

Kalamkarian Bechstein – Teil 2

EINE SAMPLE-LIBRARY ENTSTEHET



Foto: Beilharz

Ein Haus zu bauen, ein Kind zu bekommen und einen Flügel zu sampeln – dies sind die letzten Abenteuer dieser Erde: Wenn man vorher wüsste, was man sich da einbrockt, würde man sich das alles noch mal gut überlegen. Rund 830 Samples einzeln nachzubearbeiten hat eher etwas Masochistisches als etwas Kreatives. Doch das Ergebnis ist die Qualen letztlich wert. Wie bei jedem guten Abenteuer.

von Roman Beilharz

► Für dieses Projekt habe ich mich mit Stefan Kortner von Pianowave (www.pianowave.de) zusammengetan, um den vielseitigen, farbenreichen Klang meines Bechstein-Konzertflügels von 1892 der Nachwelt in Form einer Sample-Library im Kontakt 2-Format zu erhalten. Der Flügel stammt aus der Erbmasse der 1988 im Alter von 85 verstorbenen armenischen Pianistin Maria Kalamkarian. Diese hatte sich nach dem Ersten Weltkrieg in Deutschland niedergelassen und war eine weithin geachtete Interpretin der großen, romantischen Klavierliteratur (Tschaikowski, Rachmaninoff).

Für alle, die den ersten Teil verpasst haben, in aller Kürze eine Übersicht über das, was bisher geschah: Nach einer frischen Stimmung des Instrumentes hatten wir den Jazz-Pianisten Donato Deliano zur Aufnahme

verschiedener Musikbeispiele im Studio, um den Flügel optimal mikrofonieren und abmischen zu können. Unsere finale Konfiguration bestand aus zwei Microtech-Gefell M930 in saitennaher ORTF-Anordnung, zwei SE3a als Stützmikrofone in einer halbnahe AB-Aufstellung seitlich des geöffneten Flügeldeckels, einem polaritätsinvertierten AT 4050 unter den Bässen und vier Røde NT1000A als ABCD-Array direkt über den Wirbeln. Schließlich wurden alle 88 Töne des altherwürdigen Instrumentes in sechs Dynamikstufen gesampelt, wobei eine elektromagnetische Spezial-Konstruktion von Stefan Kortner die Tasten konsistent mit einer präzise definierbaren Stärke anschluss – und das vollkommen geräuschlos. Die zusätzlichen Sustain- und Release-Samples, Pedalgeräusche sowie die „nackten“ Rückfallgeräusche

der Hammermechaniken wurden manuell angeschlagen bzw. mit dem Fuß getreten. Alle Samples wanderten in Form von neun separaten Audio-Kanälen mit 24-Bit/88,2-kHz-Auflösung unter Samplitude 10 Pro auf die Festplatte. In diesem Teil steht es nun an, die so erstellte Fülle von Klangmaterial zu organisieren, zu editieren und schließlich zu einem spielbereiten Multi-Sample zusammenzustellen.

Prozesskonzeption

Stefan Kortner hatte bereits für seinen beliebten Pianowave Steinway-B (erhältlich unter www.soundsondemand.com) ein Kontakt-Template mit vielen Basis- und Spezialfunktionen erstellt, die mit Hilfe eines Kontakt-Skriptes realisiert worden waren. Dieses Template erwartet eine bestimmte Layer-

Struktur und standardisierte Sample-Namen. Wenn diese beherzigt werden, erhält man im Handumdrehen einen spielbaren Sample-Flügel, der sich dann mit den reichlich vorhandenen Kontakt-Bordmitteln verfeinern, justieren und abstimmen lässt. Meine selbst gewählte Aufgabe besteht darin, unsere Aufnahmen so aufzubereiten, dass sie in dieses Kontakt 2-Template importiert werden können. Dazu müssen alle mehrspurig vorliegenden Samples in Samplitude 10 Pro abgemischt, als 24-Bit/44,1-kHz-Stereo-Dateien heraus gerendert, normalisiert, geschnitten, ggf. mit Fade-outs versehen sowie schließlich in der Form umbenannt werden, die das Pianowave-Skript erwartet. Falls Sie sich nun fragen sollten, was es gebracht hat, die Aufnahmen in 88,2 kHz zu machen, wenn man sie am Ende ohnehin in 44,1 kHz umwandelt, so lautet die Antwort: Erstens, weil die Stereo-Summierung der Einzelkanäle im Mix mit höherer zeitlicher Auflösung besser klingt und zweitens, weil EQs und andere Effekte bei hohen Raten mit weniger Rundungsfehlern zu kämpfen haben – und daher ebenfalls besser klingen; auch nach dem Downsampling auf 44,1 kHz. Um die Prozesskonzeption konkret zu erarbeiten, nehme ich mir einige Samples als „Versuchskaninchen“ und probiere verschiedene Varianten durch. Es stellt sich heraus, dass sich mit Wave-labs Stapelbearbeitung zwar alle Samples wunderbar in einem Rutsch normalisieren lassen, doch die Funktion zum masseweisen Sauberschneiden der Anfänge bzw. Enden der Samples lässt sich nicht in die Stapelbearbeitung integrieren. Zudem merke ich, dass diese Funktion zwar die Anfänge der Samples zuverlässig nach Pegel-Schwellwert schneiden kann, die Enden jedoch aufgrund unbrauchbarer Ergebnisse manuell geschnitten und ausgeblendet werden müssen. Schade eigentlich.

Hybrid-Bit Technology

Immerhin funktioniert meine Idee, nur die drei lautereren Dynamik-Layer (im Pianowave-Jargon: VelD bis VelF) in 24 Bit zu belassen, während die drei leiseren Layer (VelA bis VelC) auf 16 Bit „heruntergedithert“ werden. Kontakt 2 verdaut die verschiedenen Wortbreiten anstandslos – auch innerhalb derselben Sample-Group – und dankt es mit einer besseren Gesamtperformance beim Streaming. Außerdem braucht das Multisample so im Ende weniger Speicherplatz, was wiederum schnellere Downloads ermöglicht (es soll nach Fertigstellung primär online

angeboten werden). Hörbare klangliche Nachteile ergeben sich durch diese Teil-Reduzierung nicht, denn wenn alle Samples vor dem Downsampling normalisiert werden, stehen auch bei den leisen Layers alle Bits voll für das Nutzsignal zur Verfügung. Um ein dynamisches Spielverhalten zu erzeugen, wird der Pegel aller – messtechnisch zunächst gleich lauten – Dynamiklayer erst im Sampler der eingehenden MIDI-Velocity zugeordnet und entsprechend leiser abgespielt. Schlägt man nun eine Taste mit einer Velocity von z. B. 24 an, so erklingt bei sechs Dynamiklayern vermutlich das zweitleiseste Layer in einer Lautstärke, welche der Original-Dynamik des Instrumentes entspricht. Ein realistischer Wert wäre dabei eine Bedämpfung des Samples um ca. 22 dB. Ein 24-Bit-Sample hätte dann noch eine Netto-Wortbreite von ca. 16 Bit, mit der es in den 32-Bit-fp-Raum des Hosts übertragen wird. Ein normalisiertes 16-Bit-Sample geht im selben Falle ebenfalls mit ca. 16 Bit in den Sampler-Ausgang – der Verlust ist quasi gleich Null. Nur die lauten Dynamiklayer eines Samples werden laut genug – und damit in voller Bit-Auflösung – an den Host weitergereicht; erst hier machen 24 Bit Sinn. Bestenfalls, denn bei vielen Multisamples, die in zwei Varianten dargeboten werden (z. B. Galaxy Piano I) sind die 16-Bit-Versionen gehörmäßig kaum von den 24-Bit-Versionen zu unterscheiden. Insofern kann man Sampledaten nicht mit Live-Audioaufnahmen vergleichen, bei denen aufgrund ihrer internen Dynamikstruktur 24 Bit in jedem Falle sinnvoll sind. Daran sollte man denken, wenn mal wieder jemand lautstark Instrumenten-Samples in 16-Bit als „Schmal-

spur-Audio“ abtut. Wir sitzen da immer wieder einem Marketing-Hype auf, der uns suggeriert, 24 Bit seien in jedem Fall besser als 16 Bit.

Zurück zu unserem Projekt: Die Kombination von 16- und 24-Bit-Samples funktioniert also prima. Da mir nicht bekannt ist, dass ein Sample-Entwickler ein solches Konzept schon einmal umgesetzt hätte, sehe ich es plötzlich klar vor meinem geistigen Auge: Ein Logo mit der Aufschrift „Hybrid-Bit Technology“. Tja, auch das ist Marketing – jeder halbwegs brauchbaren Konzeption einen klangvollen Namen zu verpassen, die das Ganze wie einen Geniestreich aussehen lässt. Und ganz wichtig: Der Name muss auch als Abkürzung funktionieren: HBT. – Toll: „Die erste Sample-Library der Welt mit HBT“. Doch nun frisch ans Werk; unsere erste Amtshandlung heißt:

Schritt 1: Samples entrauschen

Dies ist bei aller Liebe zum Originalsignal bei einem Multisample unerlässlich. Warum? Stellen wir uns einmal vor was passiert, wenn wir bei einem gesampleten Instrument einen Akkord greifen. Nun werden im Prinzip die Aufnahmen aller ausgelösten Töne simultan abgespielt. Dabei addieren sich jedoch nicht nur die tonalen Anteile, sondern auch das Rauschen aller Einzelaufnahmen. Bei einem Vierklang mit Doppelbass wird das bei der Aufnahmesession vorhandene Eigenrauschen quasi versechsfacht. Da sechs unserer neun bei der Aufnahme verwendeten Mikrofone ein extrem geringes Eigenrauschen besitzen (M930 und NT1000) und auch die SPL-



Schritt 1: Der TC DeNoiser von der PowerCore sorgt dafür, dass auch volle Akkorde in den leisen Layers rauschfrei bleiben werden.



Schritt 2: Alle klanggestaltenden Bearbeitungen auf einen Blick: Pegel- und Panoramaverhältnisse, zwei HydraTone-Instanzen, zwei Oxford-EQ-Instanzen und bx_control in der Summe.

Mikrofonvorstufen mit einem überdurchschnittlichen Signal-Rauschabstand glänzen, fällt dies allerdings nur bei den leisen Dynamiklayern wirklich ins Gewicht. Nichtsdestotrotz verwende ich je einen TC DeNoiser von der PowerCore auf den beiden Haupt-Mikrofonsystemen; die vier Rødes fasse ich zu einer Subgruppe zusammen und entrausche sie in einem Rutsch. TCs DeNoise bedient sich eines sogenannten „Noiseprints“, eines Probeschnipsels des Signalrauschens alleine. Dazu ist es wichtig, bereits bei der Aufnahme genügend „Stille“ mit aufzunehmen, die man später dem DeNoiser nach einem Klick auf den Learn-Button anbieten kann.

Anschließend wird das Spektrum des Noiseprints als gelbe Kurve angezeigt. Nun regelt man das Ceiling (CEIL) so weit herunter, dass es gerade so über der lautesten Stelle dieser Kurve liegt. Alle Signalanteile oberhalb des Ceilings bleiben von jeder Entstellung verschont; lediglich die meist extrem hoch- und tieffrequenten, niedrigpegeligen Störschalle unterhalb des Ceilings werden anhand des Noiseprints um den eingestellten Reduktionswert (RED) abgesenkt. Ein Wert von -15 dB sollte hier auch für die leisen Töne genügen. Es ist immer wieder faszinierend, wie artefaktfrei und perfekt TCs DeNoise diese Aufgabe erfüllt.

Schritt 2: Mischung erstellen

Ich beginne nun mit der Abmischung der Hauptmikrofone, indem ich das VIP („Virtuelles Projekt“ in Samplitude) der Musikbeispiele unseres Jazz-Pianisten öffne und die mit „Close“ benannte Spur (zwei M930 in saittennaher ORTF-Konfiguration) solo schalte. Aus Erfahrung weiß ich, dass der Flügel im verwendeten Aufnahmeraum bei aller Pracht etwas nasal und besonders bei hartem Anschlag in der Mittellage leicht gepresst klingt, da die Energie in den unteren Mitten rund um 800 Hz überbortet. Ich senke diesen Bereich mit meinem Lieblings-EQ HydraTone um ca. 4 dB ab, entschlacke die Tief-

► Tipp: Samplitude auftakeln

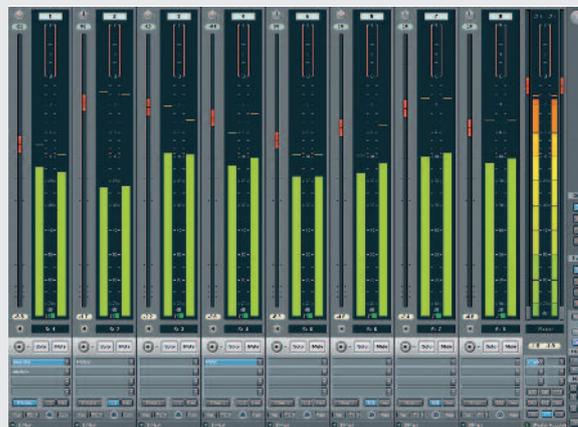
Keine Frage, Samplitude 10 Pro von Magix ist eines der leistungsfähigsten Recording- und Mastering-Programme auf dem Markt – und mit Sicherheit das mit der besten Ressourcen-Effizienz. Leider wirkt dessen Bedien-Konzeption nicht so stringent und intuitiv, wie man es z. B. von den Magix-Video-Programmen gewohnt ist; hier kommen sich immer noch zu viele für sich genommen sinnvolle Ansätze in die Quere.

Auch das Standard-GUI wirkt trotz beständiger Aufhübschungen bei großen Versionssprüngen recht altbacken. Dies lässt sich ändern: Orange Hill Audio bietet unter www.pixelarranger.de/alloy diverse Farb-Varianten seiner schicken und ergonomischen Alloy-Skin für Samplitude an.

Die meisten auf verschiedene Zwecke optimierte Mischpult-Ansichten (z. B. Kopfhörer-Mischung) beinhaltet die Silver-Alloy-Skin, kostet mit 49.99 US-Dollar im Online-Shop jedoch auch ein hübsches Sümmchen. Für 10 Dollar weniger bekommt man die Dark-Alloy-Skin in dunklem Anthrazit-Look.

Damit verpasst man Samplitude Pro endlich den Look und die Übersichtlichkeit, den das Programm schon lange verdient hat. Magix sollte sich überlegen, die Alloy-Skins zum Bestandteil von

Samplitude 11 (oder „Pro-X“?) zu machen, denn dass man hier für einen zeitgemäßen Look so viel drauflegen muss, tut schon ein bisschen weh.



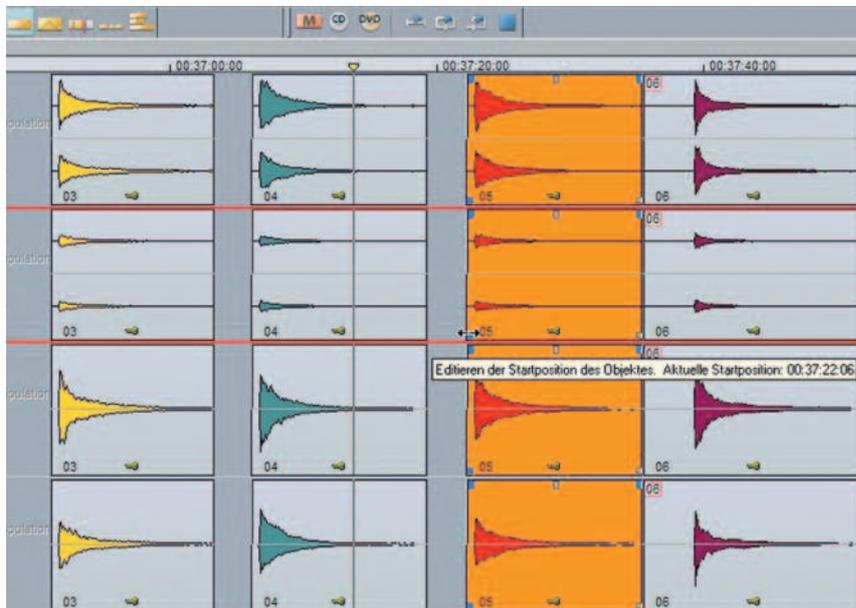
In der Big-Meters-Ansicht der Dark-Alloy-Skin kann man auch lange Plug-in-Namen komplett lesen.

bässe prophylaktisch mit einem tief ansetzenden Low-Cut und verpasse den Höhen mit dem Höhenfilter einen leichten „Glanz“ bei rund 15 kHz (im Screenshot die obere HydraTone-Instanz). Die Stütz-Mikrofone („Room“) räume ich lediglich mit einem High- und einem Low-Cut-Filter in den Frequenz-Extremen auf (s. obere Oxford-EQ-Instanz) und nehme sie insgesamt um 7,4 dB zurück. Dadurch verleihen sie dem Haupt-Stereosignal eine dezente Räumlichkeit und Wärme, ohne dass der Gesamtklang an Direktheit verliert.

Schließlich soll das Flügel-Sample im Ende trocken genug klingen, um sich im Mix mit beliebigen Raum-Effekten nach Wahl verhalten zu lassen. Die Bass-Stütze (AT 4050) begrenze ich mit dem Oxford-EQ spektral so drastisch, dass dieser Kanal hauptsächlich da zum Tragen kommt, wo die untere Grenzfrequenz der M930 liegt – nämlich bei 40 Hz (s. untere Oxford-EQ-Instanz). Mit -4,5 dB hinzugemischt, bekommt der Flügel dadurch mehr „Bums“ in den Bässen. Das Panorama der Stützmikrofone wird gehörmäßig so korrigiert, dass sich deren Stereopanorama möglichst kongruent zu den Hauptmikrofonen in der Horizontalen aufstaffelt. Dabei hilft unsere in weiser Voraussicht erstellte Aufnahme einer chromatischen Tonleiter auf- und abwärts. Nach dem ersten Hinzuschalten der vier Røde NT1000A bin ich mir nicht sicher, dass sie dem bereits recht eindrucksvollen Klang noch viel zu geben haben; durch deren große Nähe zu den Saiten klingen sie im Vergleich zu den bisherigen Kanälen etwas wummerig und topfig.

Nachdem ich alle unerwünschten Signalanteile mit einer weiteren HydraTone-Instanz eliminiert habe (im Screenshot unten), gewinne ich jedoch den Eindruck, dass der Flügel mit dem so bearbeiteten Røde-Array minimal lebendiger, näher und wärmer klingt. Um diesen Mehrwert vorsichtig gegenüber den restlichen Signalen abstimmen zu können, schicke ich die drei ersten Kanäle auf eine eigene Subgruppe („Main“), die vier Rødes hatte ich bereits beim Entrauschen in einer eigenen Subgruppe zusammengefasst („Direct“). Sie gehen um 3,5 dB leiser in die Stereo-Summe ein als die Main-Signale. Dort nutze ich schließlich den Mono-Maker des Brainworx bx_control, um dafür zu sorgen, dass sich alle Bass-Anteile unter 80 Hz gleichphasig verhalten; eine Maßnahme, die ich inzwischen fast in jedem Mastering-Projekt standardmäßig durchführe. Die Mischung speichere ich im Amplituden-

Schritt 3: Anfänge grob setzen



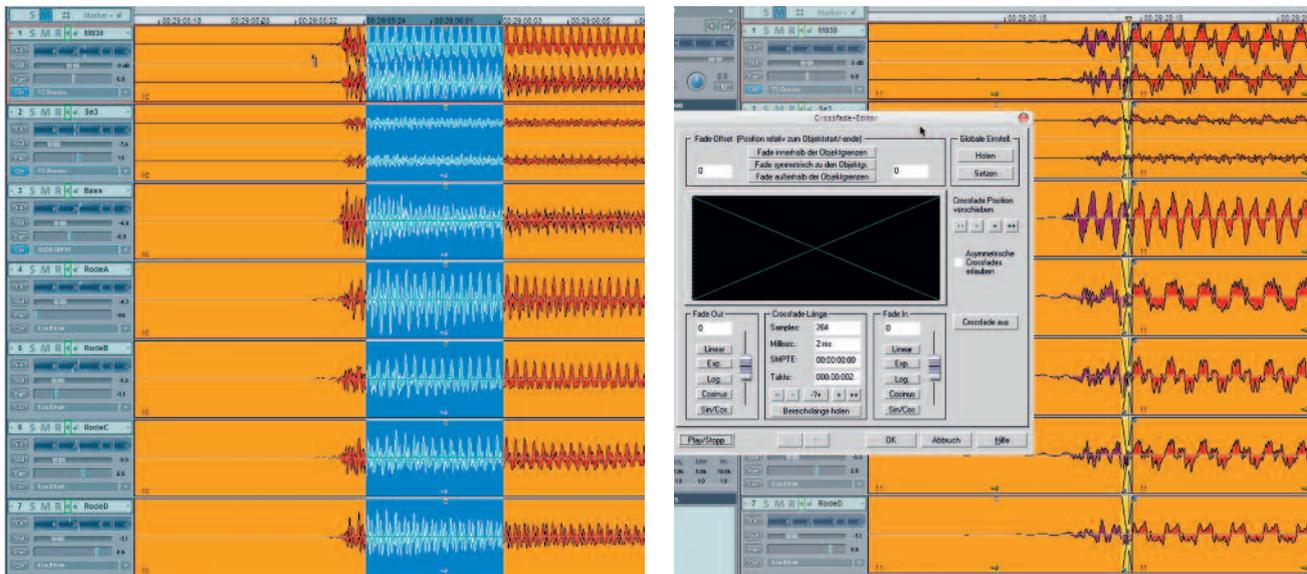
Schritt 3: Damit das automatische Sauberschneiden der Sample-Anfänge funktioniert, dürfen keine Störgeräusche im Vorlauf sein.

Mixer als MIX-Datei ab, öffne nun das Projekt mit den Schwarze-Tasten-Samples und erzeuge die benötigten Subgruppen, denn diese erzeugt Amplitude bislang noch nicht zuverlässig beim Übertragen einer Mischung. Dann kann ich die Mischung laden und mich an den Grobschnitt der Samples begeben. Damit nämlich das automatische Beschneiden der Sample-Anfänge funktioniert (s. Schritt 7), dürfen im Vorlauf keine Störgeräusche wie Sprache („Läuft“) oder Klackern sein. Andernfalls hält Wavelab diese Klangergebnisse für den gewünschten Startpunkt. Es macht trotzdem Sinn, dessen Automatik nebst manueller Vorbereitung zu verwenden, denn das exakte Schneiden des Sample-Starts von Hand dauert um ein Vielfaches länger, als nur mal eben „auf Sicht“ – also anhand der Wellenform-Darstellung – den Anfang jedes Audio-Objektes grob zurechtzuschieben. Wavelab braucht anschließend für den automatischen Schnitt von Hunderten von Samples kaum mehr als eine Minute. Mit den übrigen Sample-Sessions (weiße Tasten, Sustains und Releases) verfare ich dementsprechend und komme zu.

Schritt 4: Trommeln entfernen

Schon bei der Aufnahme war uns aufgefallen, dass manche Töne des alten Bechstein zum Trommeln neigen. So nennt man das,

wenn es beim Drücken einer Taste zu einem Doppelanschlag kommt. Anstatt sanft zurückzuschwingen, trifft das Hämmerchen die Saite(n) ein zweites Mal – meist sehr leise, aber dennoch hörbar. Bei Live-Aufnahmen war das Problem nie störend aufgefallen; kein Pianist hatte das Trommeln bisher moniert. Bei einer derart detaillierten Mikrofonierung ist es jedoch, als würde man das Instrument mit einem Stethoskop abhören. Und je länger man dies tut, desto schmerzlicher nimmt man solch einen Mangel wahr. Ich entscheide mich also letztlich trotz aller Liebe zur Authentizität für eine Schönheits-OP: Das Trommeln soll weg. Ich komme daher nicht drum herum, schon jetzt alle Samples komplett durchzuhören und den „Trommlern“ unter ihnen den lästigen Zweitanschlag herauszuoperieren. Damit dieser Eingriff möglichst unhörbar gelingt, muss man beim Schnitt um jeden Schwingungsdurchlauf feilschen, damit so viel wie möglich brauchbares Klangmaterial erhalten bleibt, ohne dass sich plötzliche Änderungen im Klangcharakter ergeben. Eine Crossfade-Länge von 2 ms bei linearem Pegelverlauf stellt sich als gute Allround-Einstellung für diese Schnitte heraus – dank Crossfade-Editor in Amplitude kein Thema. Anschließend muss der zweite Teil des geschnittenen Audio-Objektes im Pegel etwas angehoben werden, damit das



Schritt 4: Den gelegentlichen Doppelanschlag der alten Hammermechanik (Trommeln) zu entfernen erfordert einiges an Handarbeit.

Anschlagsgeräusch nach dem Normalisieren nicht überproportional laut wirkt. Auch wenn ich von Sample zu Sample schneller werde, so dauert diese Prozedur unzählige Stunden.

Schritt 5: Stereo-Dateien rendern

Anschließend können alle Samples Session für Session per Export als Stereo-Dateien mit 24 Bit/44,1 kHz „herausgerendert“ werden (Offline-Bouncing). Doch damit Samplitude nicht eine einzige, ellenlange Datei mit allen Samples einer Session erzeugt, muss dem Rendern stets der Befehl zum Erzeugen von Track-Markern an den Objektkanten vorausgehen. Wenn man dies mit der entsprechenden Export-Option kombiniert, behält Samplitude die Objektfänge bei und spuckt lauter einzelne Stereo-Dateien aus. Für das Downsampling reicht die zweithöchste Qualität aus, denn bei der Konvertierung von 88,2 auf 44,1 kHz hat man ein gradzahliges Teilungsverhältnis, sodass eine höhere Präzision hier nichts bringt: 8 : 2 bleibt 4, da helfen keine Pillen. Übrigens wählten wir im Vorfeld diese Kombination von Sampling-Raten, da die meisten Musiker immer noch in 44,1 kHz arbeiten. Die in den Audio-Daten von Sampling-Libraries häufig anzutreffenden 48 kHz werden als Projekt-Clock hauptsächlich bei Filmmusik-Projekten verwendet. So wird – hoffentlich – die Mehrzahl der Nutzer dieser Sampling-Library ohne Kontakts internes Resampling arbeiten können, was deren Performance weiter verbessert. Ich nutze den POW-r-2 Dithering-Algorithmus (mit Noise-

Shaping), um die Samples aus dem 32-Bit-fp-Raum des Amplitude-Mixers in 24 Bit zu überführen. Samplitude merkt sich beim Export die letzten verwendeten Format-Einstellungen und Bouncing-Optionen, vergisst jedoch leider stets die letzte gewählte Sampling-Rate, sodass ich als Zielformat bei jeder Session erneut 44,1 kHz einstellen muss. Dann kann ich in aller Ruhe einen Kaffee trinken gehen, denn bei hunderten von Mehrspur-Samples in 88,2 kHz und relativ aufwendigen Plug-in-Prozessen hat Samplitude auf meinem betagten 2-GHz-Athlon eine gute Weile zu tun.

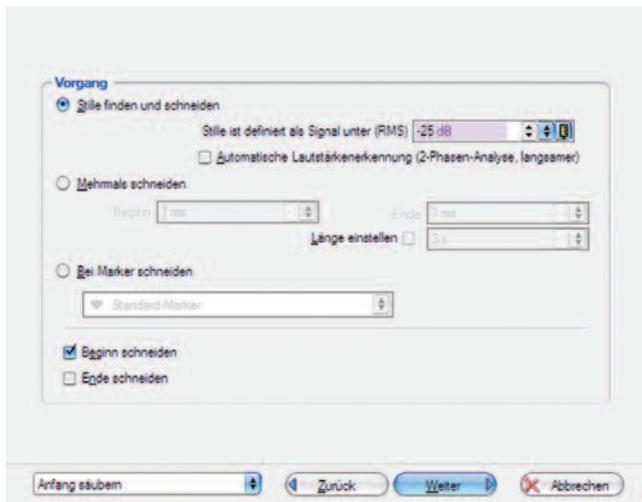
Schritt 6: Normalisieren

Nun kann ich endlich die Stapelbearbeitung von Wavelab 6 anwerfen, damit die Nachbearbeitung ein bisschen rückt. Unter Werkzeugen → Stapelbearbeitung öffnet sich ein Fenster, in welchem sich ggf. mehrere Ordner mit Audio-Dateien in einem Zug umrechnen lassen. Hier lasse ich nun alle Samples auf einen Spitzenpegel von -0,3 dB normalisieren. Dieser Mindest-Sicherheitsabstand zu 0 dB FS empfiehlt sich übrigens bei jeder Normalisierung, denn durch Inter-Sample-Peaking kann es nach der DA-Wandlung zu Verzerrungen kommen, auch wenn vorher nie mehr als sechs Samples in Folge 0 dB erreicht haben, was die meisten Peaking-Anzeigen einen Clip bzw. Overload darstellen lässt. Wer ganz sichergehen möchte, wählt einen Spitzenpegel von -0,5 dB beim Normalisieren. Das gleiche gilt für die Einstellung des Output-Ceilings in Brickwall-Limitern am Ende einer Prozesskette beim Mastering, es

sei denn, der Limiter verfügt über einen eingebauten Schutz vor Inter-Sample-Peaking (z. B. Sonnox Oxford Limiter).

Schritt 7: Anfänge säubern

Wavelab verfügt bereits seit Version 4 über eine Auto-Split-Funktion, welche Audio-Dateien anhand eines Pegel-Schwellwerts (zer-)schneiden kann. Dadurch können z. B. aus einer durchgängigen Live-Aufnahme mit Pausen zwischen den Titeln einzelne Dateien erzeugt werden. Alternativ lassen sich auch nur Anfänge und Enden automatisch sauber schneiden. Da sich diese Bearbeitung auch auf Ordner mit vielen Audio-Dateien anwenden lässt, wäre die Auto-Split-Funktion durchaus ein Kandidat für die Stapelbearbeitung, lässt sich jedoch nicht dort integrieren. Sie wird vielmehr separat gestartet und muss in der Art eines interaktiven Wizards über mehrere Seiten eingestellt werden, bis man die Bearbeitung schließlich ausführen kann. Ich setze den Eintrag „Stille ist definiert als Signal unter (RMS)“ auf den experimentell ermittelten Wert von -25 dB und wähle „Beginn schneiden“ aus. Im Gegensatz zu allen anderen Wavelab-Bearbeitungen lässt ein das Programm im Dunkeln, wann eine Auto-Split-Bearbeitung abgeschlossen ist. Eine Prozessanzeige gibt es hier nicht, erst ein Blick in den Zielordner schafft Klarheit. Wavelab hat nun alle Sample-Anfänge so geschnitten, dass der Beginn jedes Samples pünktlich mit dem „Nutzsignal“ startet. Wie ich später sehen werde, ist dies besonders bei leisen, tiefen Tönen zu früh, denn dem eigentlichen Ton geht hier ein tieffrequentes



Schritt 7: Mit Wavelabs Auto-Split-Funktion lassen sich Audio-Dateien stapelweise anhand eines Pegel-Schwellwerts schneiden.

Geräusch der Hammermechanik voraus. Spielt man solche Samples per MIDI, wirken die Tonanschläge verspätet, denn das Mechanik-Geräusch müsste „authentischerweise“ schon bei Beginn des Tastendrucks erklingen. Dieser wird jedoch nicht per MIDI übermittelt; erst wenn die Taste komplett gedrückt ist, wird das MIDI-Event mit dem entsprechenden Velocity-Wert gesendet. Es ist traurig, aber wahr: diesen Ton-Anteil muss man bei Sample-Flügeln über Bord werfen. Da diese Latenz bis zum Erklingen des eigentlichen Tones von Taste zu Taste unterschiedlich ist, werde ich die betreffenden Sample-Anfänge erst in Kontakt 2 nachjustieren.

Schritt 8: Enden optimieren

Es dauert bei tiefen Tönen eines Flügels bis zu einer Minute und länger, bis diese vollständig verklungen sind. In der Praxis hält man einen Ton jedoch selten länger als zehn Sekunden. Stefan Kortner hat daher auch seine bisherigen Sample-Flügel nur mit 10–12 Sekunden pro Ton gesampelt und längere Töne für den seltenen Fall der Fälle geloopt (z. B. Schlussakkorde). Ab der oberen Mittel-

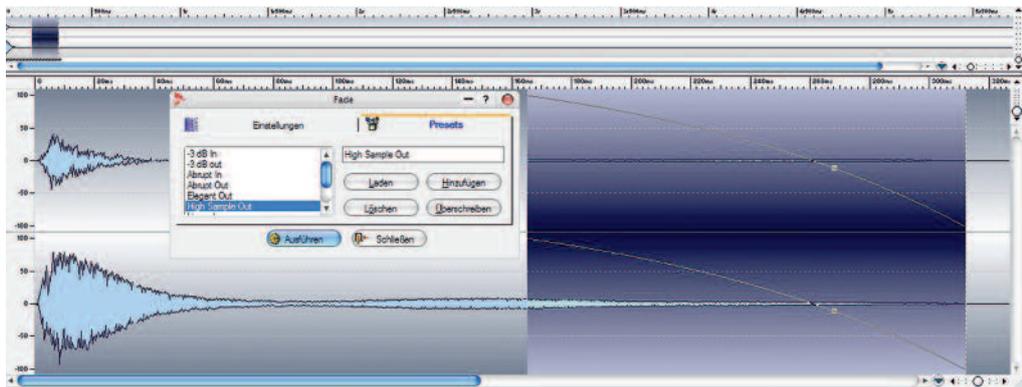
lage kommt man mit 10 Sekunden gut ohne Loops aus, im Diskant verklingen die Töne schon nach 6, 5 und ganz oben nach 3 Sekunden. Die zeitliche Begrenzung der einzelnen Samples war zudem nötig, um bei der Aufnahme-Session mit der zur Verfügung stehenden Zeit auszukommen. Positiver Nebeneffekt: Der Sample-Pool kann ohne nennenswerte Einbußen kompakter gehalten werden. Wenn man zudem noch alle Töne, die schneller verklingen als die Aufnahme lang ist, kürzt und ausblendet, verbessern sich zudem die Polyphonie-Werte, da dadurch nicht jedes kurze Tönchen eine meterlange Rauschfahne hinterher schleppt. Sie merken es schon: Statt auf Teufel kommt es uns eher darauf an, eine effiziente und dennoch klanglich hochwertige Library zu erstellen, die musikalisch Sinn macht. Kleiner Nachteil dieser Konzeption: Jedes Sample im Bass und in der unteren Mittellage muss auf Aufnahmelänge gebracht und geloopt werden, jedes Sample in der oberen Mittellage sanft ausgeblendet und jedes Sample im Diskant gekürzt und ausgeblendet werden. Von Hand.

Schritt 9: Leise Layer dithern

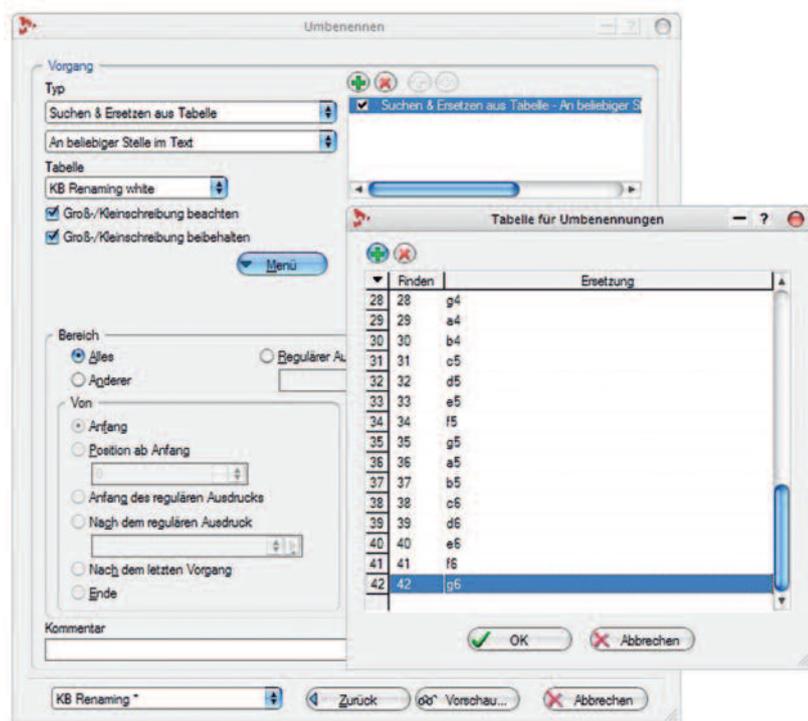
Einige Arbeitsstunden später: Um das Konzept der „Hybrid-Bit Technology“ umsetzen zu können, lasse ich Wavelabs Stapelbearbeitung die drei leiseren Velocity-Layer (VelA, VelB und VelC) per UV22 HR-Algorithmus auf 16 Bit herunterdithern. Das geht ratzfatz, und damit ist für den Import der Samples in Kortners Kontakt 2-Template nur noch Folgendes zu tun:

Schritt 10: Dateien umbenennen

Ich habe im Internet lange nach einem Umbenennungs-Tool gesucht, welches es erlauben würde, alle bislang stumpf anhand der Amplitude-Marker aufwärts durchnummerierten Dateien über eine Tabelle in Notennamen umzuwandeln. Denn das Pianowave-Skript erwartet Sample-Namen wie z. B. „VelF-c#7“; dieser Ton (C#7) entspricht der 32. schwarzen Taste im lautesten Dynamiklayer (VelF). Da wir aus technischen Gründen erst alle schwarzen und dann – in einem neuen Amplitude-Projekt – alle weißen Tasten aufgenommen haben, sind auch diese fortlaufend durchnummeriert. Die 32. weiße



Schritt 8: Je nach Oktavlage und Resonanzverhalten der Töne sind die Samples ganz unterschiedlich lang; hier ist eine manuelle Optimierung unumgänglich.



Finden	Ersetzen
01	a0
02	a#0
03	b0
04	c1
(...)	

Nun kann ich jeweils anhand der passenden Tabelle alle schwarzen und weißen Tasten, die Release und die Sustain-Samples, welche jeweils in verschiedenen Ordnern vorliegen, in vier Vorgängen umbenennen lassen. Hat man erstmal die Tabellen klar, sind 830 Samples in Sekunden umbenannt. Nun kann ich erstmals den kompletten Sample-Pool ohne Dubletten in ein und denselben Ordner kopieren, sodass Stefan Kortner den Import in das Pianowave-Template vornehmen kann.

Schritt 11: Erster Kontakt

Das erste Kontakt-Instrument, welches mir Stefan Kortner per Mail schickt, erfordert noch etliche Anpassungen. Wie bereits erwähnt, müssen die Sample-Anfänge etlicher Töne von Hand justiert werden. Im Zuge dessen ersetze ich nach und nach einige Samples, deren klangliche Probleme einzeln nicht aufgefallen waren. So manches Klackern im Attack muss noch dem SPL Transient Designer weichen; hier und da sind Töne separat zu filtern.

Bei Redaktionsschluss sind wir so weit, dass die Spielbarkeit des Kalamkarian Bechstein immerhin ein so hohes Niveau erreicht hat, dass es jedes Mal schwer fällt, mit dem Testspielen wieder aufzuhören.

Dieser Flügel macht wirklich Spaß. Die Dynamik-Ansprache mancher Taste ist noch nicht perfekt ausgewogen, aber nun ist es nur noch eine Frage von relativ wenigen letzten Schritten, bis wir die erste Version dieser Library zum Testen und Kaufen herauszurücken werden. Falls es Sie interessiert, sich unseren Flügel als kostenlose Demo-Version für Kontakt 2 (und 3) herunterzuladen, so schauen sie doch ab Anfang März mal unter www.pianowave.de herein. ■

Schritt 10: Dank Wavelabs hoch flexibler Stapel-Dateiumbenennung lassen sich die rund 830 Samples in vier Tranchen umbenennen.

Taste entspricht z. B. dem D5. Um es kurz zu machen: Ich war schon völlig verzweifelt und dachte, ich müsste über 800 Samples von Hand umbenennen, da keines der gängigen Freeware-Tools die gewünschte Funktionalität besaß. Da entdeckte ich wie durch Zufall ein weiteres Wavelab-6-Werkzeug, welches ich die ganze Zeit aus unerfindlichen Gründen geflissentlich übersehen hatte: Die Stapel-Dateiumbenennung. Diese erlaubte es mir, komfortabel und schnell drei Tabellen anzulegen: Eine für durchnummerierte weiße Tasten, eine für schwarze Tasten und eine für die Sustain- und Release-Samples, die chromatisch – also in Halbtönen hochgezählt – vorliegen. Nach dem Erstellen einer neuen Tabelle gibt man in diese für die Umbenennung der weißen Tasten Folgendes ein:

Finden	Ersetzen
01	a0
02	b0
03	c1
04	d1
(...)	

Nach einem ähnlichen Schema lassen sich die Namen für die schwarzen Tasten in einer weiteren Tabelle in Notennamen umwandeln:

Finden	Ersetzen
01	a#0
02	c#1
03	d#1
04	f#1
(...)	

Eine dritte Tabelle sorgt für die korrekte Umbenennung der halbtonschrittigen Nummerierungen:



Schritt 11: Endlich spielbereit: Kontakt-Instrumente lassen sich mit einer TGA-Grafik „skinnen“, um ihnen einen individuellen Look zu geben.