

2017

ISSN 1433-2620 > B 43362 >> 21. Jahrgang >>> www.digitalproduction.com

Deutschland € 17,70

Published by ATEC

Österreich € 19,-

Schweiz sfr 23,-

1

DIGITAL PRODUCTION

DIGITAL PRODUCTION

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

JANUAR | FEBRUAR 01:2017



CG-Characters!

Im Fokus: Helden-Design und Statisten-Creation

Maya Bifrost

Neue Wassersimulation – bis zum letzten Schluck

Neue Tools

ZBrushCore, Trapcode, Blender 2.78 & mehr



4 194336 217709

01 >

Alltag eines Gartengnomms – Character Animation in Blender

Die Idee zum kleinen Steinfresser entstand mit der Absicht, den Arbeitsablauf in Blender durchzuspielen sowie die Grundlage für eine Kurzfilmserie zu schaffen. Die Hauptfigur ist ein leicht durchgeknalltes, verfressenes Männchen, das unscheinbar mitten unter uns lebt und aufgrund seiner Neugierde auf alles Essbare von Abenteuer zu Abenteuer stolpert, wobei es als CG-Character auf real gefilmtem Hintergrund agiert.

von Titus Fehr, Momotion GmbH

Nach Abschluss der Konzeptphase begann der Gnom in Blender 2.77 seine Existenz als Cube – dieser wurde im Sculpt Mode zu einer Büste geknetet. Dabei erleichterten im Tool Shelf die Einstellungen „Dyntopo“ und „Symmetry“ den Prozess erheblich: Sie erlauben beim Sculpten eine automatische Geometrieverfeinerung gemäß Voreinstellung und Spiegelung des Modellierens an der gewünschten Achse.

So entstand eine immer detailliertere Büste, deren Proportionen zwischendurch mit dem Proportional Editing Tool korrigiert wurden. Daraufhin folgte die Übertragung der polygonlastigen Sculpt-Geometrie in ein animierbares Mesh (Retopology). Als Ausgangspunkt diente eine Plane, deren Vertices sich mit im 3D-Header aktiviertem Snap During Transform/Snap To Faces auf die zugrunde liegende Sculpt-Geometrie setzen ließen. Um auch

hier nur eine Hälfte des Characters erstellen zu müssen, erhielt diese Geometrie einen Mirror Modifier. Mittels fortwährendem Extrudieren und Snapping von Vertices/Edges entstanden so Loops um Auge und Mundöffnung, die sich dann mit Flächen füllen ließen.

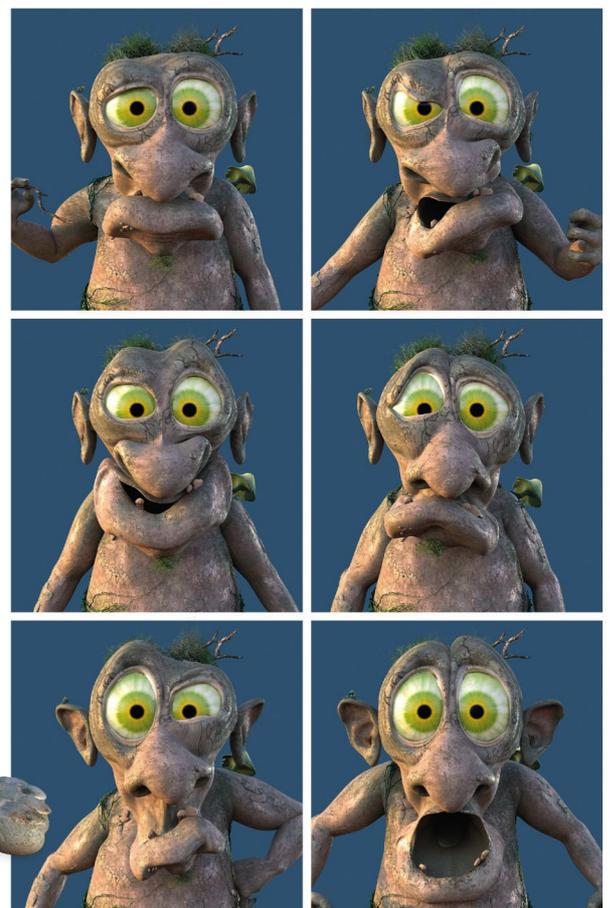
Nach Abschluss des Retopology-Prozesses vervollständigten freies Extrudieren und Modellieren die neue Kopf-Schulter-Geometrie mit Unterleib, fehlenden Gliedmaßen, Mundhöhle, Ohren und Augenschalen. Danach wurde der Mirror Modifier appliziert, und das jetzt überflüssige Sculpting-Modell konnte entsorgt werden.



Unter der Gesichtsgeometrie liegt ein Rig aus Bones, was beim Animieren grössere Freiheit bedeutet als eine mit Shape Keys vorgefertigte Mimikbibliothek.

Texturen

Nun brauchte die fertige Geometrie eine Steinmännchen-Haut mit verschiedenen Texturen für Farbgebung (Diffuse), Unebenheiten (Bumps/Normals) und Glanz- und



Matt-Bereiche (Glossy), die mittels UV-Koordinaten auf die Geometrie projiziert werden sollten. Das hierfür nötige Unwrapping des Characters ließ sich über das Definieren von Schnittlinien (Mark Seam) steuern, welche für eine möglichst verzerrungsfreie Abwicklung an Arm- und Beinansätzen, Hinterkopf, Hals, Rücken und Ohren eingefügt wurden. Die Gesichtsregion der abgewickelten Geometrie erhielt auf Kosten von Hinterkopf und Ohren im UV-Editor mithilfe des Proportional Editing Tools eine Hochskalierung, damit der prominentere Gesichtsbereich des Characters mehr Details durch höhere Auflösung zuließ. Die mit 4096x4096 Pixeln festgelegte Texturgröße stellte diese Detaillierung für Nahaufstellungen im HD-Format sicher.

Für das direkte Bemalen der Geometrie mit einer Textur hält Blender den Texture Paint Mode bereit. Darin lässt sich ein Bild verwenden, das dann wie eine Stempelschablone bei stetigem Rotieren der Geometrie auf diese geklont wird. Nach dem Heraussichern der so generierten Texturen geschah deren

weitere Verfeinerung und Farbanpassung in Gimp. Schließlich wurden sie im Node Editor zu einem Cycles-Material zusammengestellt und der Geometrie zugewiesen.

Haare, Pilze, Flechten und mehr

Ergänzend zur Körpergeometrie erhielt der Steinfresser weitere Objekte wie Pilze, partielle Körperschuppen, Augen, Zweig, Gräsgrütel, Haare und Flechten.

Letztere entstanden mit dem Add-on Ivy Generator (Ivy Gen). Generell wuchern die Flechten dieses Add-ons vertikal nach oben. Damit sie aber am Körper des Gnom teils auch schräg verliefen, diente für ihr Wachstum eine bei mehreren Wachstums-Durchgängen verschieden rotierte Gnom-Kopie als Grundlage. Die somit erhaltenen Ivy Curves wurden zusammengefügt, mit Bevel verdickt und in ein Mesh umgewandelt, das ein Unwrapping und mittels Texture Painting eine Bemalung erhielt. Die Gnom-Kopie war nun überflüssig und konnte gelöscht werden.

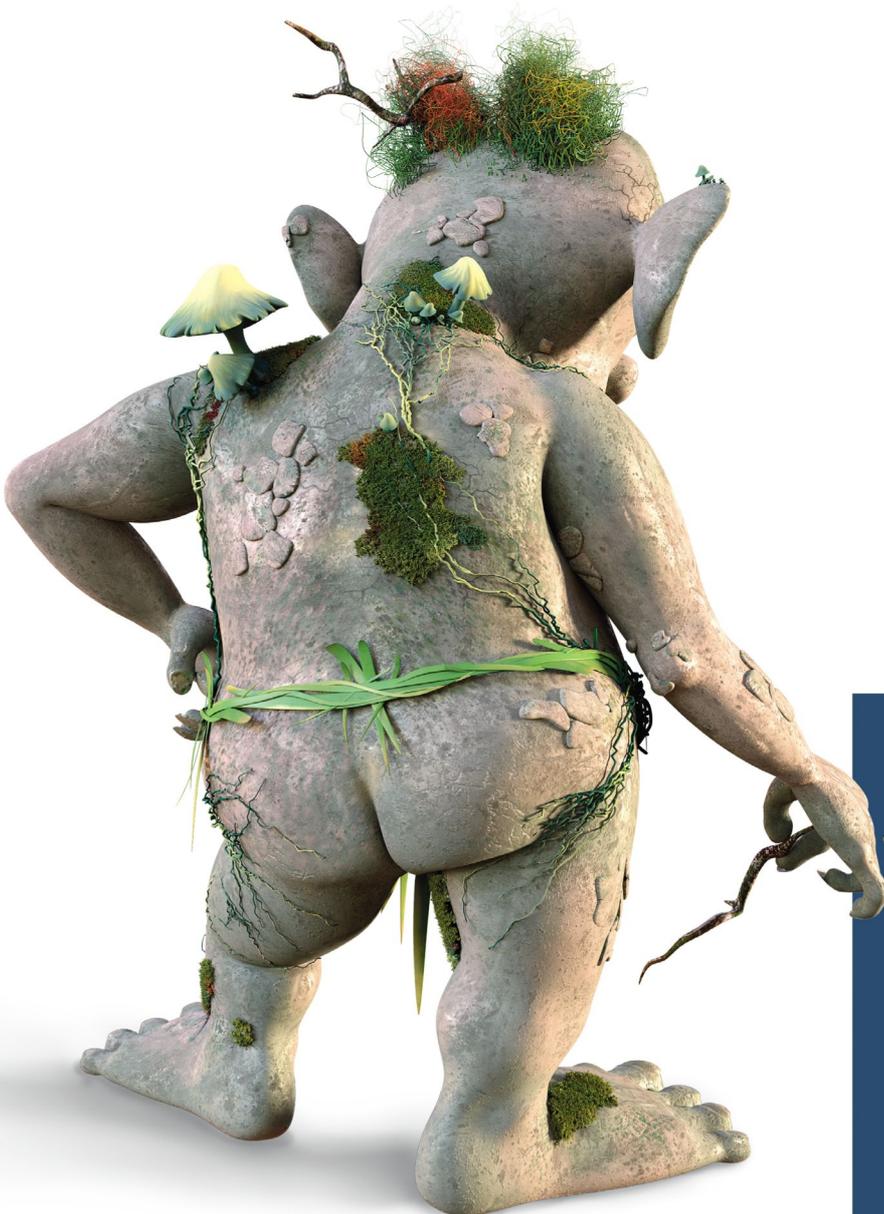
Der Gräsgrütel entstand als Curve-Geflecht, welches extrudiert, in ein Mesh umgewandelt und editiert wurde und dann ebenfalls das Unwrap- und Texture-Paint-Prozedere durchlief. Mehrere Particle Systems mit einem auf „Hair“ gestellten Emitter Type ließen Kopf-, Kinn-, Ohrenhaare und Moosflecken sprießen. Theoretisch könnten diese verschiedenen Haarbereiche auf derselben Geometrie wachsen, doch erleichterte hier eine separate Emitter-Geometrie für die einzelnen Bereiche das unterschiedliche Editieren und Frisieren.

Rigging

In Blender heißt das Rig standardmäßig „Armature“. Darin wurden im Edit Mode die den Character betreffenden Knochen (Bones) durch Extrude und Subdivide erstellt. Nach deren sinngemäßer Benennung können Arm- und Bein-Bones für die andere Körperhälfte gespiegelt, das heißt dupliziert und in der entsprechenden Achse mit -1 skaliert werden. Für die neuen Bones ermöglicht Blender bequemerweise eine automatische Umbenennung im 3D View Header über Armature/Flip Names.

Nachdem das Rig erstellt wurde, wurde davon in derselben Armature im Edit Mode eine Kopie gemacht. Der Grund dafür lag in der Aufteilung der jeweiligen Funktionen: Während die einen Knochen ausschließlich die Deformation der Geometrie übernehmen (nennen wir sie Def-Bones), diente das andere Gerüst als Anfasser für die Forward-Kinematics-Animation (FK-Bones). Nun wurden die Def-Bones – jeder einzeln und nachdem ihre hierarchischen Verbindungen (Parent-Child) aufgehoben wurden – mittels Copy Transform Modifiers an die einzelnen FK-Bones gebunden.

Das Kopieren ist nicht zwingend nötig, da die FK-Bones auch gleich die Mesh-Anbindung übernehmen könnten – doch hat in gewissen Situationen eine Separierung von FK- und Def-Bones Vorteile wie



zusätzliche Flexibilität durch Aufsetzen eines Spline IK Setups der Def-Bones, was bei Cartoon-Charakteren Rubber-Hose-Animationen und extreme Verbiegungen ermöglicht. Oder es kann zu einem späteren Zeitpunkt an der Mechanik (IK- und FK-Setup) gebastelt werden, ohne die Def-Bones zu tangieren und den Character neu weight-painten zu müssen.

In den Kopf-, Bein- und Armbereichen, wo auch zeitweise IK-Animation möglich sein soll, übernahm eine weitere Kopie der jeweiligen Knochen das Setup von Inverse Kinematics Constraints (IK-Bones). Die FK-Bones erhielten mittels Copy Transform Modifiers eine Bindung an diese IK-Bones.

In der 3D-Szene ermöglichten Kontrollknochen (Ctrl-Bones) das Umschalten zwischen FK und IK, die mittels Drivers die Influence Sliders dieser Copy Transform Modifiers steuerten. Eine Alternative zu Control Bones wären Custom Properties für die jeweiligen IK-Anfasser, deren Sliders dann im Properties Panel (3D View, N-key) über Drivers den Einfluss der Copy Transform Modifiers steuern und so das Umschalten zwischen FK und IK ermöglichen.

Um die Animation der Hand zu vereinfachen, steuerte ein Ctrl-Bone die Hierarchie über Drivers für jeden Finger, indem er bei Rotation nur das erste Fingerglied, bei Skalierung hingegen die übrigen Fingerglieder zugleich rotierte. Die Bones der Füße wurden so aufgesetzt, dass ihre Funktionsweise im IK-Modus separate Fersenmanipulation ermöglichte. So blieben Fußballen und



Zehen beim Anheben der Ferse am Boden haften, der ganze Fuß konnte aber über die Ferse rotiert werden. Die Zehen waren zusätzlich einzeln rotierbar.

Damit später die Ctrl-, FK- und IK-Bones beim Anbinden der Geometrie an die Armature (Parent With Automatic Weights) keine unnötigen Vertex Groups generieren und dadurch überflüssiges Weight Painting zulassen würden (da das ja die Aufgabe der Def-Bones ist), musste in deren Bone Settings die Deform-Option deaktiviert werden.

Nach dem Anbinden der Gnom-Geometrie an die Armature erfolgte die Verfeinerung der Vertex-Gewichtung mit den Weight Paint Tools. Die X-Mirror-Option vereinfachte diesen Prozess erheblich, indem die Vertex Weights beim Painten der einen Mesh-Seite automatisch gespiegelt wurden.

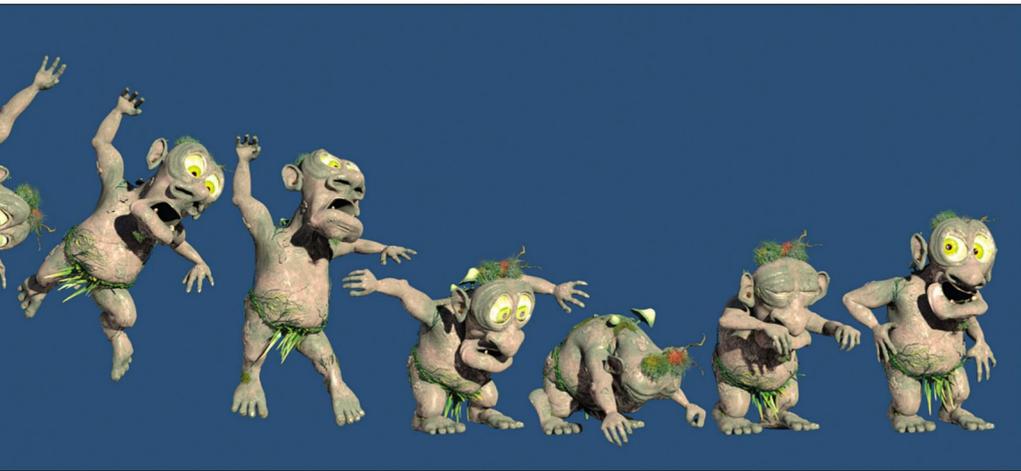
Um glaubhafte Gesichtsmimik zu erhalten, bietet sich einerseits das Modellieren ei-

ner Serie verschiedener Gesichtsausdrücke an (in Blender: Shape Keys), oder aber das Aufsetzen eines Gesichts-Rigs aus Bones in derselben Gnom-Armature.

Hier wurde die zweite Methode gewählt: Da der Gnom eine extrem ausgeprägte Gesichtsgeometrie besitzt, und in Blender bei Shape Keys die Vertices zwischen Punkt A und B auf direktem Weg, also linear interpolieren, ist der Übergang zwischen den Shape Keys hier problematisch und kann zu unerwünschten Verzerrungen führen. Das Erstellen eines Gesichts-Rigs mit dem dazugehörigen Weight Painting ist zwar aufwendig, doch zahlt es sich später bei der Animation durch weitgehende Flexibilität betreffend Mimik und Kontrolle über die Geometrie aus.

Lider, Brauen, Nase/Nasenflügel, Wangen, Mundwinkel, Kiefer etc. wurden durch Bones bewegt (Rotation), wobei daran wiederum Bones für weitere Verformungen





hängen können: Beispielsweise hing am Kiefer-Bone ein Knochenring, der über IK-Target-Bones die Lippengeometrie animierbar machte. Um den Animationsprozess zu beschleunigen, lässt sich in Blender für einmal erzeugte Gesichtsausdrücke und Körperposen eine Pose Library erstellen, welche später wiederverwendet werden kann. Der Weg über ein Gesichts-Rig schließt übrigens Shape Keys nicht aus, sollte nachträglich eine Korrektur oder Überspitzung einzelner Gesichtsausdrücke nötig sein.

Der Übersichtlichkeit halber wurden alle animierbaren Bones auf nach IK und FK getrennte Bone Layers gelegt, damit bei der Animation irrelevantes ausgeblendet werden konnte. Letztendlich wurden für diese Bones geeignete Darstellungen erstellt (sogenannte Custom Shapes), welche sie im 3D-View-Fenster als bequemere Anfasser repräsentierten.

Mesh Deform, Parent to Vertex und das Child Of Constraint

Nicht nur die Gnom-Geometrie sollte den Knochen folgen, sondern auch die übrigen Elemente wie Moos, Flechten, Gurt, usw. Hierfür bietet sich unter anderem der Mesh Deform Modifier an. Dieser verwendet ein Low-Poly-Mesh, das den Character wie ein etwas größeres Gitter umspannt und alle Objekte in seinem Einflussbereich, die an ihn gebunden sind, mit sich nimmt. Das Low-Poly-Mesh wurde ebenfalls über die Gnom-Armature verformt und war gleich gewichtet wie die Hauptgeometrie.

Dies war wichtig, weil sonst bei der Animation kleinmaschige Geometrie-Anhänger auf dem Gnom-Körper herumrutschen, darin verschwinden oder darüber schweben. Um mühelos dieselbe Gewichtung für das Low-Poly-Mesh zu erhalten, kann die fertig ge-

wichtete Character-Geometrie kopiert, von unnötigen Modifiern (z.B. Subdivision Surface) befreit, etwas skaliert und durch Herauslösen von Loops und Zusammenfassen von Polygonen weitgehend vereinfacht werden.

Zudem wird sie im Outliner als nicht sichtbar für die Kamera markiert (Restrict Rendering) und der Display-Modus auf Wire gestellt. Dieses Low-Poly-Mesh sollte alle gebundenen Elemente voll umspannen, aber auch nicht unnötig groß sein, da sonst sein Einflussbereich ungenauer wird und die Elemente nicht präzise auf der Gnom-Geometrie kleben.

Für den Grasgurt war vorgesehen, dass er sich einerseits mit der Character-Geometrie verformte, andererseits separat animierbar blieb, um bei Extremposen eventuellen Geometrie-Verdeckungen entgegenwirken zu können oder Aktionen wie Verrutschen oder Hängenbleiben zu ermöglichen. Deshalb wurde er mit einer vom Gnom getrennten Armature versehen, deren Bones die Bewegungen des Gnoms übernehmen, aber auch zusätzlich über IK-Targets animierbar bleiben sollten. Um das zu bewerkstelligen, wurde jeweils pro Gurt-Target-Bone ein Empty an eine Dreier-Vertex-Gruppe des Rumpfes geklebt (Parent To Vertex Triangle), und an dieses Empty der Gurt-Target Bone im Pose Mode mittels Child Of Constraint angeheftet. Resultat: Gurt-Target-Bone geht mit Empty geht mit Rumpf-Vertices, Gurt-Target-Bone bleibt aber zusätzlich animierbar. Der Vorteil einer eigenen Armature gegenüber



Mittels Mesh Deform Modifier werden auf dem Körper liegende Flechten, Moos und Hautschuppen durch die grobe Geometrie (Orange) mitverformt.



Die Verschmelzung mit dem Filmmaterial erfolgt über das Simulieren von Schattenwurf und Beleuchtung, die Farbanpassung des CG-Gnomes und das Abmaskieren verschiedener Bild-Elemente (Blätter, Steine, Tannenzapfen,...) vor der Geometrie.

Hooks und Vertex Groups ist das einfache Zurücksetzen einmal verschobener Target Bones über Clear Transform (Alt+G,+R oder +S), womit sie wieder an ihre relative Ursprungsposition springen.

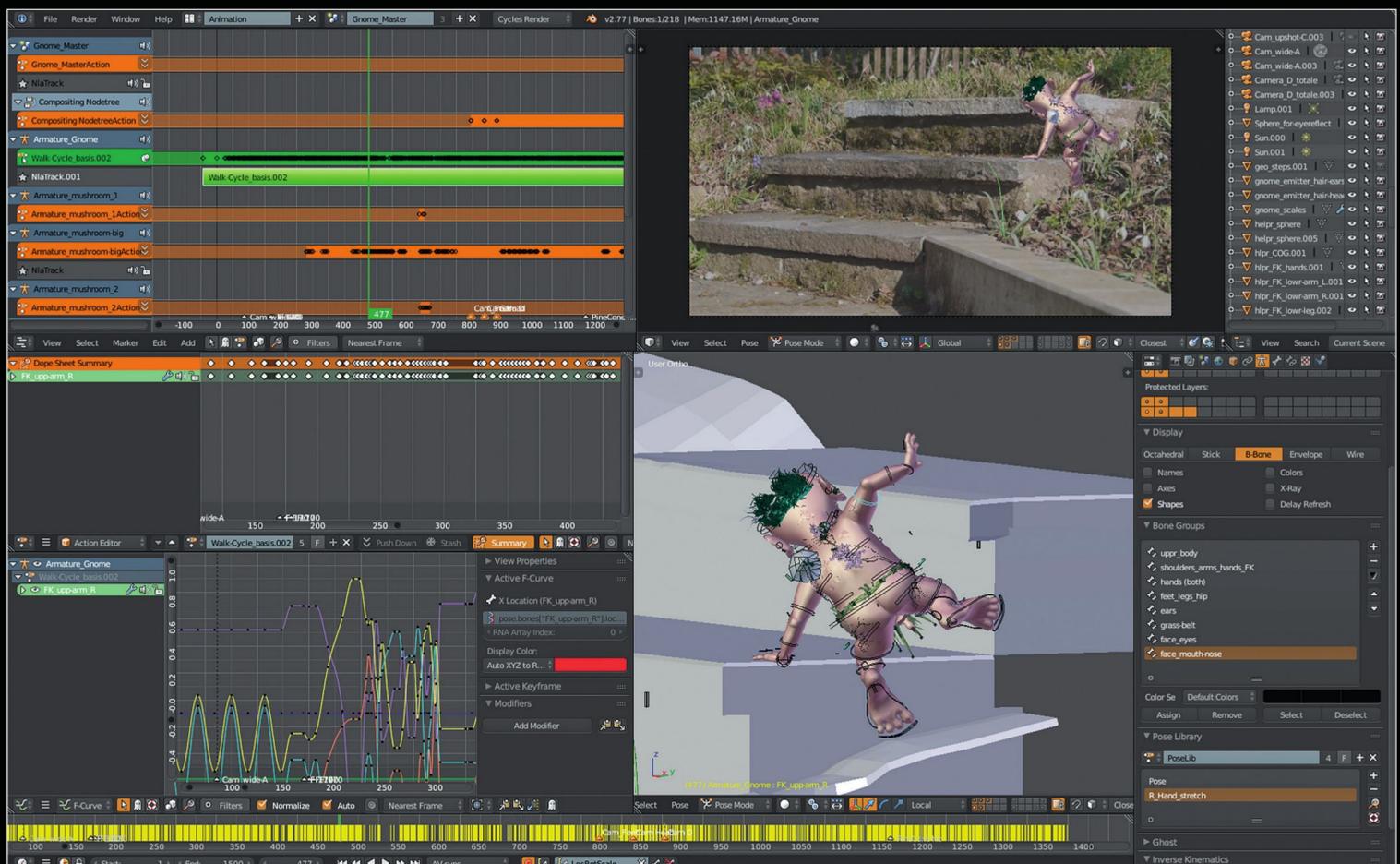
Einzelne Grashälme des Gurtes bekamen eigene Bones zum Ermöglichen von Sekundäranimationen. Die größeren Pilze wurden aus demselben Grund mit separaten Armatures versehen, welche am Rücken der Gnom-Geometrie klebten: Hierfür wurde die Gnom-Geometrie selektiert, im Edit Mode die gewünschten drei Vertices ausgewählt, zurück im Object Mode zuerst die Pilz-Arma-

ture, dann die Gnom-Geometrie selektiert, Parent To Vertex Triangle (Ctrl+P). Der Zweig im Haar wurde direkt an den Kopf-Knochen geklebt (Parent To Bone im Pose Mode).

Animation

Blender stellt für das Animieren von Charakteren vielfältige Kontrollmöglichkeiten über Dope Sheet/Action Editor, NLA Editor (Non-LinearAnimation), Graph Editor und Timeline zur Disposition, wobei das System sowohl eine lineare als auch eine nicht lineare Arbeitsweise zulässt.

Das heißt, man kann entweder eine Szene im Dope Sheet und Graph Editor von Anfang bis Ende animieren, oder man erstellt für repetitive Bewegungsabläufe im NLA Editor einzelne Animationschnipsel, sogenannte Action Clips, die man später in beliebiger Reihenfolge zeitlich anordnet, vervielfältigt oder miteinander kombiniert. Mit Drücken des Pfeilknopfs im NLA Editor (Push Action down...) werden jeweils die Keyframes einer Animation zu einem solchen Action Clip zusammengefasst. Dessen Keys können umgekehrt im Action Editor jederzeit isoliert editiert werden (Tab-Taste). So macht es



Blocking-in und Recycling von Animationschnipseln mittels NLA-Editor und Action Editor

beispielsweise Sinn, einen erstellten Walk Cycle als Action Clip zu speichern, diesen in den NLA Editor zu laden und nach Bedarf auf der Zeitachse zu vervielfältigen, langsamer oder schneller zu machen, und individuelle Kopf-/Augen-/Armbewegungen separat darüber zu animieren. Über „Append“ können diese Action Clips auch in andere Blender Files importiert und dort auf einen Character appliziert werden, sofern dessen Armature-Struktur und Bone-Benennung der ursprünglichen entspricht. In der Praxis erwies es sich jeweils als vorteilhaft, für anfängliches Blocking solche Action Clips zu verwenden, dann aber mit zunehmender Verfeinerung der Animationen sich überlagernde Action Clips wieder ineinander zu kopieren. Das gewährleistete die Übersichtlichkeit beim Finetuning und vereinfachte das Manipulieren einzelner Keys.

Compositing

Nach dem Import der Realbildaufnahmen als Hintergrund in Blender erfolgte die Anpassung der Szenen-Kameras auf die individuellen Shots. Da die bisherigen Szenen auf Stativ gedreht worden waren, entfiel hierfür das Kamera-Tracking. Für die virtuelle Kamera ist es natürlich hilfreich, möglichst viele relevante Metadaten zu erfassen, welche beim

Einpassen als Referenz dienen. Bei genügend stürzenden und parallelen Linien in der Szene ließe sich hier das Add-on Blam verwenden, welches den Einpassungs-Prozess erheblich erleichtern kann. Die Angleichung der virtuellen Beleuchtung an das Licht des Realbilds erfolgte für den diffusen Lichtanteil über eine entsprechende HDRI-Textur, während für das direkte Sonnenlicht und zusätzliches Finetuning eine klassische 3-Punkt-Beleuchtung (Key/Fill/Rim) aufgesetzt wurde.

Neben der Ausleuchtung war auch der Schattenwurf des CG-Characters auf seine real gedrehte Umgebung wichtig für eine überzeugende Integration. Das erforderte das Nachbauen aller relevanten Flächen, wobei diese per Render Layers nur den Gnom-Schatten empfangen und separat gerendert wurden. Da Cycles in Blender den Vector Blur bisher noch nicht unterstützt (Version 2.77), wurde die ganze Szene kopiert und die Animation in Blenders Internal Renderer mit der Vector-Blur-Information als .exr-Datei separat ausgerendert, welche im Compositing dann für den Vector Blur Node die Speed- und Z-Kanal-Information lieferte. Die Verschmelzung von Gnom, Schatten, und Realbild fand in einem weiteren Compositing Setup statt, worin schließlich Farbkorrektur, Kontrast, Körnung und Vector Blur individuell für jeden Input Node und die Abmaskierung

einzelner vor dem Charakter liegender Realbild-Elemente (Grashalme, Bodenelemente) vorgenommen wurden. Das fertige Compositing wurde als Einzelbildserie ausgerendert, im Sequencer mit einer in externer Software erstellten Tonspur unterlegt und mit Titeln als Film ausgespielt.

Zum Schluss: Blender wird kontinuierlich und in rasendem Tempo weiterentwickelt. So sind beispielsweise bezüglich Armature-Struktur in der neuesten Version 2.78 (Release: Oktober 2016) die Bendy Bones sehr interessant, anhand welcher sich das Rigging gewisser Partien nun wohl einfacher gestalten lässt – so zum Beispiel im Bereich Mund und Augenbrauen. Und wer den fertigen Gnom sehen will: vimeo.com/184651505. >ei



Titus Fehr ist diplomierter Architekt ETH und hat an der Vancouver Film School eine Ausbildung in „3D Character Animation & Digital Effects“ absolviert. Er ist Inhaber der Firma Momotion bei Zürich und erstellt seit über 14 Jahren professionelle CG-Character- und Prozess-Animationen, Visualisierungen und digitale Welten für Kunden aus verschiedenen Bereichen wie Werbung, Design, Industrie, Medizin. Neben Auftragsarbeiten produziert Momotion auch Kurzfilme in Eigenregie. Seit mehr als vier Jahren wird Blender als hauptsächliches Werkzeug für fast alle Bereiche des CG-Workflows verwendet. www.momotion.ch



Abmaskieren und Compositing der finalen Szene