Ref twice in the timeline.) and get the corresponding token string displayed. With „Build“ we confirm this process and our timeline is gaining one new track.

Not only that, but a new bin named „Transcode“ is created in the Project Bin and filled with the appropriate clips. Only the Color Transform might be a bit off, it's a

pity that Nuke Studio can't read this from its own Export Preset. But that's no problem, we just select all the clips in our Transcodes bin and right-click Set Media Input Transform and select the corresponding Color Transform, i.e. the one we selected when creating the Export Preset.

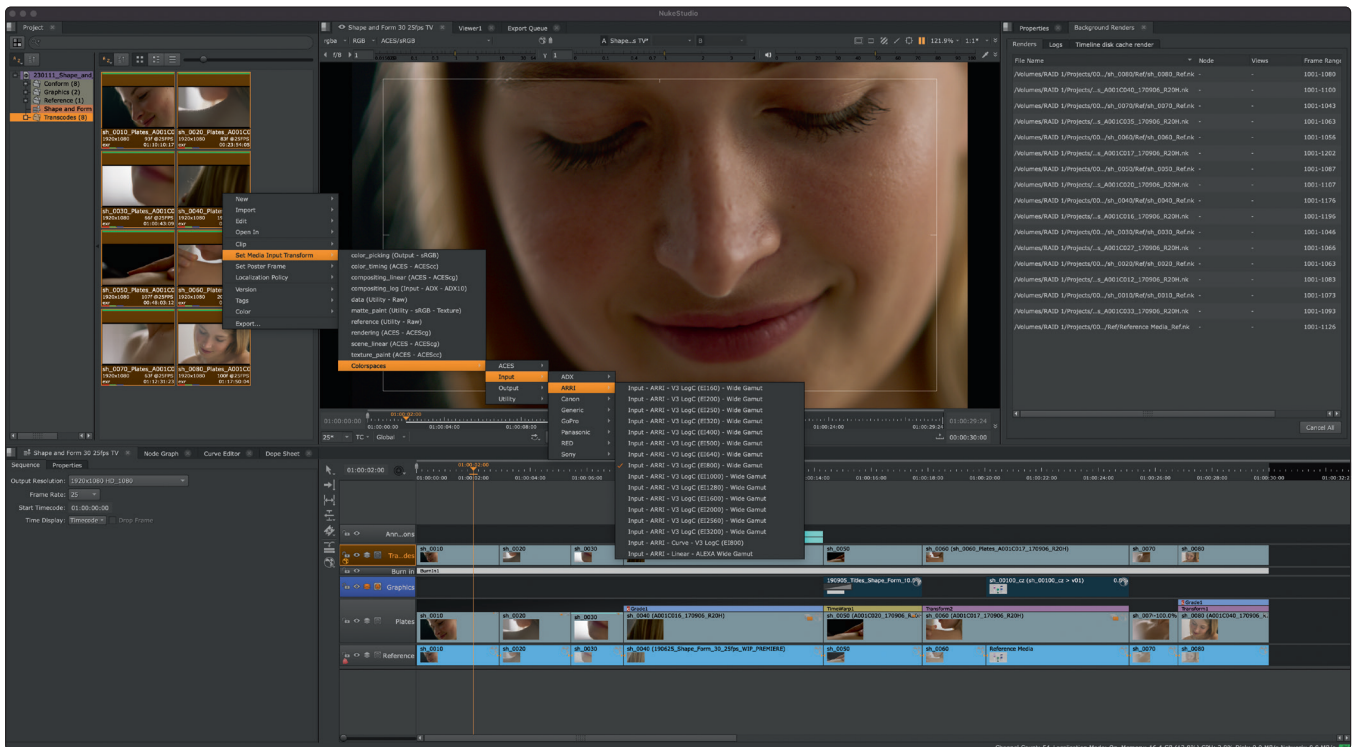
By the way, the „Build Track“ is also very, very helpful for something else: bringing assets from other departments or applications into Nuke Studio: No matter if CGI Rendering, Motion Graphics or Previs – as long as they are in a consistent folder structure, it is possible to import these files automatically. To do this, simply select „External Render“ as the content when creating the export preset. And even if it sounds a bit absurd: You can also create „Export Presets“ solely for importing things.

Building Comps

Everything we've gone through in this workshop so far can also be done in Hiero without a problem. But the concept of Nuke Studio – and Nuke Indie – is that you can do not only finishing but also compositing within the same application. For this, we need to discuss two basic workflows: „Create Comp“ and „Create Comp Special“. With „Create Comp“, the basic idea is to mimic the behavior of apps like Flame or Resolve, meaning Nuke takes care of the media management in the background and leaves the creative part to the artist. „Create Comp Special“, on the other hand, is more about putting as

Name	Date Modified
> 01_Rushes	5. January 2023 at 08:45
> 02_Edit	5. January 2023 at 08:46
> 03_Shot_Publish	Today at 09:05
> sh_0010	Today at 09:05
> Plates	Today at 09:05
> sh_0020	Today at 09:05
> Plates	Today at 09:05
> sh_0030	Today at 09:05
> Plates	Today at 09:05
> sh_0040	Today at 09:05
> Plates	Today at 09:05
> sh_0050	Today at 09:05
> Plates	Today at 09:05
> sh_0060	Today at 09:05
> Plates	Today at 09:05
> sh_0070	Today at 09:05
> Plates	Today at 09:05
> sh_0080	Today at 09:05
> Plates	Today at 09:05
> 04_Application_Data	Yesterday at 10:03
> 05_Grafics	12. January 2023 at 18:16
> 06_Render	5. January 2023 at 09:59
> 07_Grading	5. January 2023 at 08:45
> 08_Audio	5. January 2023 at 08:45
> 09_Documents	5. January 2023 at 08:45
> 10_Export	Yesterday at 09:26

And this is the result . . . Everything neatly sorted away thanks to our token string.



We still have to help a little with the Color Transforms.

much control as possible over the location of comp, annotation and rendering in the hands of the artist to accommodate the needs of larger projects and timelines.

Create Comp

Before we start just creating comps haphazardly, we should take a look in the Menu Bar under „Project – Edit Settings“ to see what the settings under the „Export“ tab look like. Here, too, we are asked for a track name, specifically the one where the renderings of our comps should go in the timeline. Let’s stick with „VFX“. Then there is the Export Directory – here we can choose the Project Directory or specify our own custom location. This is where Nuke Studio will then create folders – based on one of the export presets – for the Nuke scripts and the renderings that will end up back in our timeline. So we should be aware of that, depending on the size of the project, quite a lot of data ends up here. We can select these presets under „Shot Preset“ and for starters we choose „Basic Nuke Shot with Notations“. If we now select a single shot in our timeline and send it to a „Create Comp“ by right-clicking, something very similar to our „Build Track“ function happens: A new bin is created in the Project Bin with the name of the track, i.e. „VFX“. A new track with the same name appears in our Timeline and there is also a rather lonely clip – but it is displayed in black. This is because there is no rendering of this newly created scene yet.

If you want to see it immediately, right-click and select Render Comp. This happens in the background and should be relatively quick, since we haven’t done anything in the comp. If you are curious, double-click on the red clip (the red indicates the status as „Not rendered“) and

you are in familiar territory: Finally back in the Node Tree. But we see that all our work has paid off. With backdrops and metadata nodes, Nuke Studio has prepared our entire comp, we even find the annotations we may have created for the shot and of course a write node that is then set up according to the „Basic Nuke Shot with Annotations“ preset and the path we set in the Project Settings. And should the clip have had soft effects, these have of course also been moved over to the node graph. Whether we want to go crazy in our freshly created comp or are just happy that our setup is up and running, at some point we’ll go back to the timeline. The best way to do this is to switch back to the finishing workspace.

If we have already rendered the comp in the node tree, our clip is now green, if not, we can render it now as described above. But who is already satisfied with a V1? We just doubleclick on our green comp again, get back to the node tree, select „File – Save New Comp Version“ from the menu bar and make a few changes. Don’t worry, the Write Node will be adjusted in the same way as that. When we return now, just like in Nuke, we can save the clip with „Alt-Up Arrow“ up

or „Alt-Down Arrow“ down. And if you lose track, you can use the shortcut „V“ on the clip to view all versions of this clip and select one.

And what about shots that consist of multiple tracks? Well, if our timeline is reasonably well organized, that shouldn’t be a problem. Just select the overlapping elements and press „Create Comp“. A popup will appear asking us if we want to create a separate comp for each track or a common one. Of course we want a common one, but before we confirm the selection, one thing is important. The track we select, that is marked orange, is the basis for our node graph, that is, it determines the length and resolution of the comp.

And here you go, when we double-click on our script, we see our selected clip as the base layer for our comp and to the right of it backdrops with the corresponding additional clips.

This is of course already convenient, meaning you don’t have to take care of all the read and write nodes manually anymore. But what is actually happening under the hood now? Well, let’s take a look at the export preset we used here, „Basic Nuke Shot with Annotations“. We can see the whole thing when we open our Export Manager by simply pretending that we want to export our timeline again. Let’s take a closer look at the path: A {shot} folder is generated directly in the root folder of the project. Three folders are created in this folder, namely „script“, where a Nuke Project file is also stored. Then

„renders“, where the content „Nuke Write Node“ is used to define this node as render destination. Last but not least, Nuke Studio creates an „Annotations“ older, because Nuke needs a place where it can make the annotations available to artists who don't use Nuke Studio. The „Nuke Annotations File“ is basically a Nuke script in which the paint strokes and keyframes are stored, which are then clearly displayed in the actual comp in

the annotations node. This sounds a bit redundant, but it is actually a bit tidier and better for performance. But Nuke also needs at least the possibility to store a rendered sequence of the annotations. In the preset, these are 10-bit Dpx files and since they are just the annotations, there is little reason to change them.

If you want to use Nuke Studio like Flame or Resolve and don't want to deal with folder structures, then this workflow is an easy way to get started. The preset works, and Nuke Studio naturally finds the things it stores on its own. And of course, as we saw at the beginning, you can use another preset, even your own or a modified one, for this workflow. This makes sense if you have build setups „on-the-fly“, for example if you want to quickly edit or supply a shot that wasn't intended for VFX initially. But let's think a bit bigger.

This doesn't mean Hollywood, huge pipeline, fifty comp artists, but rather: We have a project where we have multiple VFX shots that need to be laid out neatly and work within our existing file structure on our server. That's where the next workflow comes into play...

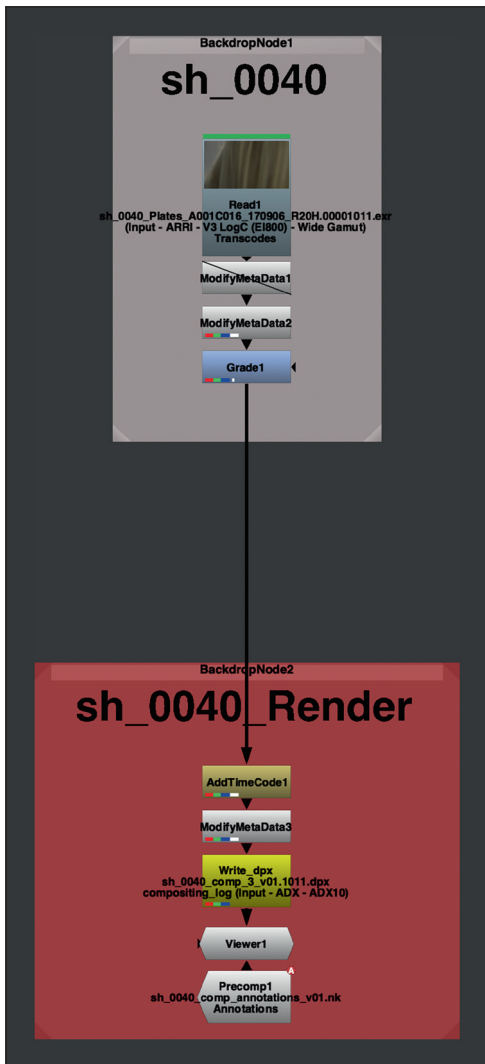
Create Comp Special

Admittedly, the work flow described above works quite well, but it is not necessarily possible to integrate it into any existing file structure. In the following case example, we want to export all this not into a shotbased, but a taskbased file structure. That means: There is a dedicated place for our plates – but we already did that in the transcoding -, there is a place for our scripts and another one for our renderings. We also want to fix a few flaws of the original setup: Renderings shall be put in separate version folders, but the annotations in a subfolder of the Nuke Scripts. And in this example we want to leave the soft effects on the timeline and not put them into the comp. Let's go!

Before we dive into the creation of our pre-set, we need to take care of one simple prep step on the timeline: All tracks that should not be part of the nuke script of the respective shots, we switch to invisible. For example, the track with the original plates, but also the burn-in. After that we can begin:

We mark all shots on the „Transcodes“ track that we want to export. Only on the Transcodes track? What about the references, possible inserts or second passes? That's why we just controlled the visibility of the timeline and set all unwanted tracks to invisible. Since we are now starting a batch processing, we need another way than the selection on the timeline to tell Nuke Studio to combine multiple shots. If we already have several tracks selected Nuke Studio will not clearly recognize which tracks the corresponding Nuke Script should refer to. Instead of selecting „Export“, we select „Create Comp – Create Comp Special“ in the right-click menu, but end up back at our old buddy, the Export Manager. We have already practiced several times how to create such a structure, so only the

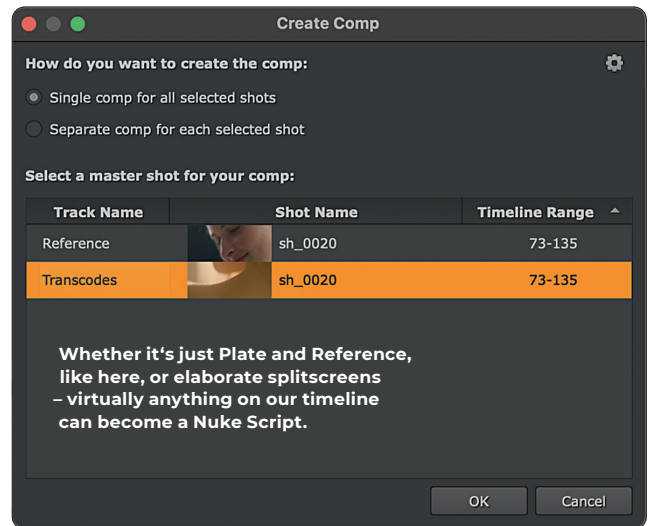
important changes will be highlighted in the following: In our existing file structure, the two folders „O4_Application_Data“ and „O3_Nuke“ already exist. Nuke Studio will recognize this during export and will not create new folders with this name, unless they do not exist yet. In this example, a task-based folder structure, we only want the setup files to go here, so we create a folder for each shot in our timeline, in which we then create a Nuke script with the token pattern {shot}_comp_{version}.{ext}. The version token here will be the space that will be versioned up when we execute „Save

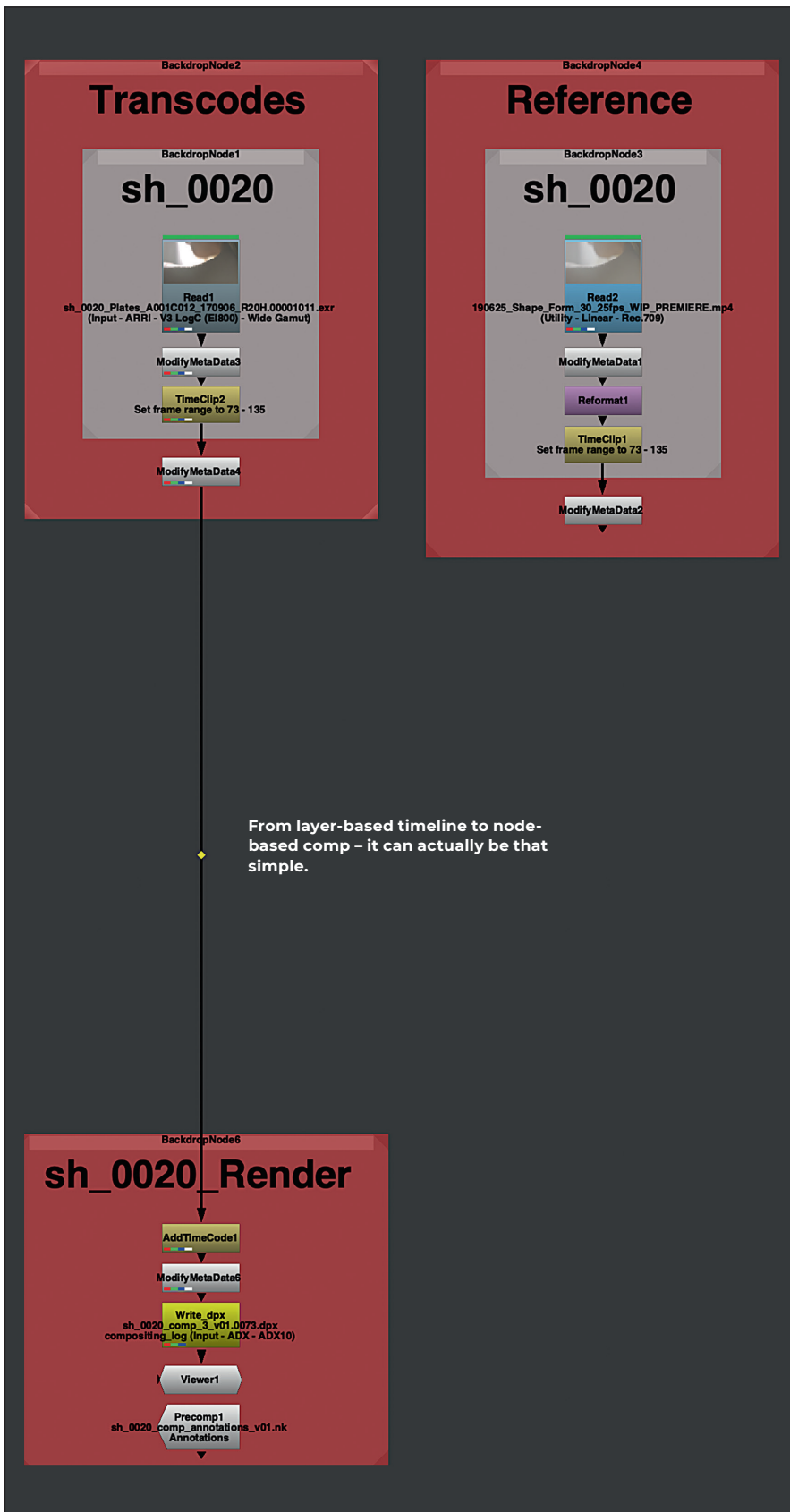


A nice script automatically build for us ... not bad at all!

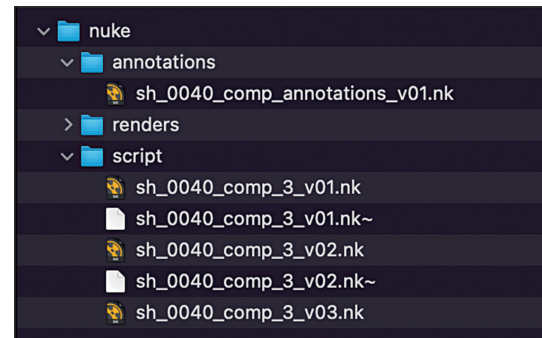
	sh_0040_comp_3_v03 1920x1080 176f @25FPS nk colour
	sh_0040_comp_3_v02 1920x1080 176f @25FPS nk colour
	sh_0040_comp_3_v01 1920x1080 176f @25FPS nk colour

Multipel Versions displayed in a nice Overview.





From layer-based timeline to node-based comp – it can actually be that simple.



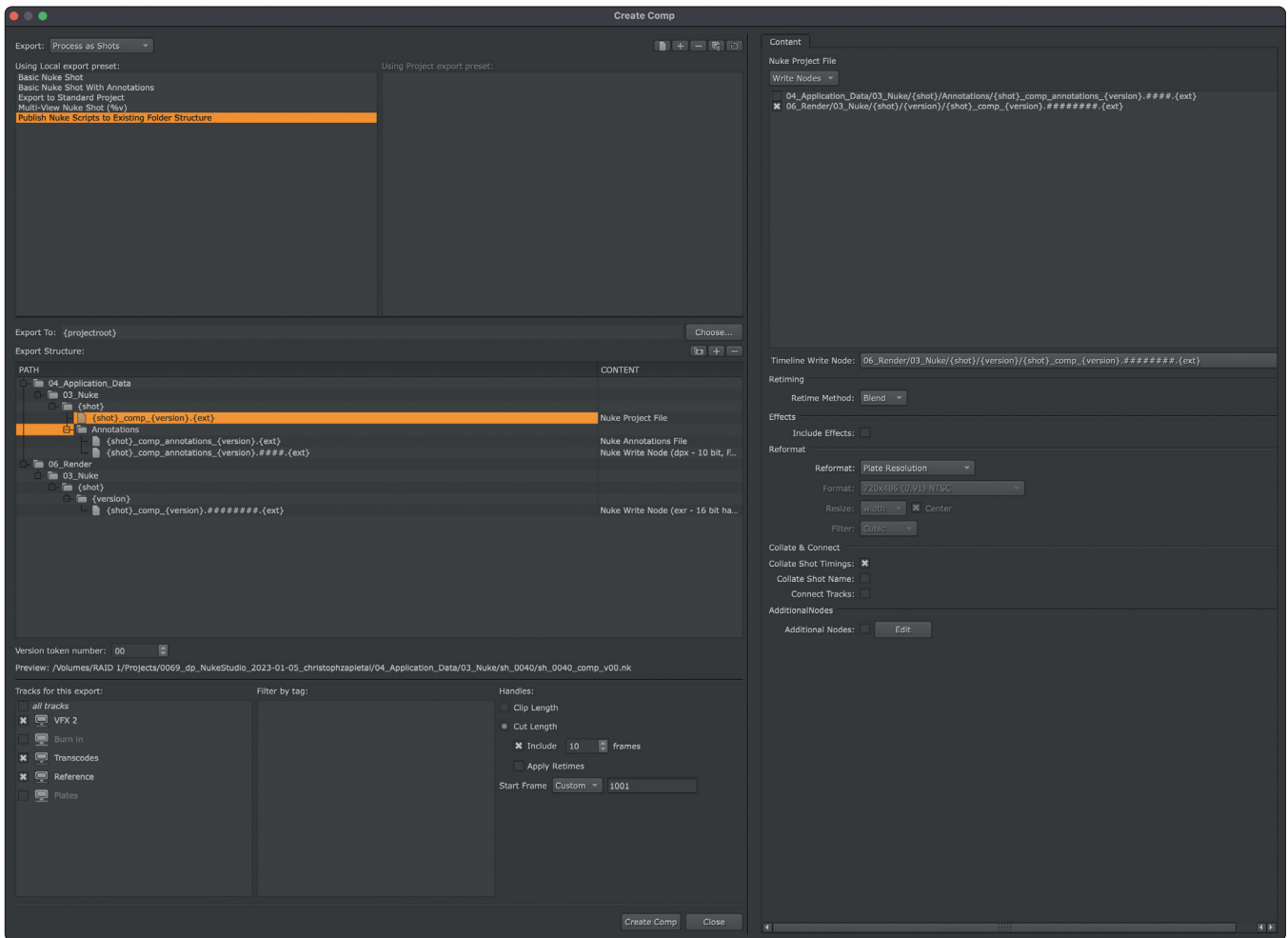
And here the result of a „Basic Nuke Script“ Export

New Comp Version“ in the Node Graph. We leave the content parameters of the Nuke Project File aside for a moment, first we define all the other elements we need.

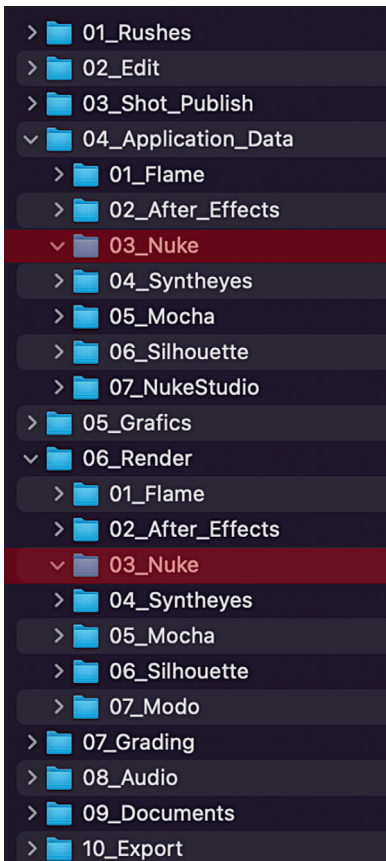
Within our Shot Folder we also create an „Annotations“ folder – for simplicity’s sake we also include the path for the annotation renderings here, but these could also be in a separate location if desired. Now to our „Render“ path. The beginning is very similar to our path for the Nuke Project File. First two fixed folders, then one with a dynamic token for the shot name. But then we add – as planned – a new folder with {version} token, which will now make sure that every rendering of a new version will be done in a separate folder. This makes housekeeping a lot easier.

Now we finally turn to our Nuke Project File, click the entry in the „Path“ window and check the following settings: At the top right we find a pulldown which is set to „Write Nodes“ by default. Here we can select the Write Node we just defined in the Render Folder as well as the one with the DPX files for the annotation. We choose the first one, because only this one should be put back into the timeline of Nuke Studio. If you at this moment have the idea to automate everything even more and to include the transcoding step in this setup, you should be warned: This will not be possible without further ado and will lead to unexpected error messages. There should be no check marks here, whereas under „Annotations“ the check mark should be placed on the actual „Annotations“ file and not on the correspon-

PATH	CONTENT
{shot}	
nuke	
This structure defines the path into which the exported content will be written. See the tokens listed within the tooltip to build unique paths for each	
renders	
{shot}_comp_{nameindex}_{version}.###.ext	Nuke Write Node (dpx - 10 bit, False, True, (auto detect))
annotations	
{shot}_comp_annotations_{version}.ext	Nuke Annotations File
{shot}_comp_annotations_{version}.###.ext	Nuke Write Node (dpx - 10 bit, False, True, (auto detect))



Export Preset – Next Level. Everything goes where it should.



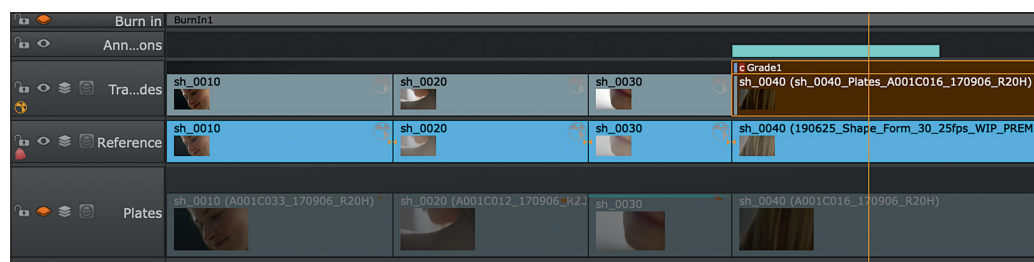
ding DPX sequence. As mentioned above, we want to leave the soft effects on the timeline and not include them in the Nuke script, which is why we do not check „Include Effects“.

We comp on the actual plate resolution like all the cool kids do and now come to the most important part: „Collate & Collect“. So this is about putting things together for the comp. We are talking explicitly about everything vertical in the timeline. We have made all the things we don't want to take

into the comp invisible on the timeline. If we select „Collate Shot Timings“, Nuke Studio will search for clips on all other visible tracks, starting from the track we have selected, and include them in the comp. In our case the Quicktime reference. That's why the option „Connect Tracks“ doesn't make sense for us, because we don't want to connect the reference to the main stream in our node graph. However, if we had inserts or CG passes here, that would be a very practical option.

The task-based file structure into which Nuke Studio should store its setups.

We can also utilize „Collate Shot Na-



me", after all, we copied cuts to our reference at the beginning and gave the respective clips the same name as the plates and the transcodes. Last but not least we have the possibility to add additional nodes to each script we create now. Here we also have the possibility to define if they are already connected in the node graph or if they should be parked at the side of the graph without connection. The operation is not quite intuitive, but with a little practice it is very useful. With one final press on „Create Comp" you are ready to go. Almost. We had to decide whether we wanted to merge all shots into one comp or create a separate script for each shot. And now it becomes clear why we had to mute tracks and use „Collate". If we were to combine all shots, we would have a huge comp over the length of the entire timeline. But since we are not megalomaniacs, we choose „Separate Comp for each selected Shot". From here on, Nuke Studio diligently creates folders and scripts in our File Structure and then links the outputs to our Timeline, just as before. Now that we have gone through both workflows, perhaps a few words of clarification: Both workflows are not strictly

separated from each other, but can be combined very well with each other. For example, you can use „Create Comp Special" to create a wonderful preset for the current project and then use it as the basis for your project. Create an initial publish and then use the same export preset with „Create Comp" to publish additional shots if necessary during the course of the project. . . You know your customers.

And even though Nuke Studio does a lot of pipelining and house keeping for the artist, you should know exactly what it's doing „under the hood". But it offers interesting possibilities: For example, you can build an export preset, which not only the folders relevant for Nuke, but for the

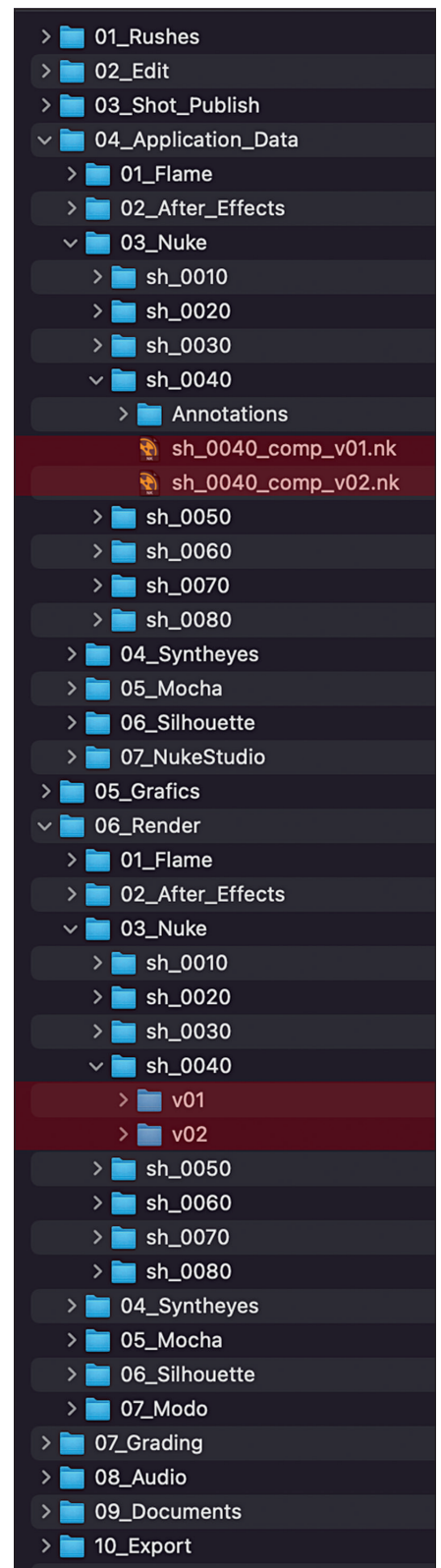
whole pipeline. Especially for people who work without a dedicated asset manage-

Just as we like it: Scripts neatly named, folders for each version.

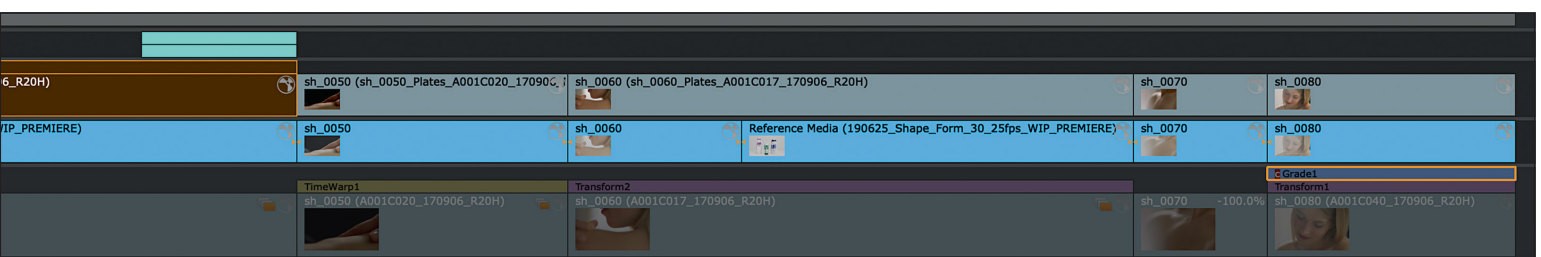
ment system like FTrack or Shotgun this is something to think about.

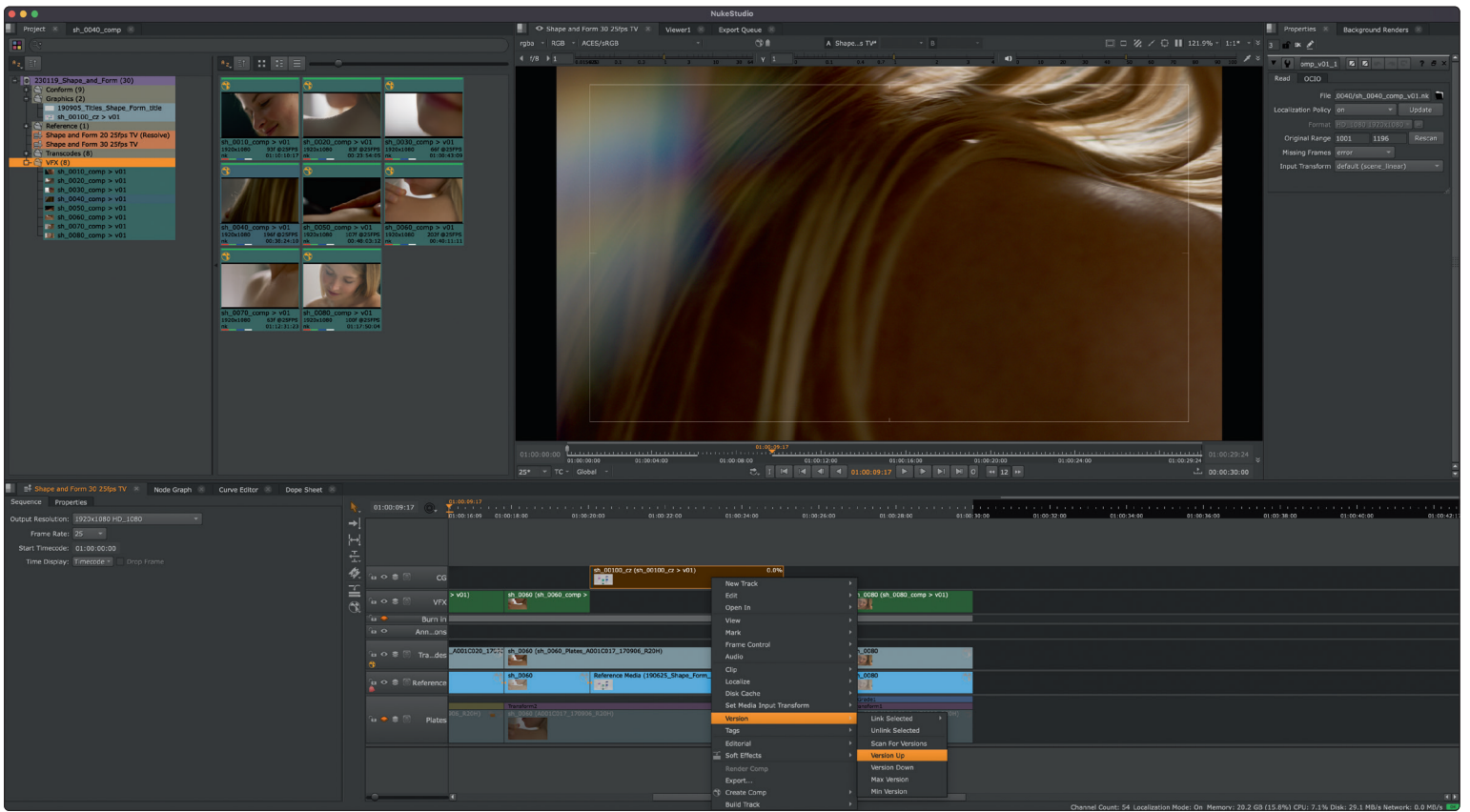
Versions

Of course, the more we work in our comps, the more versions we create and render. We already have some experience with versions and know how to up- and down-version them with the shortcuts we know from Nuke. But there are still a few tricks that are important when we get renders that are not directly linked to a Nuke script, for example because they come from another application and we have to render them using the previously explained „Build Track" process. To find these versions on its own, Nuke Studio lacks (at least out-of-the-box) the direct connection that exists in a Nuke-only workflow. However, there is a good workaround for this. Within the right-click menu, we can click on „Versions – Scan for New Versions". Nuke Studio will now scan the folder referenced by the selected clip(s) and let us know how many new versions are available. In order for the whole thing to work properly, however, you must of course always work within the folder and naming convention specified in the original „Build Track" procedure. Another feature worth mentioning is „Max Version" . In which situations is this feature useful? To give just one example: You come back to your Nuke Studio session after a hard-working comp team has done a great job, and you tag your entire VFX track, have it scanned for new versions, and – still with the same selection – choose „Max Version" and you're up to date in every way. And then there's the scenario where a clip has been used more than once. Either cut into the timeline multiple times or – those who work in advertising know this well – spread over several edits. Here it is worthwhile to look into version linking a bit. A simple experiment shows this quite won-

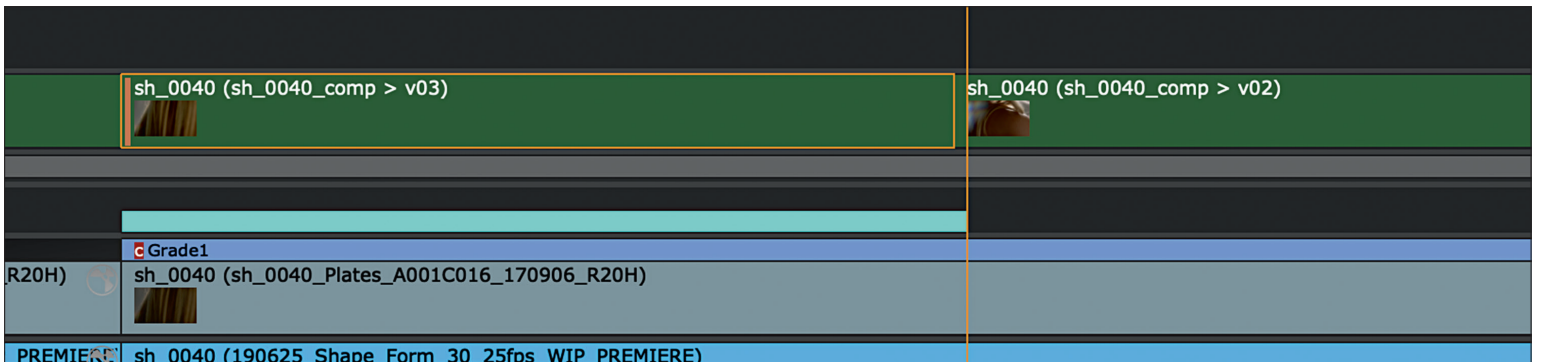


The visibility of layers is quite relevant for the creation of comp setups.





This way it works with other software and versions....



Versions can be linked or unlinked at any time.

derfully. We create several versions in our edit on one shot, three should be enough for now. Then we take the „Razor“ tool (hotkey „T“) and cut the clip in the middle.

If we now change the version of the clip, whether via the „V“ hotkey or via the „Alt“ and the arrow keys, both clips simultaneously change the version. But now we make a change on one of the clips. Under the right-click menu we select „Versions – Unlink Version“. Now this one clip on which we have executed the function is independent from the rest of the clip. We need to stress this point: Only the one clip is detached from the sync. All other clips, including those in other timelines or the clip in the Project Bin, continue to be up- and down-synced. If we want to restore the sync, we can either relink to

the clip in the Project Bin or to another clip. Even with these few options, complex version orgies can be handled. It doesn't matter if it's about localizations or customers who don't like to make decisions...

There's a lot more to discover in Nuke Studio, but that's where we quickly get into areas where we need to consider the individual pipeline, the used asset manager, or simply the personal way of organizing jobs, makes the way Nuke Studio is used very individual. But for a good start into Nuke Studio, this workshop should be a solid foundation. Have fun conforming, editing, finishing . . . and of course comping.

This text has been translated with the help of A.I. and is probably not representing the authors actual style of writing.



Christoph Zapletal has been working as a freelance compositing artist and VFX supervisor in Hamburg for several years. He has worked with Autodesk Flame as well as The Foundry's Nuke on various commercial and feature film projects. He is also a teacher for fxphd and at the HFF Munich. More info at: www.christophzapletal.de