

Bachelorarbeit

Falk Hofmann

Audiovisuelle Medien WS 2010/2011

Hochschule der Medien Stuttgart

Erstprüferin: Prof. Katja Koepl

Zweitprüfer: Dipl. Ing. Peter Ruhrmann

Thema:

[METADATENBASIERTE

VFX PIPELINEOPTIMIERUNG]

Zusammenfassung

Die vorliegende Arbeit erforscht die Verwendung von Metadaten in der Filmproduktion mit speziellem Augenmerk auf der Postproduktion in Verbindung mit Visual Effects. Es wird erläutert wie diese Zusatzinformationen den Arbeitsablauf beschleunigen und vereinfachen können. Weiterhin wird beschrieben, wie diese gespeichert, verwaltet und bearbeitet werden können, um sie auf den jeweiligen Produktionsprozesses abzustimmen.

Neben einem theoretischen Abschnitt umfasst diese Arbeit ein Beispiel einer solchen Lösung in einem praktischen Teil. Das Beispiel stellt die Kombination aus einer eigenständigen Software und diversen Erweiterungen des Compositingprogramms „Nuke“ von „The Foundry“ dar. Ziel der selber entwickelten Software und Skripte ist es zu belegen, dass diese nicht nur theoretisch, sondern auch praktisch anwendbar sind. Die Software wurde von Januar bis März 2011 im Rahmen einer realen Filmproduktion getestet. Somit konnten Vor- und Nachteile der zuvor präsentierten theoretischen Pipeline erkundet und mit anderen Workflows verglichen werden. Zudem wird auf das Archivieren und Kennzeichnen von Material bei der Erstellung von Innerbetrieblichen Bibliotheken eingegangen und es wird erläutert, was man bei diesen Arbeitsschritten beachten sollte.

Inhaltsverzeichnis

Zusammenfassung	02
Inhaltsverzeichnis	03
Abkürzungsverzeichnis	05
Glossar	06
Abbildungsverzeichnis	07
Einleitung	08
1. Geschichte der Metadaten	08
1.1. allgemein	08
1.2. Dublin Core	10
2. Metadaten Workflow	12
3. Erläuterung der Pipeline	14
3.1. Der Dreh	14
3.1.1. Kameraparameter	15
3.1.2. Kamerabewegung	16
3.1.3. Kameraposition	16
3.1.4. MovieSlate	17
3.2. Der Schnitt	19
3.3. Visual Effects	22
3.3.1. Namensgebung	22
3.3.2. Datenverwaltung	23
3.3.3. Stock Footage/Archivmaterial	23
3.3.3.1. Archivieren – Ordnerstruktur	24
3.3.3.2. Archivieren – Keywords	25
3.3.3.3. Ordnerstruktur <> Keywords	27

3.3.4. XML Panel	26
3.3.4.1. Python	27
3.3.4.2. PyQt	28
3.3.4.3. Nuke	28
3.3.4.4. XML Tool	29
3.3.4.1.1. allgemein	29
3.3.4.1.2. detailliertes Funktionsprinzip	31
3.3.5. Nuke Panel	33
3.3.6. Datenbanken versus Dateibasierte Speicherung	36
3.3.6.1. Datenbanken	36
3.3.6.2. Dateibasierte Speicherung	38
3.3.7. Metadaten im OpenEXR Format	38
3.3.7.1. Kameradaten	38
3.3.7.2. Compositing Skripte	40
3.3.7.3. Hyperlinks	41
4. Abschlussbetrachtung	42
4.1. Fazit	43
4.2. Zukünftige Erweiterungen	44
4.3. Weiterführende Gedanken	45
Eidesstattliche Erklärung	46
Anhang	47
Quellenverzeichnis	48

Abkürzungsverzeichnis

API	Application Programming Interface
CSV	Comma-Separated Values
CVKC	Controlled Vocabulary Keyword Catalog
DCMI	Dublin Core Metadata Initiative
DIT	Digital Image Technican
etc.	Et cetera
EXIF	Exchangeable image file format
ggf.	gegebenfalls
GPS	Global Positioning System
HTML	Hypertext Markup Language
IPTC	International Press Telecommunications
IPTC IIM	International Press Telecommunications Council Information Interchange Model
NCSA	National Center for Supercomputing Applications
OCLC	Organisatoren Online Computer Library Center
VFX	Visual Effects
WI-FI	Wireless Fidelity
XML	Extensible Markup Language
XMP	Extensible Metadata Platform

Glossar

Compositing	Ist das Zusammenfügen unterschiedlich aufgenommene Bildebenen oder computergenerierten Bilder zu einem homogenen Bild
Daguerreotypie	Fotografie-Verfahren aus dem 19. Jahrhundert
Header	Zusatzinformationen am Anfang eines Datenblocks
Proxys	Komprimiertes auflösungs- und farbtiefen reduziertes Bildmaterial von den Rohdaten
Re-framing	Das nachträgliche Verschieben des Bildausschnittes
Set Extension	Digitale Erweiterung des gedrehten Sets
Shader	Beschreibt die Oberflächeneigenschaft innerhalb 3D-Applikationen
Shot	Aufnahme zwischen zwei Schnitten
Stock Footage	Archivmaterial welches nicht explizit für einen Film gedreht wurde
Tags/Tag	Eintrag im XML
Timecode	Zeitreferenz während der Aufnahme
VFX-Supervisor	Kreativer Leiter eines Visual Effects Studio
Visual Effects	Manipulationen des Bildes in der Postproduktion
VRay	Software zur Generierung eines Bildes aus einer 3D-Applikation

Abbildungsverzeichnis

Abb. 01	http://www.pureblendsoftware.com/movieslate
Abb. 02	http://www.pureblendsoftware.com/movieslate4
Abb. 03	http://www.pureblendsoftware.com/movieslate5
Abb. 04	Screenshot der erstellten Ordnerstrukturen
Abb. 05	Screenshot des Programms XnView
Abb. 06	Screenshot des zu erstellten Programms XML_Tool
Abb. 07	Screenshot des zu erstellten Programms XML_Tool
Abb. 08	Screenshot der XML Ordnerstruktur
Abb. 09	Screenshot des Programms Nuke
Abb. 10	Screenshot des Programms Nuke
Abb. 11	Screenshot des Programms Nuke
Abb. 12	Screenshot des Programms Nuke
Abb. 13	Screenshot des Programms Nuke
Abb. 14	Screenshot des Programms Nuke

Metadaten

Einleitung

In einer Zeit, in der die Auflösungen von digitalen Kameras und Filmsscans immer größer werden, wachsen die Datenmengen zeitgleich um ein Vielfaches mit. Hierbei ist es von essentieller Bedeutung, die Daten akkurat und kontrolliert zu verwalten, zu katalogisieren und zur Verarbeitung bereitzustellen. Um dieses möglichst automatisiert zu lösen, bieten sich sogenannte Metadaten an, welche entweder bei der Aufnahme, dem Scannen oder in der weiteren Bearbeitung erzeugt werden. Metadaten sind separierte Daten, die den eigentlichen Datensatz beschreiben. Es sind folglich Informationen über die Daten, in diesem Fall über das Filmmaterial. Die Metadaten können die üblichen, wie beispielsweise Herstellungsdatum, Farbtiefe, Autor oder Ähnliches sein. Digitale Kameras geben heutzutage über 200 verschiedene Metadaten aus. Problematisch ist hierbei, dass viele Personen, die mit digitalen Kameras arbeiten, diese Metadaten gar nicht kennen und somit ihr Potential nicht nutzen können.

1. Geschichte und Standards:

1.1. allgemein

„Meta“, griechisch (μετά) bedeutet in diesem Zusammenhang: „mit“, „zugleich mit“, „zusammen mit“, „in“, „bei“ *¹. Metadaten können somit als Daten die den eigentlichen Daten mit beigefügt sind definiert werden. „Daten über Daten“ ist die allgemeine und häufigste Definition. Das Konzept der Metadaten ist keineswegs neu. Vielmehr gibt es im Bereich der Fotografie bereits solange wie die Fotografie selbst. Wurde zum Beispiel auf eine Daguerreotypie der Name des Fotografen oder der des Abgebildeten eingekratzt, so sind das Metadaten in ihrer ursprünglichen Form. Auch im weiteren Verlauf der Fotografie hielten Metadaten vermehrt Einzug in die alltägliche Herstellung von Bildern. So hat es sich in den 1990er Jahren durchgesetzt, dass das Erstellungsdatum direkt von der Kamera mitgeliefert wurde. Nachdem man die analogen Filme entwickelt hatte, befand sich das Datum der Ablichtung entweder auf der Rückseite des Bildes oder auf Fotografie selbst. Seit dem Beginn des digitalen Zeitalters gibt es nun keine Bildrückseite mehr, auf der man Metadaten

¹ http://de.wikipedia.org/wiki/Liste_griechischer_Wortst%C3%A4mme_in_deutschen_Fremdw%C3%B6rtern

vermerken kann. Heutzutage werden verschiedene Standards verwendet, um diese Daten zu sichern. So gibt es beispielsweise das International Press Telecommunications Council Information Interchange Model (IPTC IIM) welches in den Anfang der 1990er Jahren entwickelt wurde um den Austausch von Bildern und Texten zwischen Nachrichtenagenturen zu erleichtern und zu beschleunigen.

Die Softwarefirma „Adobe“ veröffentlichte 2004 ihr eigenes Schema zur Metadatenverwaltung, welches den bisherigen IPTC Kern beinhaltet und diesen weiterverarbeiten kann. Das von „Adobe“ entwickelte Schema heißt Extensible Metadata Platform (XMP) und ist ein auf Extensible Markup Language (XML) Sprache basierendes Schema, welches das Verwalten der Daten erneut vereinfachen sollte. IPTC und XMP wurden vorrangig für das weitere Verarbeiten von Daten wie beispielsweise Schlüsselwörter zum Archivieren vergeben geschaffen. Zudem gibt es einen weiteren Standard, um die Kameraeigenschaften festzuhalten, unter denen die jeweilige Ablichtung erstellt wurde. Dieser Standard, der von den meisten Herstellern verwendet wird, heißt Exchangeable Image File Format (EXIF). Diese sind zwar die gängigsten Standards, aber es gibt noch zahlreiche andere. Genau hier liegt ein essentielles Problem der Metadatenverwaltung in der heutigen Zeit. Es gibt viele Standards, aber keiner wird einheitlich von allen verwendet. Andrew S. Tanenbaum von der Vrije Universität, Amsterdam fasst diese Problematik zusammen: “The nice thing about standards is that there are so many of them to choose from.”²

Das derzeitige Defizit ist eine uneinheitlichen Definition und Handhabung von Metadaten. Eine optimale Lösung wäre eine weltweit einheitlich und detaillierte Namenskonvention, die angibt, welche Daten standardmäßig mitgeliefert werden und wie diese namentlich gekennzeichnet sind. Eine weitere Lösung wäre eine Erweiterungsmöglichkeit, da in der Praxis die Informationen, die man von Consumer- bis Profigeräten geliefert bekommt, kaum ausreichen. Als Erweiterung können Terme angesehen werden, die die Daten weiterführend und präziser als die Standardausdrücke beschreiben und somit das Maß an zugänglichen Informationen erhöhen. Eine erste Basis bildet dafür das Dublin Core Model.

² <http://www.quotationpage.com/quote/473.html>

1.2. Dublin Core

Die Dublin Core Metadata Initiative (DCMI) wurde 1994 in Chicago gegründet.³ Interessierte der Informationsbeschreibung beschlossen, eine Konferenz im März 1995 abzuhalten um das Thema der Metadatenverwaltung- und Erweiterung zu erörtern. Diese fand in Dublin /Ohio statt und wurde nach den Organisatoren Online Computer Library Center (OCLC) und dem National Center for Supercomputing Applications (NCSA), OCLC/NCSA Metadata Workshop benannt. Im Rahmen der Konferenz wurde eine Grundmenge an Begriffen festgelegt, die der Beschreibung von Webressourcen dienen und nannten diese Begriffssammlung nach dem Tagungsort: *Dublin Core Metadata*. Diese beinhalteten fünfzehn Terme, welche allgemein die eigentliche Datei beschreiben. Sie sind so angelegt, dass sie von Suchmaschinen interpretiert werden können und dass die Dokumente anhand dessen katalogisiert werden können. Diese fünfzehn Beschreibungen bilden noch heute die Grundlage der Metadaten in den meisten Betriebssystemen. Im Folgenden sind die 1994 festgelegten fünfzehn Begriffe aufgeführt.

01: Title	Titel des Dokuments
02: Creator	die letzte Person die das Dokument überarbeitet hat. Muss keine Natürliche Person sein.
03: Subject	Thema des Inhaltes in Suchwörtern, die optimaler weise einem bestimmten Schema folgen
04: Description	selbstverfasste Kommentare oder kurze Inhaltsangabe, muss keinem Schema folgen
05: Publisher	Verleger, Herausgeber, auch hier muss es nicht, sollte aber eine Natürliche Person sein
06: Contributor	Personen, die einen Beitrag zur Erstellung des Dokumentes geleistet haben
07: Date	Zeitpunkt der Erstellung oder der letzten Änderung, kann aber auch spezialisierter ausgedrückt werden: <i>dateCopyrighted, created, dateSubmitted, modified, dateAccepted, issued, available</i> und <i>valid</i>
08: Type	bezeichnet die Art des Dokuments kombiniert mit einem Terminus

	wie beispielsweise: Sound, Image, Text, Software, Dataset etc.
09: Format	gibt an, womit was das Dokument dargestellt beziehungsweise geöffnet werden soll
10: Identifier	Identifizierung des Dokuments nach einem passenden Katalog
11: Source	Quellen, von denen das Dokument (ganz oder zum Teil) abgeleitet wurde
12: Language	Sprache des Dokumenteninhaltes
13: Relation	verweist auf Objekte, die mit dem Dokument in Beziehung stehen
14: Coverage	zeitliche, sprachliche, inhaltliche und/oder regionale Eingrenzung des Inhaltes
15: Rights	Definition der Rechte oder ggf. Verweise auf Lizenzbedingungen

Das Dublin Core Schema bietet somit eine wesentliche Basis zur Festlegung von Metadaten im Internet oder auf unterschiedlichen Betriebssystemen. Es wird als Kern (engl. Core) bezeichnet, da es sich auf die jeweiligen Bedürfnisse erweitern und anpassen lässt. So werden innerhalb von Windows verschiedenen Terme standardmäßig angezeigt und verwendet. Hierzu zählen beispielsweise das Kommentarfeld (*description*), das Änderungsdatum (*date*) und der Dateityp (*type*). Metadaten sind folglich nicht nur für das Internet gemacht. Vielmehr wurden sie, nachdem der zusätzliche Nutzen erkannt wurde für verschiedene Bereiche angepasst. In der vorliegenden Arbeit wird sich auf Metadaten in der Fotografie und im Filmbereich beschränkt, da nur diese Bereiche für die Forschungsfrage von Relevanz sind.

Die Fotografie ist ein Anwendungsbereich von Metadaten. So ist beispielsweise für einen professionellen Fotografen und dessen Materialarchivierungen eine strikte Ordnung und Definition der Metadaten notwendig. Ohne eine solch strukturierte Sortierung, muss der Fotograf mehr Zeit in die Suche von Bildern investieren, als in die Bearbeitung. Um diese Ordnung herzustellen, ist es notwendig, ein passendes Vokabular für die jeweiligen Zusatzinformationen zu entwickeln, damit man anhand dessen später beispielsweise die Suche nach bestimmten Bildern starten kann. Als gute Grundlage wird hier der Controlled Vocabulary Keyword Catalog (CVKC) von David Riecks empfohlen. In diesem sind über 11.000 Schlüsselwörter (engl. Keywords) vordefiniert und hierarchisch unterteilt. Auf diese Weise hat jedes Schlüsselwort seine eigene Hierarchie und Auswirkung auf die Suchoption. Dies

erleichtert die Suche. Es ist daher notwendig, ein definiertes (kontrolliertes) Vokabular zu verwenden, da sonst die Schlüsselwörter unterschiedlich sein können, obwohl sie dieselbe Bedeutung haben.

Im Folgenden sei ein Beispiel genannt: Wenn man ein Bild mit einem Wildschwein Suchbegriffe hinzufügen will, so kann man das mit „*Wildschwein*“, „*Schwein*“, „*Sau*“, „*Eber*“, „*Bache*“, „*Keiler*“, „*Eber*“, „*Ferkel*“ oder „*Frischling*“. Es zeigt sich, dass derjenige, der die Schlüsselwörter vergibt, verantwortlich dafür ist, dass man diese später auch sinnvoll verwenden kann. Um diesem Umstand vorzubeugen, benutzt man einheitliche und systematisierte Wörter. So würde an das obige Beispiel Schlüsselwörtern, wie: „*boar*“, „*male*“ oder „*female*“ vergeben. Es ist festzustellen, dass nicht nur die Wortwahl auf ein zuvor definiertes Vokabular zurückgreift, sondern auch die Sprache ausschlaggebend ist. Um in der heutigen Zeit des weltweiten Datenaustausches sicher zu stellen, dass die Schlüsselwörter auch überall Verwendung finden, sollte man englische Begriffe verwenden. Ein geordnetes Begriffsvokabular in englischer Sprache, das einheitlich verwendet wird, ist daher die Grundlage für eine effiziente Nutzung von Metadaten.

2. Metadatenworkflow in der Visual Effects Postproduktion

In den vergangenen zwanzig Jahren hat der Anteil von Visual Effects (VFX) enormen Zuwachs in vielen Sparten der Filmindustrie verzeichnen können. Es sind nicht nur mehr reine „Effektfilme“, die Aufmerksamkeit generieren. Zunehmend beinhalten Filme unsichtbare Arbeiten, die dem Zuschauer nicht auffallen. Beispiele hierfür sind Retuschen, Set Extensions, Blue/Greenscreenshots, Re-framing oder Ähnliches. Auch wenn solche Aufgaben meist nicht viel Arbeit bedeuten, so wird der Datenumfang dennoch um ein Vielfaches höher. Die Ursache hierfür ist, dass meist für jeden Shot Proxys erstellt werden und diese mehrere Iterationen des Renderns durchlaufen, bevor dieser zum Kunden gesendet wird.

Um diesen Umfang an Daten leicht verwalten zu können, ist es wichtig, eine strikte Ordnung zu wahren. So sollten Ordnerstrukturen definiert und Namenskonventionen festgelegt werden an diese sich jeder Mitarbeiter strikt halten muss. Wenn diese Bedingungen erfüllt sind, ist es möglich man eine Pipeline darum aufzubauen. Als Pipeline wird hier der Arbeitsablauf bezeichnet, welcher die Reihenfolge der Arbeitsschritte regelt. Um die Postproduktionsmitarbeiter (folgend Artist) zu entlasten und den Vorgang zu beschleunigen,

werden an dieser Stelle oftmals Automatismen eingebaut, welche meistens bei der jeweiligen Firma entwickelt und auf die eigene Ordnung und Namenskonvention abgestimmt sind.

Im Folgenden soll aufgezeigt werden wie, eine solche Pipeline basierend auf Metadaten gesteuert und abgestimmt werden kann. Das Wichtigste in der Pipeline ist die Informationsabfrage und Weitergabe. Um diese Schritte zu gewährleisten, darf nicht erst in der Postproduktion damit begonnen werden, diese zu sammeln und zu verwalten. Die strukturierte Arbeitsweise muss schon beim eigentlichen Dreh berücksichtigt werden. Bislang werden Script-Schreiber und VFX-Supervisor selbst damit beauftragt, sich die Kameradaten zu erfragen und zu notieren. Ziel der eigen entwickelten Pipeline ist es jedoch, diese Schritte zu automatisieren und so Fehler auf Basis der strukturierten automatisch erstellten Notizen auszuschließen. Digitale Kameras geben heutzutage Unmengen Metadaten aus.

Problematisch ist jedoch, dass kaum eine Person, die damit arbeitet weiß, wie man diese effektiv verwenden kann. Daher bleibt es zwar die Aufgabe des VFX-Supervisors, die Daten zu sammeln und in den späteren Produktionsprozess einzubinden. Das Notieren mittels einer strukturierten Arbeitsweise, wird jedoch nicht mehr auf Papier oder mit Hilfe eines privaten Laptops vollzogen, sondern die Daten werden eigenständig in ein extra dafür vorgesehenes Dokument eingetragen. Dieses Dokument sollte bei den eigentlichen Daten, sprich dem Filmmaterial, gelagert werden, da das Metadatendokument direkt mit den Filmmaterial verbunden ist und dies näher beschreibt. Zudem ist bei einer solchen Arbeitsweise gewährleistet, dass die Metadaten im weiteren Verlauf der Filmproduktion nicht abhanden kommen. Als eine solche Dokumentformatvorlage wird bei der Kamera „Arriflex Alexa“ ein XML-File mitgeliefert, welches die technischen Aspekte der Aufnahme sowie den Pfad zum File auf dem Aufnahmemedium beinhaltet. Zudem werden die Dauer und der Name des jeweiligen Takes sowie die Information, welche digitale Filmrolle (Speicherkarte) verwendet wurde, vermerkt.

Die gespeicherten Informationen können anschließend in den weiteren Filmproduktionsprozess übernommen werden. Dies kann direkt aus dem File oder anhand eines Tools geschehen, welches die Informationen in ein Format mit entsprechender Struktur bringt. Ein derart strukturiertes Dokument kann im späteren Verlauf als zentrales Element zur Verwaltung der Daten dienen. Letzteres steht im Fokus der vorliegenden Arbeit.

3. Erläuterung der Pipeline

Im Praktischen Teil der vorliegenden Arbeit soll als zentrales Bedienelement, ein Softwaretool stehen, welches eine XML-Datei verwaltet und editiert. Dies geschieht auf einer speziell strukturierten Grundlage, welche das Verwalten an sich ermöglicht. Da es bislang keine Software gibt, welches auf genau diese Ansprüche ausgelegt ist, wurde im Rahmen dieser Arbeit ein eigenes Tool programmiert.

Hierfür bedarf es zunächst einige Recherche darüber, welche Programmiersprache gute Anbindungen zur XML-Bearbeitung hat. Um darüber hinaus die Anbindung an Postproduktionstools zu erhalten wurde hier Python gewählt, da die meisten Softwarepakete in der Film-Postproduktion diese Sprache nativ unterstützen. Auf Python wird in Abschnitt 3.3.3.1. näher eingegangen.

Um eine Pipeline aufzubauen oder zu verbessern, sollten zunächst die Ziele der Pipeline festgelegt werden. Hierzu zählen neben dem Hauptziel der Datenübergabe, auch Aspekte, die automatisiert, beschleunigt oder vereinfacht werden sollen. Das Hauptziel der vorliegenden Untersuchung besteht darin, Informationen während des gesamten Produktionsprozesses zu erhalten, jederzeit und jederorts abrufbar zu machen und sie gegebenenfalls zu verändern. Das praktisch entwickelte Beispiel dieser Arbeit setzt voraus, dass die VFX-Arbeiten in einem Studio vollzogen werden und dass Filmmaterial in Form von Rohdaten geliefert wird.

3.1. Der Dreh

Das Sammeln und Bereitstellen von Metadaten beginnt nicht erst bei der Nachbearbeitung, sondern direkt beim Filmdreh. Die wichtigsten zu notierenden Informationen sind hierbei die Daten bezüglich der Kamera. Auf diese wird in Abschnitt 3.1.1. näher eingegangen.

Desweiteren sollten auch Informationen wie beispielsweise die Umstände des Filmdrehs festgehalten werden. So kann es mitunter relevant sein, wie das Wetter während des Filmdrehs und wie die Lichtsituation während der Aufnahme war. Alles das sind Metadaten des Filmdrehs.

3.1.1. Kameraparameter

Matchmove heißt der Vorgang, wenn im späteren Verlauf der Produktion Kameradaten rückwirkend berechnet werden müssen. Um die Arbeit des Matchmove-Artists zu vereinfachen beziehungsweise überhaupt erst möglich zu machen, sollten alle Parameter der Kamera katalogisiert werden. Diese Aufgabe übernahm bei analogen Aufnahmen noch das sogenannte „Scriptgirl“. Es wurden alle Informationen wie beispielsweise Brennweite, Blende und Umlaufblende beim Kameraassistenten angefragt und sorgfältig notiert. Bei digitalen Aufnahmen bietet sich eine Automatisierung dieses Prozesses an, da bei aktuellen digitalen Kameras die meisten Parameter automatisch ausgegeben werden. Diese sind jedoch nicht alle abgespeichert. Häufig werden diese Informationen unbeachtet gelassen und gehen somit verloren, bevor das Filmmaterial den Schneiderraum erreicht.

Es ist daher für eine effektivere Nutzung der Metadaten notwendig, die Informationen, welche die Kamera zur Verfügung stellt, zu nutzen und vor Allem zu speichern. Einige digitale Filmkameras speichern diese bereits in Form einer XML-Datei oder als Notiz im Header der Rohdaten. Auch hier werden jedoch nicht alle Daten gespeichert, sodass eine individuelle Notierung und Speicherung erforderlich ist. Ziel sollte es im Anschluss an die Datenspeicherung sein, dass der Digital Image Technican (DIT) diese Daten verwaltet und im späteren Verlauf der Postproduktionsfirma zur Verfügung stellt. Dieser Arbeitsschritt ist die Aufgabe eines DIT, da dieser für die Bilddaten zuständig ist und Metadaten zu den Bildinformationen gezählt werden, sollte dieser Arbeitsschritt in seinen Aufgabenbereich fallen. Der Perfekte Workflow wäre dergestalt konzipiert, dass die Kamera all diese Daten selbstständig abspeichern würde. Eine weiterer Lösungsansatz wäre, die Metadaten in eine externe Datei zu schreiben und in diesem wiederum auf das Originalmaterial verwiesen wird. In einem Falle, in dem anschließend das Trennen der Daten im Anschluss an den Filmdreh nötig wird, ist die zuletzt genannte Methode von Anfang an zu wählen.

3.1.2. Kamerabewegung

Die Bewegung der Kamera während der Aufnahme ist ebenfalls eine wichtige Information in der VFX-Produktion. Sind die Daten vorhanden wie sich die Kamera am Set bewegt hat, wird der Arbeitsschritt des Matchmoves stark erleichtert. Die Idee der Metadatenutzung in der VFX-Produktion folgt dem Prinzip des Motion Controls. Dieses Prinzip beschreibt eine spezielle Technik, die hauptsächlich dazu verwendet wird, ein und die Selbe Kamerabewegung mehrfach ohne Abweichung zu wiederholen. Die Bewegungsdaten können allerdings auch ohne die meist teure Spezialtechnik erfasst werden. So verfügt beispielsweise „Panavision“ an allen ihren Kamerakränen (Technocranes) einen „Data Port out“, an den man eine Daten-Capture-System anschließen und somit alle Bewegungen des Kranes aufnehmen kann⁴. Diese Informationen können wiederum später an das Matchmove-Department weitergeleitet werden. Durch diesen Prozess der Kamerabewegungsspeicherung wird Arbeit des Matchmoves um ein Vielfaches erleichtert.

3.1.3. Kameraposition

Eine weitere hilfreiche Information am Set ist die Kameraposition. In der Vergangenheit wurde und selbst heutzutage wird diese, wenn überhaupt, durch Skizzen festgehalten. Diese wiederum, sind sehr ungenau und sind nur selten mit Maßangaben versehen. Dadurch fällt die Orientierung bei größeren Sets ziemlich schwer wenn man in der Postproduktion nur das gedrehte Material ohne zusätzliche detaillierte Informationen über die Kameraposition geliefert bekommt. Heutzutage, in einer Zeit, in der Global Positioning System (GPS) bei Mobiltelefonen oftmals Standard ist, sollte es kein Problem darstellen, die Kameraposition mittels GPS auf den Meter genau bestimmen zu können. Zudem verfügen diverse Mobiltelefone über einen integrierten Kompass, mit dem man auch in einem Zug die Position und Richtung notieren kann. Darüber hinaus sollte ein Foto aus derselben Kameraposition aufgenommen werden, um später in der Postproduktion prüfen zu können, ob die Kamera tatsächlich in der gleichen Position steht, wie das gedrehte Material. Man kann dadurch mit wenig Aufwand am Set, die kompletten Kamerapositionen nachvollziehen und sich ggf. auch eine „Shotkarte“ erstellen. Eine solche bildet das Set aus der Vogelperspektive ab und beinhaltet gerichtete Markierungen, welche Aufnahme an welcher

⁴ Interview mit *Dave Stump* auf [http://www.creativecow.net/magazine.creativecow.net/article/metadata-the-future-of-filmmaking](http://www.creativecow.net/magazine/creativecow.net/article/metadata-the-future-of-filmmaking)

Position des Sets gedreht wurde. Dies ist durchaus nützlich bei Set-Extensions, dem digitalen Erweitern der Filmkulisse in der Postproduktion bei denen auf Kontinuität geachtet werden muss. Mit Hilfe von Kamerapositionsdaten und den dazugehörigen Bildmaterial lässt sich beispielsweise ein Panorama für eine 360° Erweiterung erstellen. So wird erleichtert, dieses Panorama zu rotieren, bis es die gewünschte Himmelsrichtung zeigt. Durch diese Vorgehensweise ist die Kontinuität der Erweiterung gesichert und Anschlussfehler können weitestgehend ausgeschlossen werden.

Auch nützliche Applikationen für das iPhone können die Datenerfassung am Set unterstützen. Für die Pipeline der vorliegenden Arbeit, wurde das Programm „MovieSlate“ gewählt. Da „MovieSlate“ bereits einige Funktionen integriert hat und man selbstdefinierte Zusatzinformationen speichern kann, bildet dies eine gute Basis. für die Entwicklung einer eigenen Pipeline. Es wird im Folgenden näher dargestellt-

3.1.4. MovieSlate

„MovieSlate“ ist eine mobile Applikation für das iPhone welche von „PureBlend“ vertrieben wird. Der Hauptbestandteil des Programms ist das Speichern aller relevanten Shot-Daten und zusätzlichen Notizen in ein passendes Dateiformat. Eine digitale Klappe (engl. Clapperboard) und die Timecode-Synchronisation bilden hierbei die Grundfunktionen. Die Arbeit mit einem Handy ist nicht die übliche und eine bislang nicht professionalisierte Weise der MetadatenSpeicherung. Aufgrund der Mobilität und der zahlreichen Anwendungsmöglichkeiten stellen Mobiltelefone jedoch ein praktisches Hilfsmittel dar, welche großes Potential für die zukünftige Verwendung aufweisen.

Clapperboard:

Sobald die digitale Klappe (inklusive Ton) durch „MovieSlate“ geschlagen wurde, werden alle Informationen wie beispielsweise „Timecode IN“ gespeichert.



Abb. 01: MovieSlate - Klappe

Timecode:

Die Synchronisation der Timecodes, Rollnummer, Takes etc. erfolgt durch Bluetooth und Wireless Fidelity (WIFI) innerhalb verschiedener Apple Produkte wie I-Phone, I-Pad und I-Pod-Touch. Es gibt jedoch auch eine Variante, den Timecode direkt von der Kamera zu übernehmen und ihn auf die anderen Arbeitsgeräte zu übertragen.

Shot Notes:



Abb. 02: MovieSlate – Shot Notes

Während der Shots können schriftliche Notizen, aber auch Sprachaufnahmen mit oder ohne Bilder aufgezeichnet und zugeordnet werden. Vor allem bei komplexen Notizen erleichtert dieses später den Umgang mit den Informationen. Bei längeren Takes ist es zudem möglich Markierungen mit einem bestimmten Zeitpunkt im Timecode zu verknüpfen.

Shot Log History:

Nach jeder geschlagenen Klappe erfolgt das Speichern in eine History. Einer Art Sammlung der generierten Informationen. Somit hat jeder Take seine eigene History in der all die Informationen abgelegt sind, die man benötigt. Die Gesamtheit aller Shots nennt man Shot Log History.

Export:

Um die gespeicherten Informationen im späteren Verlauf verwenden zu können, bietet das Programm die Formatierung in Hypertext Markup Language (HTML), Comma-Separated Values (CSV) oder einfachen Text an. Diese können direkt nach dem Speichern per E-Mail versendet werden. Somit kann auch das Back-Up sehr effizient gestaltet werden. Um die Daten automatisch auslesen zu können, wird nur ein kleines Skript benötigt, welches beispielsweise die HTML-Formatierung ausliest und

in ein firmeninternes Format umwandelt. Da es sich bei diesen Daten meist nur um Text und den Verweis (engl. Link) auf Bilder und Sprachaufnahmen handelt, stellt dies kein Problem bei der praktischen Umsetzung dar. Darüber hinaus gibt es die Option auf ein spezielles Final Cut Pro XML. Dies kann direkt vom Set in Final Cut importiert werden und darauf die Clips inklusive der kompletten Shot-Daten in Final Cut laden⁵. Eine Erweiterung der „MovieSlate“ Applikation stellt das automatische Abspeichern der Himmelsrichtung der Kamera dar, wenn man die GPS-Daten erfasst. So ist beispielsweise im aktuellen iPhone 4 ein Kompass integriert. Diese Funktion kann später zur Erstellung der Shotkarte dienen. Durch die GPS Daten ist die Position der Kamera und durch die Anzeige des Kompass die Ausrichtung der Kamera festgehalten. Da es nur um die Erstellung einer groben Shotkarte geht, sind geringe Abweichungen tolerierbar.

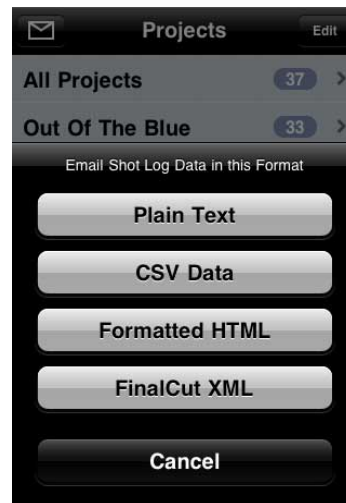


Abb.03: MovieSlate - Exporteinstellungen

3.2. Der Schnitt

Die Postproduktionsarbeit beginnt mit dem Kopieren der Daten vom Dreh auf den jeweiligen Server der Postproduktionsfirma. Da zum aktuellen Zeitpunkt noch nicht viele Postproduktionspakete nativ Rohdaten unterstützen und verarbeiten können, ist zunächst meist eine verlustfreie Konvertierung notwendig, um die Daten im späteren Verlauf bequem und optimal zu nutzen. Zudem ist es ratsam, die Daten nicht nur in ein Format zu konvertieren, sondern gleich in mehrere. So sollten beispielsweise gleich mehrere Proxys erstellt werden. Proxys sind auflösungsreduzierte Formate, die eine verlustbehaftete Kodierung haben. Hierbei besteht die Möglichkeit, plattformunabhängige und effizient kodierte Formate wie Quicktime und JPG-Sequenzen zu verwenden. Ebenfalls ist es empfehlenswert, auch diese Formate in verschiedene Auflösungen umzuwandeln. Dies wiederum ist sinnvoll, um ein schnelleres Arbeiten zu gewährleisten, da nicht bei allen Arbeitsschritten die volle Auflösung benötigt wird. Eine Abteilung, die von einer solchen Vorgehensweise profitieren könnte, ist die 3D-Abteilung. Diese muss basierend auf den

⁵ http://www.pureblendsoftware.com/movieslate_reports

Originalaufnahmen Objekte modellieren, animieren und Shader (Oberflächeneigenschaften) zuweisen. Oftmals reicht in diesem Fall die halbe Auflösung des kodierten JPG-Zustands. Die Erstellung der Proxy und das Verschieben der Daten an den jeweils richtigen Ort auf dem Server lässt sich anschließend effizient in Form von Skripten automatisieren. Zudem ist, wenn das Tool passend programmiert wurden ist, gewährleistet es, dass alle Dateien die strikten Namen- und Ordnerkonventionen einhalten. Bei einem manuellen Sortieren, Benennen und Verschieben wäre die Wahrscheinlichkeit um ein Vielfaches höher, einen Tippfehler oder ähnliche Fehler zu machen. Diese Fehlerquelle wird anhand von Skripten ausgeschlossen. Die gesamte Idee hinter einem derart automatisierten Vorgang ist, dass das Verschieben, Kopieren, Rendern und Benennen nicht mehr manuell geschehen muss. Das Einzige noch von Hand definiert werden müsste, ist angelieferte Datei welcher interne Shot ist, da oft die angelieferten Daten oftmals noch den Namen aus der Kamera haben. Ein Beispiel hierfür ist die Namensgebung bei einer „ARRI Alexa“: „A001C001_20100821_R1KM“. Solche Bezeichnungen können nur selten verwendet werden. Vielmehr sollte der Film in Sequenzen und diese wiederum in Shots unterteilt werden. Sind mehrere Firmen an der Produktion beteiligt, sollte überall die gleiche Vorgehensweise verwendet werden, um einen leichteren Austausch und eine effektivere Kommunikation zu gewährleisten. Die Namensgebung wird dann von dem verantwortlichen Editor angelegt. Dieser sortiert und verwaltet das Filmmaterial, welches zu den Visual Effects Studios ausgeliefert wird. Der Editor fungiert somit als Schnittstelle. Sobald die bearbeiteten Tonspuren und Shots von den Tonstudios und VFX Firmen eintreffen, ist er es, der diese in den aktuellen Schnitt einfügen muss. Da es jedoch mehrfach vorkommt, dass der Editor den Schnitt aktualisieren muss, ist es an dieser Stelle ebenfalls sinnvoll einen automatisierten Vorgang zu entwickeln. Ein Lösungsansatz wäre, einen Ordner anzulegen, in welchem der gesamte Film in Shots unterteilt ist. Auf diesen Ordner greift anschließend das Schnittprogramm zu. Erhält man neues Material von den Zulieferern, müssen man die Files exakt so benennen wie die vorherigen und die alten Daten mit diesen ersetzt werden. Wird die Schnittsoftware gestartet, greift diese in den meisten Fällen auf diese Daten zu. Somit ist der Schnitt aktualisiert ohne den eigentlichen Schnitt verändern zu müssen. Diese Methode ist aufwendig, da Back-Ups angelegt werden müssen. Zudem müssen immer wieder die Daten passend umbenannt werden.

Eine elegantere und effizientere Lösung ist eine Abfrage der Metadaten oder ggf. den Dateinamen selbst. Hierbei ist es das Ziel, den Schnitt auf Basis eines XML's anzulegen. Dazu muss das Schnittprogramm soweit modifiziert werden, dass dieses die Pfade und Dateien anhand der XML-Datei ausliest. Die XML-Datei wiederum muss dahingehend verändert werden, dass die neuen Pfade zu neuen Versionen eingetragen werden. Um bei diesem Arbeitsschritt Tippfehlern vorzubeugen und vor dem Erzeugen eines „illegalen“ XML's zu schützen, sollte hierzu ein eigenständiges Tool geschrieben werden, welches die Struktur und Syntax von XML-Files beachtet und dieses regelkonform verändert. Ein Teil dieses Tools ist Bestandteil der vorliegenden Arbeit und wird im späteren Verlauf näher erläutert (siehe Abschnitt 3.3.4.4.). Ein Lösungsansatz wäre beispielsweise den Ordner, in dem alle gelieferten Shots liegen, zu durchsuchen und zu kontrollieren, an welcher Stelle neue Versionen vorhanden sind, um anschließend in das XML zu übertragen. Dies erfordert lediglich eine Überprüfung des Dateinamens, wenn er eine Versionsnummer enthält oder eine Überprüfung anhand der Metadaten, wann die Datei kopiert beziehungsweise erstellt wurde. Diese Informationen kann man mit den aktuellen Einträgen des XML's abgleichen und anschließend entsprechend ändern. Dieser Prozess ist per Skriptsprache mit wenig Aufwand umsetzbar. Zudem kann noch hinzugefügt werden, dass der Renderprozess automatisch angestoßen wird und man den fertig gerenderten Film oder die Sequenz erhält, ohne das man das Programm manuell öffnen muss. Zusammen mit „geplanten Tasks“ (beispielsweise in Windows), kann man auf eine manuelle Kontrolle verzichten und stattdessen jeden Tag (oder falls nötig mehrmals am Tag), eine automatische Überprüfung durchführen und so die Schnittliste (XML) und den komplett aufgesetzten Schnitt aktualisieren. Diese ist eine Idee, wie man den Filmschnitt im Verlauf der Produktion vollautomatisch aktualisieren kann. Hierbei ist noch zu sagen, dass bei dieser Variante keine ästhetischen Änderungen wie Timing berücksichtigt werden, sondern der aktuelle Schnitt mit den Cliplängen ersetzt wird. Dadurch ist die Möglichkeit der Kontrolle während des manuellen Ersetzens der Clips nicht mehr gegeben. Es bleibt somit nur die Kontrolle des fertig gerenderten Films. Dennoch ist diese Automatisierung eine effiziente Art der Aktualisierung, da im Normalfall alles manuell abgearbeitet werden muss. Der Aufwand des Programmieren und Testens ist daher schnell kompensiert, wenn man dieses Vorgehen konsequent in den Produktionsprozess einbindet.

3.3. Visual Effects

Die Verwendung von Metadaten im Postproduktionsprozess nimmt immer mehr zu. So auch im Visual-Effects-Bereich. Metadaten können hierbei helfen, den gesamten Arbeitsablauf zu steuern und zu kontrollieren. Eine nützliche Information ist hier beispielsweise das Änderungs- oder Erstellungsdatum von Dateien sowie deren Namensgebung. Anhand von diesen Informationen können ganze Teile des Prozesses automatisiert werden. Eine Möglichkeit soll im Folgenden aufgezeigt werden.

3.3.1. Namensgebung

Die Benennung der einzelnen Shots ist wichtig um einen Überblick zu gewährleisten. Hierbei wird oftmals die Unterteilung in Sequenzen empfohlen. Dadurch können separierte Abschnitte des Films unterteilt und aufgespalten werden. Dies kann dazu verwendet werden um anfallende Arbeiten besser aufteilen zu können. So sind oftmals die Aufgaben an einer Sequenz ähnlich, da diese häufig am selben Ort gedreht wurden sind.

Der Metadaten-Workflow beginnt mit der Vergabe der Sequenz- und Shotnamen. Die Namensgebung sollte zwar manuell vorgenommen werden, allerdings kann das technische Umbenennen wiederum automatisiert werden, in dem ein Wörterbuch (engl. Dictionary) angelegt wird. Hierbei sind die mit den Rohdaten eingehenden Namen auf der einen Seite und die selbstdefinierten auf der anderen Seite vermerkt. Was nur noch von Hand gemacht werden sollte, wäre der Start eines Skriptes welches das Ändern der Dateien schließlich umsetzt. Aus der Originalbezeichnung „A001C001_20100821_R1KM“ kann im Rahmen dieses Vorgehens beispielsweise „BSP_010“ werden. Um den Bezug zu den Originaldaten auch ohne dieses Wörterbuch zu erhalten, ist es ohne Weiteres möglich, den Originalnamen und den Pfad auf den Server als Zeichenkette einer neuen Meta-Information in das umbenannte Element zu schreiben. Diese Vorgehensweise ist vor Allem dann nützlich, wenn am Ende der Produktion auf das Originalmaterial zurückgegriffen werden muss. Durch die mitgelieferte Information ist dies ohne größeren Aufwand lösbar. Aufwändiger hingegen ist, das vorliegende Material den gesamten Produktionsprozesses zurück verfolgen.

3.3.2. Datenverwaltung

Um die gesammelten Informationen zu speichern, zu verwalten und abzurufen bieten sich zwei verschiedene Modelle an. Zum einen ist es bei großen Datenmengen eine datenbankbasierte Lösung, welche Vorteile bietet wenn von mehreren Standpunkten aus mit einer Datenmenge gearbeitet werden muss. Zum anderen bietet sich eine dateibasierte Herangehensweise an, welche die Informationen in einer Datei (engl. File) speichert. Dies kann beispielsweise eine Excel Tabelle, ein HTML Code oder ein XML-File sein. Die in der vorliegenden Arbeit entwickelte Pipeline verwendet ein XML-File, welches zentral gespeichert und dadurch der Zugriff von verschiedenen Workstations gewährleistet wird. Ein Vergleich zwischen datenbankenbasierten und einer dateibasierten Speicherung wird später erläutert (siehe Abschnitt 3.3.6.).

3.3.3. Stock Footage/Archivmaterial

Bei Filmproduktionen sammelt sich über längere Zeit eine große Anzahl an Archivmaterial, Texturen oder Stock Footage an. Dies sind oftmals nicht spezifisch für einen Film gedrehte Aufnahmen, sondern bieten ein Repertoire an Standardelemente wie beispielsweise Rauch, Feuer, Funken etc. Um diese sinnvoll zu verwalten, ist ein durchdachtes System mit vordefinierten Schlüsselwörtern erforderlich.

Soll nach Jahren ein Archiv mit (neuen) Schlüsselwörtern versehen werden, so nimmt diese Aufgabe viel Zeit in Anspruch. Dennoch ist eine solche Strukturierung langfristig nicht nur sinnvoll, sondern sie stellt auch eine Steigerung der Produktivität dar, weil das oft langwierige Suchen entfällt.

Zur zweckmäßigen Erstellung eines neuen Archivs gibt es zwei Vorgehensweisen. Eine Lösung ist das Verwenden von Ordnerstrukturen. Hierbei werden die Ordner hierarchisch benannt und angelegt. Eine weitere Variante ist zwar ebenfalls mit einer Ordnerstruktur verbunden, basiert allerdings hauptsächlich auf der Verwendung von „Keywords“, Schlüsselwörtern, welche den Dateien zugeordnet sind. Nachfolgend werden beide Varianten näher beschrieben und anschließend miteinander verglichen.

3.3.3.1. Archivieren – Ordnerstruktur

Das Archivieren mittels einer Ordnerstruktur basiert auf den Ordernamen und deren Hierarchien. So kann man beispielsweise ein Ordner mit dem Namen „Archiv“ erstellt werden, in welchem das komplette Archiv abgelegt werden soll. In diesem wiederum kann beispielsweise ein Unterordner mit dem Namen „Texturen“ und ein Ordner mit dem Namen „Stock“ (für Stock Footage) erstellt werden. In gleicher Weise können so beliebig viele Ordner und Unterordner angelegt werden. Daten und Dokumente werden so auf die Ordner aufgeteilt und in den passenden Ordner verschoben. Je detaillierter man die Ordnerstruktur anlegt, desto kleiner wird die Anzahl der darin befindlichen Elemente. Das Prinzip hinter diesem Vorgehen erschließt sich schnell, birgt jedoch auch mögliche Fehlerquellen. So ist es beispielsweise problematisch, wenn zwei Ordner in beiden Unterordnern („Texturen“ und „Stock“) denselben Namen tragen. Eine solche Namensdopplung sollte vermieden werden, da es die Suche nach Ordnern und Inhalten erschwert. Um dieses Problem zu vermeiden, sollte von Beginn an eine strukturierte Ordernamensgebung erfolgen. Sinnvoll kann an dieser Stelle sein, eine Ordnerbezeichnung anhand spezieller Namenskonventionen vorzunehmen.

Diese Konvention kann beispielsweise folgendermaßen strukturiert sein: [Art-Element-Auflösung-Format]. Dieser Namenskonvention folgend würde ein Ordner „[stock-smoke-1920x1080-mov]“ heißen. Jedoch fällt die Unterscheidung mehrerer gleicher Elemente schwer. So ist es nicht möglich, zwei oder mehrere Elemente die ähnlich zueinander sind nur mit Ordernamen zu unterscheiden.

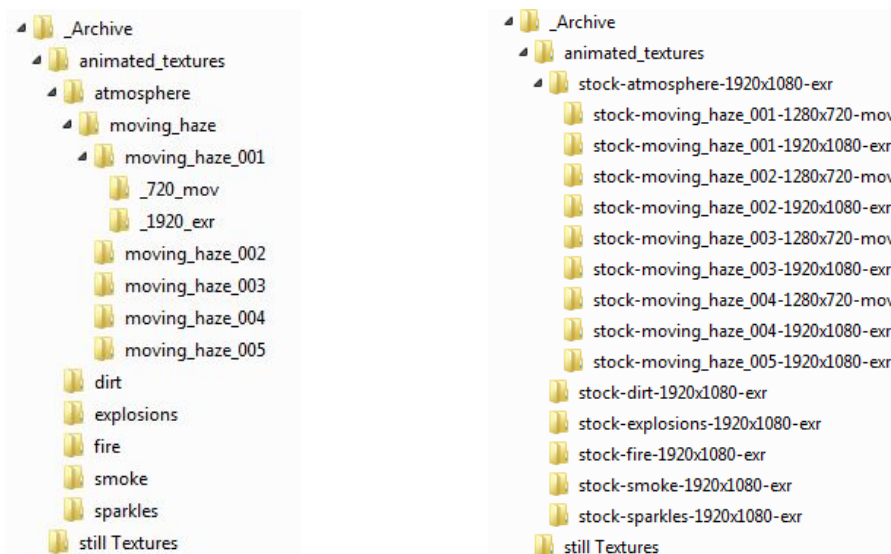


Abb.04: verschiedene Ordnerbenennungs- und Struktur

3.3.3.2. Archivieren – Keywords

Eine Archivierung mit Hilfe von „Keywords“ bietet die Möglichkeit, jedem Bild ein oder mehrere Tags zuzuweisen. Dies geschieht mittels einer speziellen Software für Metadatenbearbeitung. Bei dieser Vorgehensweise muss nicht jedes Bild einzeln mit Tags versehen werden. Vielmehr sollten mehrere Bilder ausgewählt und diesen ein bestimmtes „Keyword“ zugeordnet werden. Abhängig davon, wie detailliert Bilder beschriftet werden sollen, können noch zusätzliche Tags vergeben werden. So könnten einem Bild beispielsweise die Tags Explosion, Feuer, Flammen, Glut, Funken und Fackel zugeordnet werden. Bei jeder Suche dieser Schlüsselwörter werden nun alle Bilder angezeigt, die diesen Tags zugeordnet sind.

Um Tags zu vergeben, werden zwei verschiedene Vorgehensweisen verwendet. Eine Möglichkeit stellt die direkte Einbettung der Schlüsselwörter in die Daten dar. Hierbei sind vorrangig zwei Standards im Einsatz: IPTC und XMP von „Adobe“, wobei XMP der zurzeit meist verwendete Standard ist. Um die Tags einzubetten, gibt es einige freier Softwarepakete, wie beispielsweise XnView (siehe Abb. 05).

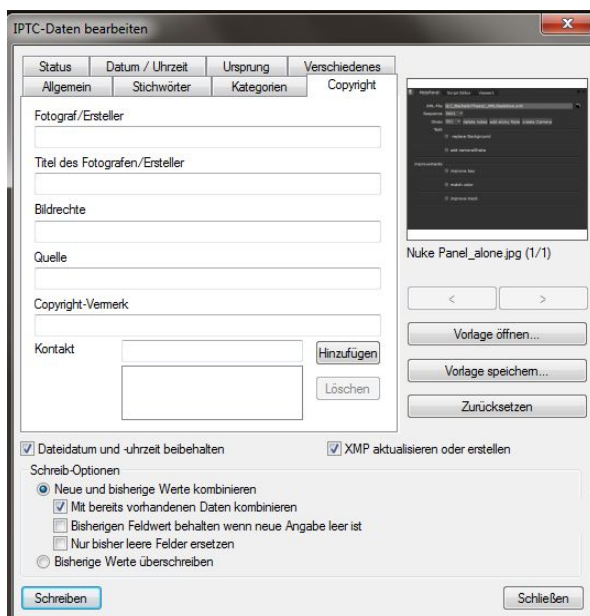


Abb. 05: IPTC-Daten bearbeiten in XnView

Eine andere Methode stellt die Verknüpfung der Daten mit Tags dar. Hierbei werden die „Keywords“ in einer Datei oder in einer Datenbank gespeichert und verweisen auf die Bilder. Diese Datenbank wird jedes Mal aktualisiert, wenn neue Tags vergeben werden. Die Vergabe

von Schlüsselwörtern ermöglicht eine schnelle Suche, da nur die Datenbank und nicht die Metadaten der Bilder selbst durchsucht werden muss.

Nachteile der „Tagging“-Methode ist, dass die Verwendung softwaregebunden ist und dass ohne die Software keine Suche stattfinden kann. Aufgrund der Softwaregebundenheit werden beim Kopieren von Daten – im Gegensatz zur Methode der direkten Einbettung der Informationen - keine Metadaten übernommen. Allerdings ist ein Umstieg von einer Software zur Anderen nur schwer umsetzbar, da diese ihre eigenen inneren Strukturen haben und eine Änderung der Archivierungsmethode eine längere Umstellungszeit benötigt.

3.3.3.3. Ordnerstruktur <> Keywords

Der größte Vorteil der Keyword-Methode ist, dass mehrere Schlüsselwörter an ein Element gebunden werden können. Dies ist bei der Ordnervariante nur möglich, wenn man die Daten in verschiedene Ordner kopiert, was wiederum aufgrund des Aufwandes und der Elementdopplung kaum sinnvoll erscheint.

Die Ordnerstruktur bietet jedoch auch einen erheblichen Vorteil. So ist man mit dieser Variante völlig softwareungebunden. Die Ordner selbst können vom Betriebssystem erstellt werden. Zudem kann das komplette Archiv kopiert werden, ohne dass die Suchfunktion beeinträchtigt wird. Kopiert man das Archiv bei der Tagging-Methode, treten Kompatibilitätsprobleme auf, sobald auf dem Ziel- und dem Ursprungscomputer nicht dieselbe Software installiert ist. Diese Softwaregebundenheit ist somit der größte Nachteil der Tagging-Methode. Es sollte sich daher vor Projektbeginn überlegt werden, welche Software benutzt wird, da man an diese während der ganzen Filmbearbeitung gebunden ist. Optimal ist eine Kombination beider Archivierungsmethoden, da man auf diese Weise die Vorteile beider Methoden auf Ordner- und Elementstruktur miteinander kombinieren kann.

3.3.4. XML Panel

Das Programm, welches im Rahmen der vorliegenden Arbeit geschrieben wurde, ist ein eigenständiges Softwaretool zur Übermittlung von Metadaten in die Compositingapplikation Nuke. Es basiert auf der Skriptsprache Python und speichert die Daten in einer speziell strukturierten XML-Datei. Die „Extensible Markup Language“ (XML) hat sich als „State of the Art“ entwickelt wenn es um speziell angepasste Speicherung geht. Ihre Struktur ist zwar vordefiniert, lässt dem Bearbeiter aber viel Spielraum, um eigene Bedürfnisse wie

beispielsweise die Namensvergabe und Hierarchien speziell anzupassen. So wird es vereinfacht, Daten zu organisieren und sie wiederum in einzelne Sequenzen und Shots zu unterteilen. Anschließend darauf können für jeden einzelnen Shot Daten eingetragen. Dies können beispielsweise Kameraparameter, Kommentare vom Dreh oder der Postproduktion sowie sonstige Informationen die während der Postproduktion relevant sind.

3.3.4.1. Python

Python ist eine objektorientierte Skriptsprache⁶ welche heutzutage von vielen Postproduktions- und VFX-Programmen unterstützt wird. Sie wurde zu Beginn der 1990er Jahre von Guido van Rossum am Centrum voor Wiskunde en Informatica in Amsterdam entwickelt.

1991 veröffentlicht, unterstützt Python die populärsten Betriebssysteme wie Mac OS, Windows und Linux/Unix. Da sie unter einer Open-Source-Lizenz verbreitet wird, ist die Skriptsprache kostenfrei nutzbar. Aufgrund ihrer leicht lesbaren Syntax und der dynamischen Definition von Variablen wird sie Programmieranfänger besonders gerne verwendet. Dies führte neben anderen Aspekten dazu, dass sich Python im europäischen und nordamerikanischen Raum als Skriptsprache etabliert hat. Python wurde anschließend schnell in die VFX-Produktion implementiert. Heutzutage unterstützen fast alle wichtigen Programme der Film-Postproduktion Python als Skriptsprache, hierzu zählen im 2D-Bereich „Eyeon's Fusion“, „Nuke“ von „The Foundry“, „Toxic“ von „Autodesk“ und „Shake“ von „Apple“ (PyShake) aufzuführen. Im 3D-Bereich unterstützen „Houdini“ von „Side Effects“, „Maya“ und „XSI“ von „Autodesk“ sowie „Realflow“ und das „Blender“ Softwarepaket Python. Um die Skriptsprache im entwickelten Standalone-Tool und die Erweiterungen in Nuke konstant zu halten, wurde auch in der vorliegenden Untersuchung Python gewählt. Das Kreieren der Benutzeroberfläche selbst wird mit einem Paket namens „TKinter“ ermöglicht, welches standardmäßig mit Python installiert wird. Da „TKinter“ nur schwer zu erweitern ist, um die gewünschte Oberfläche in der vorliegenden Arbeit zu realisieren, wurde stattdessen PyQt gewählt.

⁶ <http://www.python.de/>

3.3.4.2. PyQt

Qt ist eine plattformübergreifende Applikation und ein „User-Interface-Framework“, somit ein Rahmen innerhalb dessen die Oberfläche kreiert wird. Im Fall von Qt basiert dies auf einer C++-Klassenbibliothek. So lässt sich in Python, das auf Python spezialisierte PyQt implementieren. Zu beachten ist hierbei, dass das Einbinden an sich unkompliziert ist, die Verwendung von PyQt allerdings einiger Übung bedarf.

Die Benutzeroberfläche innerhalb von Python kann grundsätzlich auf zwei verschiedene Arten erstellt werden. Eine Vorgehensweise ist die klassische Art, das manuelle Programmieren. Dies wiederum ist aufwendig, da das Programm oftmals neu gestartet werden muss, um zuvor gemachte Veränderungen auch optisch wahrzunehmen. PyQt bietet hierbei eine schnellere Variante. PyQt besitzt ein Interface, in dem sich die zu erstellenden Programme optisch anlegen lassen, ohne einen Programmcode zu erstellen. So kann ein Interface schnell in groben Zügen zusammengestellt und anschließend als separierter Python-Code ausgegeben werden. Dieser Code wiederum muss später in das eigentliche Programm eingebunden und dem Interface müssen die passenden Funktionen zugeordnet werden. Der Vorteil dieser Methode ist, dass das Interface geändert werden kann, ohne bei jedem Schritt den eigentlichen Funktionscode manuell ändern zu müssen. Zudem besteht die Möglichkeit, ein erstelltes Interface in einen Testmodus laufen zu lassen, um eventuelle Fehler während der Oberflächennutzung zu erkennen. So können beispielsweise Fehler beim Verschieben oder Skalieren der einzelnen Fensterebenen sichtlich gemacht werden.

3.3.4.3. Nuke

Nuke ist ein von der US-amerikanischen Visual-Effects-Firma „Digital Domain“ entwickeltes Compositing-Programm. Es wurde seit 1993 als firmeninterne Software entwickelt und genutzt. Ab 2002 wurde es der Öffentlichkeit zugänglich gemacht und 2007 wurde es vom britischen Plug-in Unternehmen „The Foundry“ gekauft. Seit der ersten Veröffentlichung unter „The Foundry“ hat sich Nuke im VFX-Bereich zügig verbreitet. Auch die Weiterentwicklung fand seither in einem solch rasanten Tempo statt, wie man es nur selten im Softwarebereich antrifft. Unterstützt wird die vertriebene Software durch den engen Kontakt mit Postproduktionsfirmen, welche neue Funktionen vorab testen und direktes Feedback geben können. Seit Version 5.0 wird Nuke nicht standardmäßig von Python unterstützt, sondern basiert vorrangig darauf. Durch die Bereitstellung eines Nuke speziellen

Application Programming Interface (API) und zahlreichen Foren im Internet, die durch Mitarbeiter von „The Foundry“ unterstützt werden, ist die Anpassung von Nuke in die eigene Firmenpipeline oftmals unkompliziert durchzuführen.

3.3.4.4. XML Tool

Im Visual Effects Bereich ist es üblich, dass ein Shot mehrere Iterationen der Bearbeitung durchläuft in denen immer wieder Teile des Bildes geändert werden, bevor er zum Kunden geliefert wird. Um diese Änderungsinformationen vom Supervisor an den Artist effizient und ohne Verluste zu übermitteln, sollte dieses Vorgehen mit Hilfe eines Softwaretools aufgezeichnet werden.

3.3.4.4.1. Allgemein

Erfahrungen aus der praktischen Arbeit belegen, dass handschriftliche Notizen oft nicht vollständig sind, keine zeitliche Referenz haben und oftmals sogar verloren gehen. Etwaige Änderungen erreichen den Artist zeitverzögert wodurch dieser durch veraltete Angaben mitunter ein Ziel verfolgt welches gar nicht mehr der aktuellen Kritik entspricht. Daher sollte die Zusammenarbeit zwischen Supervisor und Artist möglichst so angelegt sein, dass keine Informationen verloren gehen.

Ist dies realisiert, kann sich der Artist bei den Besprechungen mit dem Supervisor auf das Bild konzentrieren, ohne nebenher alle Anmerkungen schriftlich festhalten zu müssen. Die Notizen werden von einer dritten Person direkt in das erstellte Programm geschrieben.

Die Anmerkungen werden ungefiltert und ohne Zeitverzögerung erstellt bzw. aktualisiert. Abgespeichert werden diese in ein speziell strukturiertes XML File. Um diese Struktur festzulegen und einzuhalten, kann anhand einer Ordnerstruktur welches die Shots beinhaltet eine XML Datei pro Projekt angelegt werden.

Die Software liest die Hierarchien der Ordner aus und legt für jeden darin liegenden Sequenzen und Shots jeweils einen Eintrag in dem XML an. Zusätzlich werden Standardeinträge erstellt. Als erstes sind das die allgemein Aufgaben die am Bild vorgenommen werden müssen. Sozusagen der Kundenwunsch. Dies wird als „Task“ bezeichnet. Hierzu als Beispiel eine Aufgabe wie „Key Person in Front of Blue/Greenscreen“ and replace the Screen with Footage to match the previous Shots“.

Der zweite Standardeintrag sind die Anmerkungen des Supervisors zu der aktuellen Version des Artists. Dieser Eintrag ist der am meisten genutzte, da dieser am häufigsten aktualisiert wird. In den meisten Fällen muss das Bild mehrere Iterationen mit Änderungen durchlaufen, bevor sie zum Kunden ausgeliefert werden können.

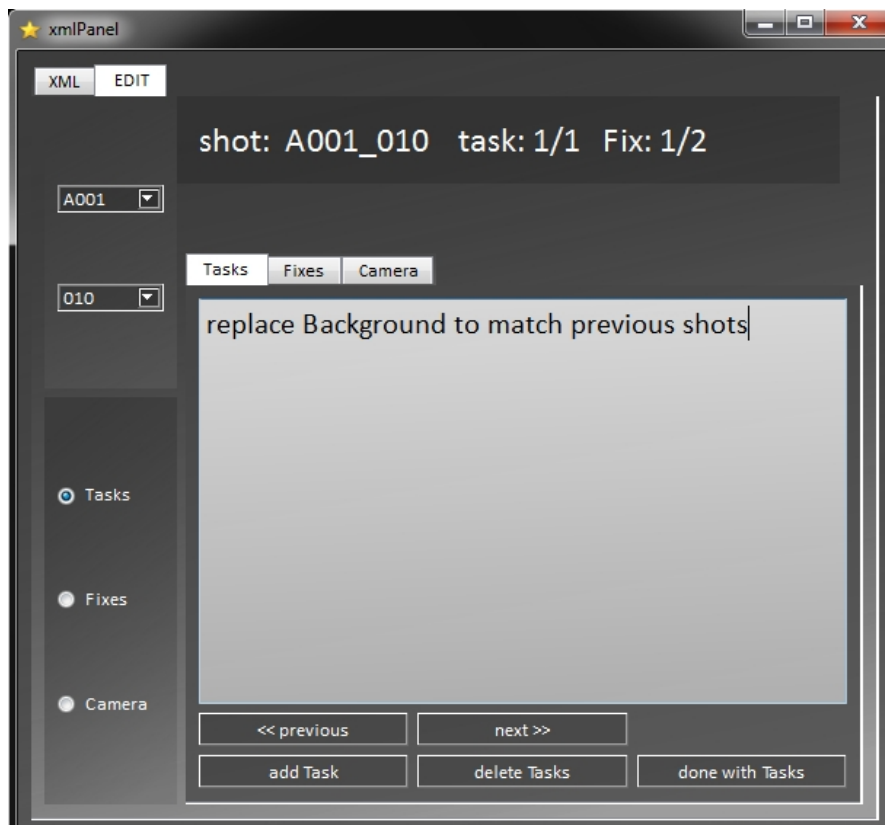


Abb. 06: XML_Tool – Edit Tab

Innerhalb des Programms wird dies als „Fixes“ bezeichnet. Als Beispiel kann man hier Maskenfehler, Farbkorrekturen oder das Positionieren von Elementen nennen. Es sind vorrangig Verbesserungen auf technischer Basis, aber auch ästhetische Aspekte sind hier zu berücksichtigen. Dazu würden zum Beispiel Bildkompositionen, Farbzusammenstellungen oder ggf. Geschwindigkeiten von Animationen zählen.

Der dritte und letzte Aspekt sind Informationen über die Kamera des jeweiligen Takes. Diese Informationen sollten mit allen wichtigen Parametern geliefert werden. Im Idealfall können diese Daten direkt vom Set übernommen werden und in das XML-File eingetragen werden. Falls ein „Matchmove“ von dem Shot erstellt wurde, ist kann statt nur den Parametern auch ein Verweis auf die generiert und exportierte Kamera auf den Server eingetragen sein.

Bei einem neu erstellten XML File werden diese Standardeinträge mit jeweils dem Vermerk „-empty-“ bezeichnet, bevor sie im Laufe des Projektes mit den Informationen gefüllt werden.

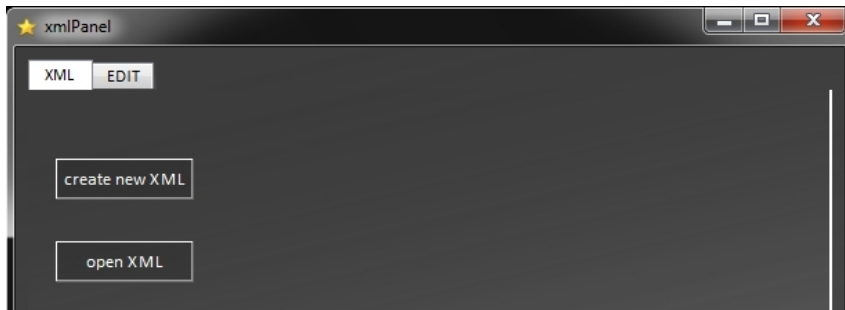


Abb. 07: XML_Tool – XML Tab

Das zu verwendende XML muss an einer zentralen Stelle auf dem Server liegen und sollte von jeder Person auslesbar sein. Die Berechtigung das Dokument zu verändern und zu speichern sollte wiederum nur an wenige Mitarbeiter verteilt werden um Fehleinträge weitestgehend auszuschließen. Es ist daher notwendig, die Lese- und Schreibrechte korrekt zu vergeben.

3.3.4.4.2. detailliertes Funktionsprinzip

Bei dem Entwickeln der Software wurde darauf geachtet, dass beim Erstellen der XML Datei keinerlei spezifische Namenskonvention eingehalten werden muss. Die korrekte Erstellung ist somit unabhängig der Namensgebung gewährleistet. Die Information wird direkt aus der Ordnerstruktur entnommen und als Grundlage verwendet. Die obere Ebene der Hierarchie der Ordner wird als Sequenzebene interpretiert. Durch diese wird iteriert und für jeden Ordner, ein Eintrag mit demselben Namen im XML erstellt. Innerhalb dieser Iteration wird eine zweite gestartet welche den Ordner der jeweiligen Sequenz durchsucht und zeitgleich für jeden Shot-Ordner ein Unterelement der Sequenz erstellt. Dadurch ist die Ordnerhierarchie ausgelesen. Um die Erstellung mit so geringem Aufwand wie möglich zu gestalten, werden in der gleichen Iteration weitere Standardeinträge erstellt. Diese sind die Kameradaten- und Kommentareinträge. Die Kamera enthält ebenfalls Unterpunkte, die die verschiedenen Parameter bereitstellen wie beispielsweise den Kamertyp (camType), die Brennweite (focalLength) und die Bilder pro Sekunde (Frames per Second, FPS). Die Kommentareinträge sind ebenfalls unterteilt. Hierbei wird zwischen den Aufgaben (Tasks)

und Änderungen (Fixes) unterschieden. Diese können eine beliebige Anzahl an Unterpunkten besitzen (siehe Abb. 08).

Soll ein vorhandenes XML bearbeitet werden, kann man anhand eines File-Dialogs eine schon kreierte XML-Datei öffnen. Anschließend wird das XML eingelesen und die jeweiligen Dropdown-Menüs mit den Sequenz- beziehungsweise den Shotnamen gefüllt. Basierend auf die Kombination welche in den Dropdown-Menüs ausgewählt ist, wird das XML iteriert. Findet es den gleichen Eintrag wie im Sequenz-Dropdown, iteriert es eine Ebene weiter bis es denselben Eintrag des Shot-Dropdown innerhalb des eingelesenen XML findet. Daraufhin werden die Kommentare in das Textfeld geschrieben. Abhängig davon, welches Tab aktiv ist, werden Tasks, Fixes oder die Kameradaten angezeigt. Bei Änderungswünschen können die Daten geändert, gelöscht oder neue Einträge hinzugefügt werden.

```
<scene name="A001">
  <shot name="010">
    <camera>
      <camType>
        RedOne
      </camType>
      <focalLength>
        50
      </focalLength>
      <fStop>
        4
      </fStop>
      <FPS>
        25
      </FPS>
      <shutter>
        180
      </shutter>
      <filmGate_height>
        24
      </filmGate_height>
      <filmGate_width>
        36
      </filmGate_width>
    </camera>
    <comments>
      <tasks>
        <task name="01">replace Background to match previous shots</task>
      </tasks>
      <fixes>
        <fix name="01">add more cars in background</fix>
        <fix name="02">add travelling smoke like previous shots</fix>
      </fixes>
    </comments>
  </shot>
  <shot name="020">
```

Abb. 08: anhand des XML_Tools kreierte und bearbeitete XML Struktur eines Shots

Das Auslesen geschieht im Compositing-Programm Nuke. Die Implementierung in andere Softwarepakete ist ebenfalls möglich, benötigt aber einer Umstrukturierung des Python-Codes auf das spezifische Compositing-Programm. So müsste der Code auf die jeweiligen Oberflächen und Möglichkeiten der zu verwendenden Software geändert werden. In der vorliegenden Arbeit wurde Nuke gewählt, da dies das aktuelle Arbeitsprogramm des Autors ist und somit das Skript im alltäglichen Gebrauch auf Fehler getestet werden kann und mögliche notwendige Optimierungen zeitnah vorgenommen werden können.

3.3.5. Nuke – Panel

Das vorrangig größte Problem ist die Implementierung von externen Daten die keine Video- oder Bilddateien sind. Durch das essentielle einbinden von Python in die Nuke Struktur stellt dies aber keine unmögliche Aufgabe dar. Das Auslesen wird alleine durch Python gesteuert während die Nuke Oberfläche vorerst nur dazu dient das XML File auszuwählen. Die daraus resultierenden Daten werden dann an Nuke übergeben. Dort bieten sich mehrere Möglichkeiten um die Informationen von den XML anzeigen zu lassen. Eine Lösung ist die Erstellung eines eigenen Panels (siehe Abb. 09). Dies ist gleichzusetzen mit einen separierten Fenster ähnlich wie im Betriebssystem. Darin werden die verschiedenen Aufgaben und „To-do-Listen“ angezeigt. Auch für die Kamera ist ein extra Tab vorhanden (siehe Abb. 10). Darin sind die Parameter aufgelistet und falls notwendig ein Verweis auf die exportierte Kamera aus dem Matchmove Departments. Um die Visualisierung benutzerfreundlicher zu gestalten wird auf jegliche zusätzlichen Informationen und gestalterische Elemente verzichtet. Ziel des Panels ist es, die Aufgaben oder die Änderungen des Supervisors innerhalb von Nuke zu betrachten ohne ein externes File oder andere Programme öffnen zu müssen. Zudem ist es möglich, einzelne Aufgaben als erledigt abzuhaken um auch ein visuelles Feedback zu bekommen wie viel noch zu ändern ist, oder schon vorgenommen worden ist.

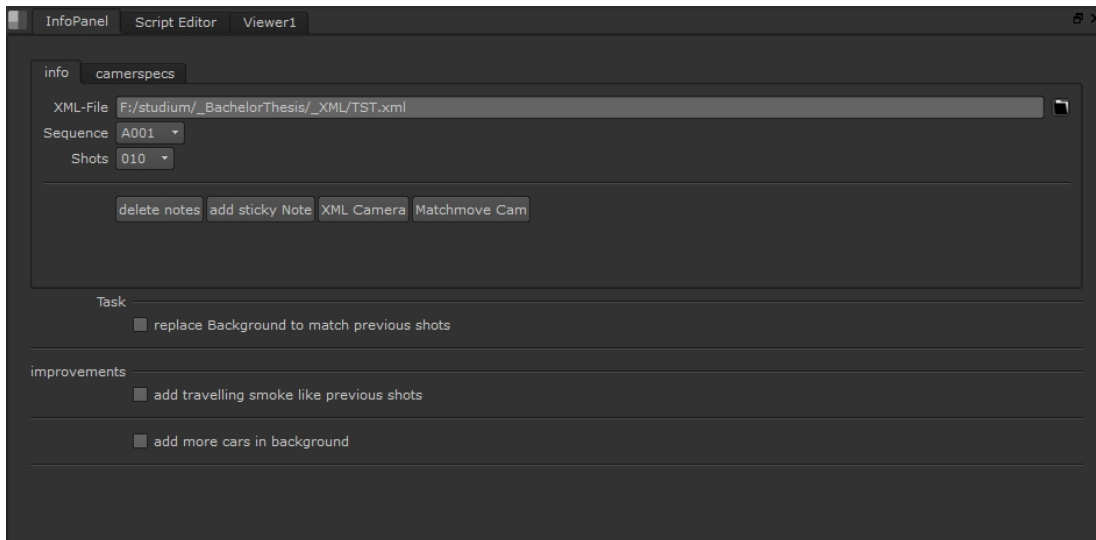


Abb. 09: InfoPanel innerhalb von Nuke mit ausgelesen Daten des XML's

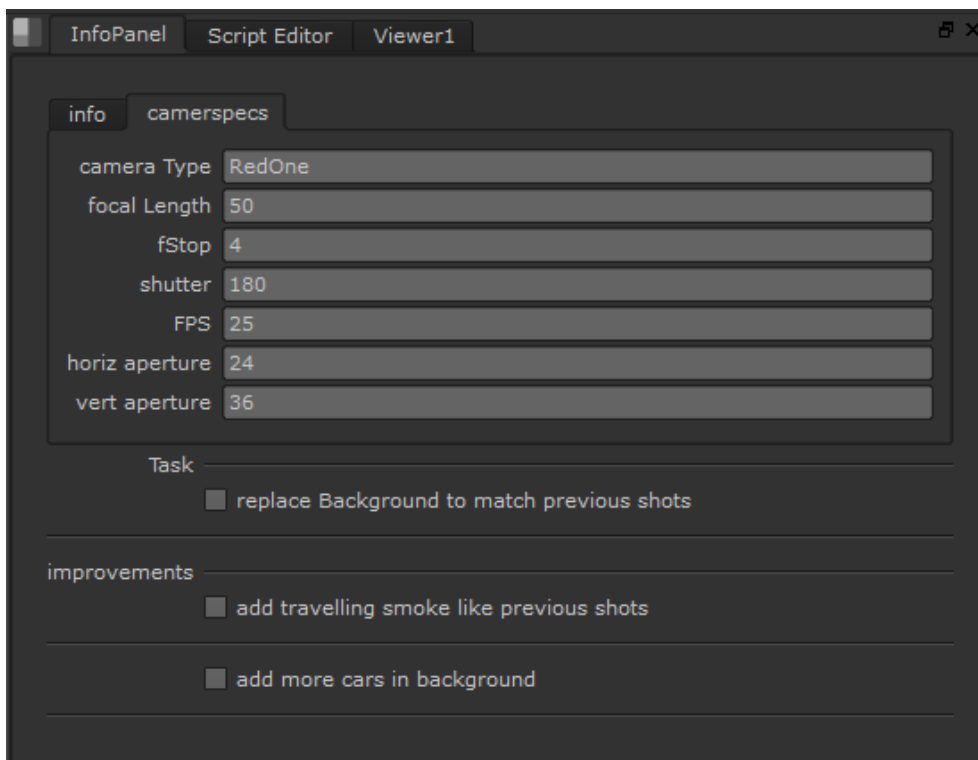


Abb. 10: InfoPanel innerhalb von Nuke - Kameraparamter

Neben der Implementierung als separiertes Fenster gibt es im Panel die Möglichkeit, sogenannten „Sticky Notes“ (siehe Abb. 11) anzulegen. Diese können als eine Art Notizzettel innerhalb der wichtigsten Teiloberfläche in Nuke (Nodegraph, siehe Abb. 12) verstanden werden. Diese Notizen sind standardmäßig nur als Text formatiert, sie können allerdings innerhalb von Nuke und Änderungen in Python leicht geändert und ergänzt werden.

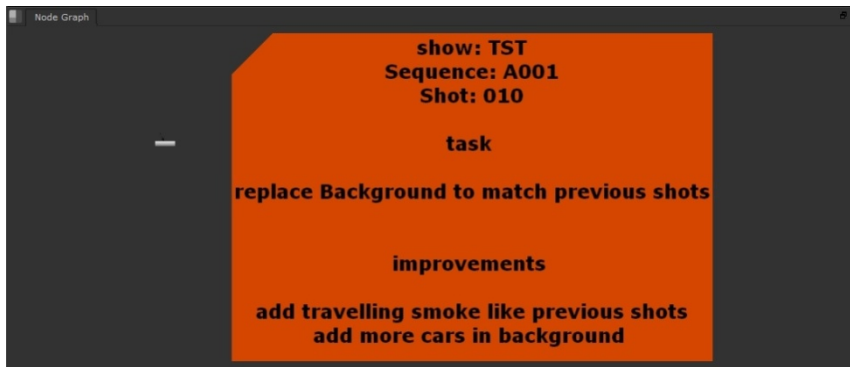


Abb. 11: Sticky Note innerhalb des „Nodegraphs“

Die „Sticky Notes“ Notizen können aus dem Infopanel heraus erstellt werden. Im Tool der vorliegenden Arbeit wurde zusätzlich eine Variante eingebunden die ohne das eigentliche Infopanel funktioniert. Der Grund hierfür liegt in der praktischen Anwendung und der unterschiedlichen Ansprüchen, die von Seiten der Artists an ein solches Tool gestellt werden. Eine nicht-repräsentative Befragung von 20 Artists, die im Vorfeld der vorliegenden Untersuchung durchgeführt wurde, zeigt uneinheitliche Anforderungen. So wollten die Hälfte der Befragten ein eigenständiges Fenster während die anderen zehn hingegen bevorzugten die Informationen direkt neben den Nodes zu sehen. Um den Ansprüchen aller Artists gerecht zu werden, wurden im vorliegenden Tool beide Varianten entwickelt. Die Erweiterung des Interfaces selbst, ist innerhalb von Nuke unkompliziert, da bei der Entwicklung dieser Software darauf geachtet wurde, dass sie möglichst flexibel bleibt. Damit wird gewährleistet, dass sich jedes Studio sich ein individuell erweitertes Nuke programmieren kann, um es auf firmeninterne Besonderheiten anzupassen. Das gelieferte Standardprogramm stellt in vielen Unternehmen nur die Basis dar.

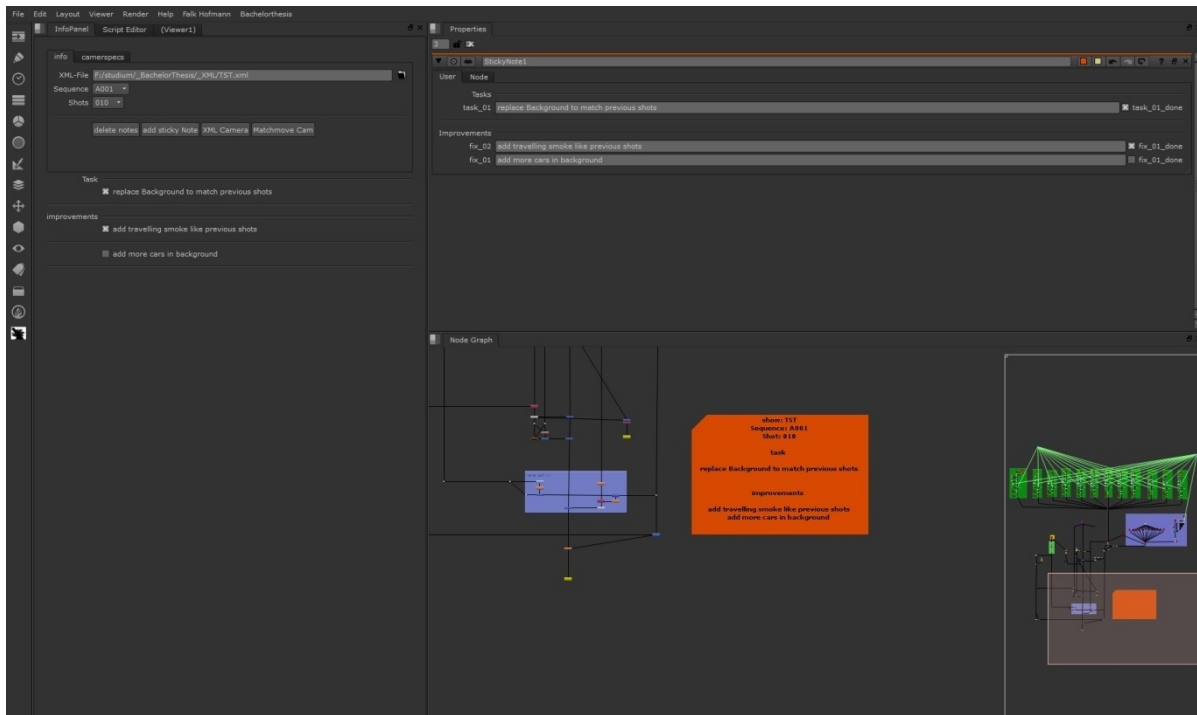


Abb. 12: InfoPanel und Sticky Note innerhalb eines Nuke-Files einer Produktion

3.3.6. Datenbanken versus Dateibasierte Speicherung

Die Entscheidung welche Art der Datenverwaltung man wählt, hängt von vielen Faktoren ab. Der relevanteste Aspekt ist wohl die Studiogröße und die bereitstehenden finanziellen Mittel. Im Folgenden soll ein Überblick über die beiden Arten der Speicherung gegeben werden.

3.3.6.1. Online-Datenbanken

Es gibt derzeit mehrere große Softwarepakete, die sich auf die Verwaltung und Organisation von Filmprojekten spezialisiert haben. Diese bieten die Möglichkeit einen Server zu mieten, auf dem die Datenbank und Software installiert sind. Diese Softwares können auf die Bedürfnisse des Studios abgestimmt und modifiziert werden. Somit wird eine leichte Anbindung in die bereits bestehende Pipeline gewährleistet. Der Vorteil von Online-Datenbanken ist, dass von verschiedenen Orten darauf zugegriffen werden kann, dass Änderungen eingetragen und schon bestehende Einträge aktualisiert werden können. Da dies ohne zeitlichen Verlust umgesetzt werden kann, ist die Information bei jedem Zugriff automatisch auf den aktuellsten Stand. Bei Informationsübergaben spielt dieser Aspekt eine wichtige Rolle, da veraltete Notizen etc. leicht zu Verwirrungen und folglich zu Fehlern führen können. Verwendet man Online-Datenbanken kann dieses Problem vermieden

werden. Ein weiterer Vorteil beruht im zeitgleichen multiplen Bearbeiten. In Online-Datenbanken können im selben Augenblick mehrere Einträge von verschiedenen Orten und Computern aus geändert werden, ohne dass eine Warteschlange in der Speicherung entsteht. Dies gewährleistet ein hohes Arbeitspensum, da kontinuierliches Bearbeiten zu jeder Zeit gegeben ist.

Ein Nachteil von Online-Datenbanken in dieser Größenordnung ist die Komplexität und die damit verbundene Vernetzung. Zwar sind die Systeme weitestgehend ausgereift, dennoch kommt es von Zeit zu Zeit zu Fehlern und Komplikationen. Wenn diese in einer wichtigen Phase von Projekten auftreten, so kann es im Projektablauf zu so schwerwiegenden Zeitproblemen kommen, dass die Fertigstellung beeinträchtigt wäre. Würden Online-Datenbanken als zentrales Element weltweite Verknüpfung verwendet werden, wären zudem Zugriffsrate und die daraus resultierenden Änderungen relativ hoch. Daher ist es schwierig, einen günstigen Zeitpunkt für Back-Up zu finden. Das Back-Up selbst stellt insofern ein Problem dar, wenn die Datenbank nicht auf einem eigenen Server installiert ist. Ein weiterer Nachteil von onlinebasierten Datenmanagement, das sich bis zur Arbeit des Artists durchzieht, ist die Abhängigkeit vom Internet und dessen Geschwindigkeit während der Bearbeitung. Sobald Probleme mit dem Provider oder der direkten Internetanbindung auftreten, kann die Arbeit nicht mehr fortgeführt werden. Zudem besteht die Notwendigkeit, auf die Beständigkeit des Datenbankbetreibers zu vertrauen. Kommt es bei dessen Servern zu Einschränkungen oder Ausfällen, wird die Arbeit der Studios in Mitleidenschaft gezogen. Auch wenn die Betreiber gut gegen einen Ausfälle gesichert sind, bleibt ein gewisses Restrisiko dennoch bestehen.

3.3.6.2. Filebasierte Datenverwaltung

Im Gegensatz zu großen Online-Datenbanken kann die Verwaltung der Informationen auch auf lokale Daten reduziert werden. Als Beispielformat ist hier XML zu nennen. Dieses Format bringt, genauso wie Online-Datenbanken spezifischen Vor und Nachteile mit sich. Diese werden im Folgenden betrachtet.

Der größte Nachteil ist, dass kein multiples zeitgleiches Bearbeiten möglich ist. Daher ist das Arbeiten auf eine Person begrenzt, was die Flexibilität stark einschränkt. Dennoch bietet diese Möglichkeit einer Art Administration zur Erstellung und Bearbeitung der zu verwaltenden Datei. Der größte Vorteil von dateibasierter Datenverwaltung beruht in der Unabhängigkeit vom Internet, wenn die Verwaltungsdaten auf den jeweiligen lokalen Servern gespeichert sind und nur bei Bedarf auf Online-Servern synchronisiert werden müssen. Das eigentliche Arbeiten bleibt davon jedoch unberührt. Auch das Sicherheitskopieren (engl. Back-Up) stellt kein Problem dar, da die Dateien kaum Speicherplatz benötigen. Somit werden ein tägliches separiertes Back-Up und ein Zugriff auf jeden zeitlichen ermöglicht.

3.3.7. Metadaten im OpenEXR Format

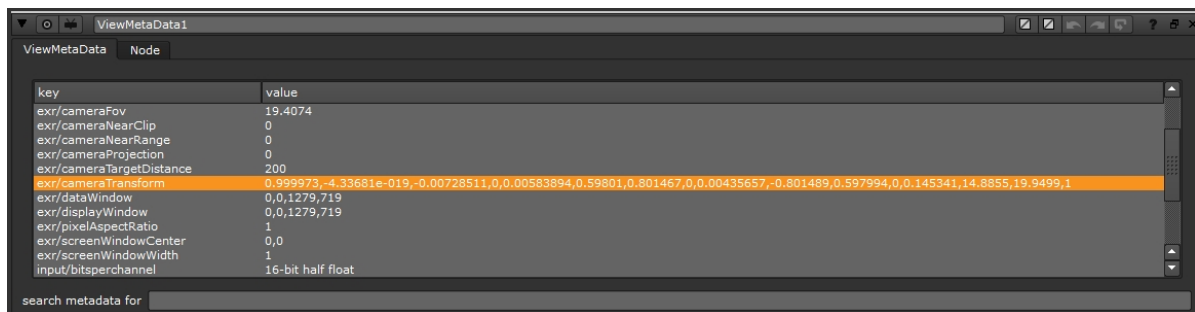
OpenEXR ist ein High-Dynamic-Range Bildformat, welches von „Industrial Light and Magic“ entwickelt und unter einer Open-Source-Lizenz veröffentlicht wurde. Es unterstützt 16- und 32-Bit Gleitkommazahlen je Farbkanal und ist somit nicht nur für Bildinhalte im eigentlichen Sinn nutzbar. Es können Tiefeninformationen und andere Bildinformationen gespeichert werden deren Werte im negativen Bereich oder über 1,0 liegen. Zusätzlich bietet dieses Format eine unbegrenzte Anzahl Metadaten die eigenständig hinzugefügt werden können.

3.3.7.1. Kameradaten

Ein Beispiel, wie man Metadaten in der eigentlichen Visual Effects Pipeline nutzen kann, stellen die gerenderten Files aus dem 3D-Department dar. So ist es möglich mit dem Renderpaket V-Ray die virtuelle Kameras anhand derer die Bilddaten berechnet wurden, innerhalb der Metadaten in Form von Matrizen mit zu exportieren. Die Frage, ob ein solches Vorgehen lohnenswert ist, soll im Folgenden geklärt werden.

Da es häufig zur Kombination zwischen real gedrehten Filmaufnahmen und computergenerierten Bildern kommt, müssen die Kameras aus beiden Bereichen perfekt

aufeinander abgestimmt sein. Dieser Abgleich wird auch „Matchmove“ genannt. Es ist nicht immer der Fall, dass das Compositing-Department den Matchmove benötigt. Wird der Abgleich in einem späteren Arbeitsschritt dennoch benötigt, kann sich mit Hilfe der Matrizeninformation die Kamera basierend auf den 3D-genierten Daten erzeugen lassen. Diese Informationen sind bereits vorhanden und müssen nur entsprechend verwendet werden. Dies gilt nicht nur für stehende Kameras. Ist die Kamera animiert, wie in den meisten Fällen, sind die Matrizen in jedem Bild (engl. Frame) mit anderen Werten versehen (siehe Abb. 13/14), woraus wiederum die Kamerabewegung resultiert und abgelesen werden kann. Es ist daher für die spätere Bildbearbeitung völlig unwichtig wie lange, schnell oder stark sich die Kamera bewegt, da alle benötigten Informationen in den Matrizen sowie den weiteren Daten wie beispielsweise Bildwinkel (Field of View) enthalten sind. Diese Informationen können aus dem Rendering entnommen werden.



bb. 13: Kameramatrix in den ausgelesenen Metadaten innerhalb von Nuke an Frame 001

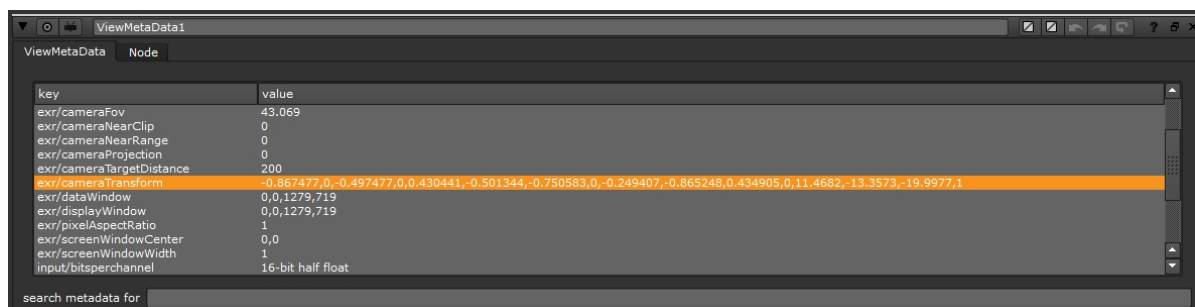


Abb. 14: Kameramatrix in den ausgelesenen Metadaten innerhalb von Nuke an Frame 040

3.3.7.2. Compositing-Skripte

Neben den Kameradaten kann man auch weitere Informationen die man durch den Produktionsprozess hinweg beibehalten möchte in die Metadaten schreiben. So ist es unter Anderem auch möglich, die gesamten Skripte der Compositing-Artist während der Produktion in die dazugehörigen exportierten Daten zu schreiben. Somit hat man in jedem Rendering die Information, welches Bild und welche Veränderung aus dem Compositing kommt. Zudem verfügt man über das dazu gehörige Skript und es kann bei Unklarheiten direkt nachgeschaut werden, ohne die Projekt-Files zu durchsuchen. Darüber hinaus bietet sich diese Methode zum Sicherheitskopieren an, da Arbeitsfiles und Resultat mit-einander kombiniert sind.

Um diese Vorgehensweise zu realisieren und anschließend zu prüfen wurden in der vorliegenden Arbeit einige Testskripte für Nuke geschrieben. Eine weitere Möglichkeit besteht in der Verwendung einzelner Teile eines Skriptes. Als Beispiel kann hier das Pre-Compositing anführen. Pre-Compositing beschreibt das Zuarbeiten beziehungsweise Vorbereiten eines Bildes mit Aufgaben wie Retuschieren (Painten), Maskieren (Rotoskopieren) und Bewegungsdaten generieren (Tracken). Dieses Vorgehen würde die Möglichkeit bieten, zunächst mit den vorgerenderten Dateien zu arbeiten, am Ende der Produktion unkompliziert das Pre-Compositing-Skript zu implementieren und die zusammengeführten Projektdaten als Masterskript zu rendern.

Befürchtungen, dass gerenderte Daten dadurch deutlich mehr Volumen an Speicherplatz benötigen, sind unbegründet, da die Projektdaten von Nuke nur Text beinhalten und dies wiederum kaum Speicherplatz bedarf. Somit stellt die das Schreiben der Projektdaten in die Metadaten betreffend den Speicherbedarf kein Problem dar. Die einzige Schwierigkeit liegt in der vorgeschriebenen Maximallänge eines Eintrages. Um dieses Problem zu lösen, unterteilt man das Textfile in verschiedene nummerierte Abschnitte mit der jeweiligen Maximallänge. Beim Auslesen werden Diese dann in der richtigen Reihenfolge zusammengefügt. Der daraus resultierende Text ist dann mit dem Arbeitsfile identisch. Vor Allem bei größeren Projekten mit entsprechend großen Skripten bietet sich diese Möglichkeit an, um die Daten miteinander zu verknüpfen.

3.3.7.3. Hyperlinks

Eine weitere Möglichkeit, Metadaten in Pipelines zu verwenden, ist das Einbinden von Hyperlinks, dem Verweisen auf bestimmte Daten auf lokalen oder webbasierten Servern. Da Metadaten als Paar (Dictionary) eingebunden sind, ist es möglich verschiedene Arten der Verweise als Zusatzinformation in die Bilddaten zu schreiben. Ein Dictionary besteht aus zwei Teilen: einem Schlüsselwort (engl. Key) und dem dazugehörigen Wert (engl. Value). Somit ist das Erstellen von Verweisen auf den Server oder der Online-Datenbank unkompliziert. Den Namen des Zielorts kann als Key und der Pfad als Value zugeordnet werden. So besteht beispielsweise die Möglichkeit, dem finalen EXR-Dateien beim Rendern jede Bilddatei und Sequenz die zur Erstellung benötigt wurden ist, als Verweis in die Metadaten zu schreiben. Nachdem das Projekt abgeschlossen ist und man es in das Archiv verschieben möchte, kann ein Skript ablaufen, welches auf den Informationen in den Metadaten basiert. Dieses Skript liest jeden Pfad aus und kopiert ihn mit dem neu generierten Pfad in einen Archivordner. Auf diese Weise liegen alle Daten, die zur Wiederherstellung benötigt werden, in einem Gesamtarchiv. Der Vorteil gegenüber anderem Lokalisierungstool ist, dass es nicht nötig wäre jeden einzelnen Shot zu öffnen um anschließend innerhalb des Compositings den Shot zu archivieren. Zudem wäre eine Stapelverarbeitung ohne weiteres möglich, indem mehrere oder alle Shots eines Projektes nacheinander in einen Batch File (*.bat) abgearbeitet werden könnten.

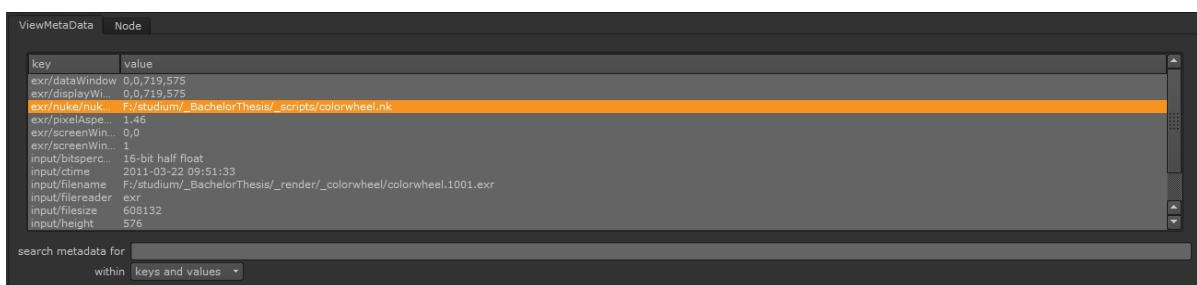


Abb. 14: Metadaten einer EXR –Sequenz mit Verweis (Pfad) auf das Nukeskriptes

4. Abschlussbetrachtungen

Das in der vorliegenden Arbeit vorgestellte XML-Tool wurde in der Praxis getestet. Es diente als Kommunikationstool zwischen Supervisor und Artist und wurde vom Supervisor als hilfreiches und geeignetes Tool bewertet:

„XML_Tool with InfoPanel by Falk Hofmann is greatly improving information flow from supervisors to artists in a high end vfx-pipeline. The tool is a shortcut for communicating directly with the artist who is working on the specific shot and needs rapid information. Artists don't need to constantly jump anymore between email or heavy databases to check what revisions are needed in shots, but all necessary information is instantly in their hands in same Nuke environment where the actual work is done. For supervisors it's easy and fast to use, and saving time from checking that information is going to the artist correctly. This tool works perfectly on reviewing dailies, as notes can be written on the spot and same moment it reaches the artist. Simple information flow is a must in well working pipeline and simplicity of XML_Tool with infoPanel provides fast, simple and efficient connection to artists.“

(Saku Partamies, VFX Supervisor, Pixomondo Shanghai)

Aufgrund dieser externen Einschätzung kann die entwickelte Software XML-Tool als effizienzsteigernd und zukunftsfähig eingeschätzt werden.

4.1. Fazit

Fasst man die problemlösungsorientierten Ansätze zusammen und betrachtet man die Erfahrungen aus der zweimonatigen praktischen Umsetzung des entwickelten Tools, so kann festgehalten werden, dass das entwickelte Metadatentool eine Verbesserung in der Informationsübergabe an den entsprechenden Artist ermöglicht hat. Darüber hinaus konnte der Verlust von Notizen oder Ungenauigkeiten in der Informationsweitergabe minimiert werden. Der größte Vorteil besteht darin, dass sich wohl der Artist und als auch der Supervisor bei den Besprechungen auf das Bild konzentrieren können ohne zeitgleich die vorgenommenen Änderungen und zusätzliche Informationen schriftlich festhalten zu müssen. Positiv ist ebenfalls, dass anschließend keine weitere Zeit zur Übergabe der Informationen an den Artist benötigt wird. Dies geschieht automatisch mit dem Speichern. Auf diese Weise können besprochene Kritikpunkte und Änderungswünsche sofort in Nuke sichtbar gemacht werden.

Trotz der Effizienzsteigerung und der Vorteile des neu entwickelten Tools, gibt es noch immer Verbesserungspotential bei der Einbindung von Metadaten in die Postproduktion. Um eine perfekte Anbindung in eine bestehende Pipeline zu etablieren, werden optimierende Abläufe benötigt, die wiederum die Basis für eine Einbindung der Informationsübergabe in den Produktionsprozess bilden. Dies könnte beispielsweise mit einer direkten Anbindung an eine Datenbank umgesetzt werden. Hierbei wäre es von Vorteil, wenn nicht der Artist direkt mit der Datenbank kommuniziert, sondern es im Hintergrund geschieht. So ist es beispielsweise möglich, bei jeder Änderung in der Datenbank, die den jeweiligen Shot betreffen, ein XML zu aktualisieren und an einem bestimmten Platz lokal auf dem Produktionsserver zu speichern. Somit wäre auch eine Kommunikation möglich, ohne dass sich der Artist in die passwortgeschützten Datenbanken einloggen und nach den passenden Informationen suchen muss.

4.2. Zukünftige Erweiterungen

Da das Programm in dem aktuellen Status zwar einsatzbereit ist, gibt es verschiedenen Punkte die in der Zukunft noch Erweiterungen und Verbesserungen.

Das im Moment am wichtigsten erscheinende ist eine Kommunikation in beide Richtungen und nicht nur vom Supervisor zum Artist. So sollte der Compositor durch ändern des Statuses innerhalb von Nuke nicht nur für sich selbst sichtbar machen, vielmehr müsste eine Zusatzinformation oder Kennzeichen in das XML geschrieben werden ohne dabei die eigentlichen Kritikpunkte zu ändern. Anhand dessen kann der Supervisor sich einen Überblick machen. Dies setzt allerdings eine andere Implementation des Zugriffes auf das XML voraus, wodurch ein zeitgleiches Bearbeiten zulässig wird.

Weiterhin ist eine Vernetzung mit bestehenden Daten durchaus nützlich. Zusätzliche Standardeinträge wie beispielsweise Referenzen würden den Arbeitsablauf ebenfalls beschleunigen. Hierbei könnte mit direkten Verweisen gearbeitet werden, die vom Supervisor oder Coordinator kreiert werden und innerhalb von Nuke den Artist erreichen. Auch eine Benachrichtigung per E-Mail wäre hilfreich um den Artist in Kenntnis darüber zu setzen das neue Kommentare eingetragen werden. Der Folgeschritt hieraus wäre eine automatische Aktualisierung der Kommentare innerhalb von Nuke.

4.3. Weiterführende Gedanken

Im Gegensatz zu den großen hochprofessionellen Firmen gibt es auch kleine Produktionsbüros die sich eine spezialisierte Datenbank nicht leisten können. Um dennoch eine zentrales Verwaltungsmöglichkeit zu haben, bietet sich eine Software wie in der vorliegenden Arbeit entwickelte (*XML_Tool*) an. Dieses kann bei Bedarf in der eigentlichen Produktion verbessert und erweitert werden. Um als ‚das‘ zentrale Element zu fungieren, benötigt es noch weitere zentrale Verbesserungen, wie beispielsweise das regelmäßige Back Up, das Rendern von Proxys, das Aktualisieren von Schnitten und das Archivieren der fertiggestellten Projekte.

Es sollte daher – und ist als realistisch einzuschätzen –, dass ein solches Programm entsprechend der Unternehmensbedürfnisse geschrieben wird. Nachteilig ist hierbei, dass die erstellte Software einige Zeit benötigt, um entwickelt zu werden und um sie fehlerfrei verwenden zu können. Zudem werden qualifizierte Fachkräfte benötigt um die Umsetzung voran zu bringen. Der Vorteil von eigener Software hingegen liegt in der individuellen Anpassung an vorhandene unternehmensinterne Strukturen von Hard- und Software. Dieser Vorteil ist bei der Entscheidung für eine eigene Software zentraler, da sie dem Unternehmen größere Effektivitätssteigerungen ermöglicht.

Eidesstattliche Erklärung

Mir ist bekannt, dass alle Hilfsmittel mit korrekter Quellenangabe versehen sein müssen.

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig verfasst und jenseits der angegebenen Quellen keine weiteren Hilfsmittel benutzt habe, insbesondere keine anderen als die angegebenen Informationen aus dem Internet.

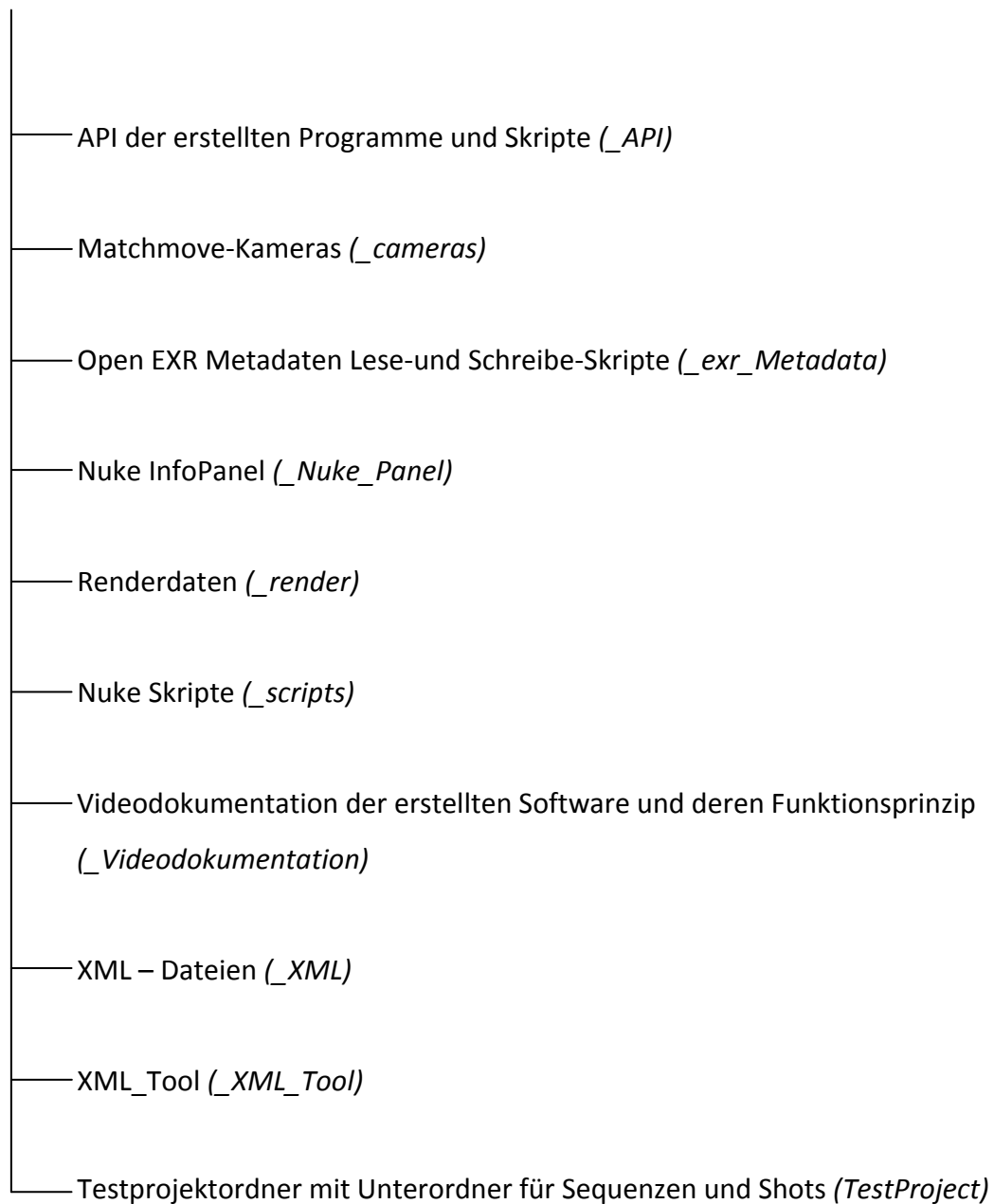
Ich habe die, etwaige Betrugsversuche betreffenden, Paragraphen der für mich gültigen Prüfungsordnung an der Hochschule der Medien Stuttgart (HdM) zur Kenntnis genommen.

Stuttgart, den 28.03.2011

(Falk Hofmann)

Anhang

Dieser Arbeit beiliegende DVD beinhaltet Dateien und Ordner die als Beleg dieser Arbeit dienen. Diese sind im Folgenden aufgelistet.



Quellen- und Literaturverzeichnis:

- Mark Pilgrim, Dive Into Python, ISBN 978-1590593561
- Mark Lutz, Learning Python: Powerful Object Oriented Programming
ISBN 978-0596158064
- Peter Walerowski , Python Grundlagen und Praxis, ISBN 978-3-8273-25
- <http://www2.sub.uni-goettingen.de/intrometa.html>
- <http://www.photometadata.org/node/57>
- <http://blog.merlinone.com/MerlinOne-Digital-Asset-Management-Blog/bid/6982/A-Brief-History-of-Metadata-for-Digital-Assets>
- http://www.getty.edu/research/publications/electronic_publications/intrometadata/pdf.html
- <http://www.controlledvocabulary.com/index.html>
- <http://www.pureblendsoftware.com/movieslate>
- <http://en.wikipedia.org/wiki/Metadata>
- <http://www.organizepictures.com/>
- <http://www.commandprompt.com/community/pyqt/?page=pyqtbook>
- <http://zetcode.com/tutorials/pyqt4/>
- <http://www.python.org/>
- http://books.google.com/books?id=nEJ-jcYF2fMC&dq=learning+python+lutz&pg=PP1&ots=3_tBP3b4hg&sig=bPd_X9MJwVw0mETxILMlf4cmg-8&hl=en&sa=X&oi=book_result&resnum=1&ct=result#v=onepage&q&f=true