

Diphone Studio

Teil 1 — Additive Analyse und Synthese

Hans Tutschku
tutschku@ircam.fr

Der Artikel bezieht eine Sammlung von Beispielklängen ein, die auf folgender web-adresse angehört werden können:
<http://www.multimania.com/hanstutschku/diphone/diphone.htm>

Unter dem Name "Diphone Studio" verbirgt sich ein Paket von drei Programmen, die in den vergangenen Jahren am IRCAM für den Macintosh entwickelt wurden und deren Funktionalität aufeinander abgestimmt ist. Die beiden Analyseprogramme **AddAn** und **ResAn** gehen von bestehenden Soundfiles aus und reduzieren deren Daten auf Repräsentationen, welche später mit dem Interpolator von **Diphone** zu neuen musikalischen Einheiten komponiert werden können.

Diphone ist die Oberfläche zur Interpolation von Klangsegmenten und damit zur Kreation von neuen musikalischen Sequenzen. Das graphische Interface enthält mehrere Editoren und ist der Rahmen für verschiedene Synthese-PlugIns wie z.B. für Additive Synthese und für die Chant Synthese.

Der Editor der Bibliotheken erlaubt, Segmente in vorhandenen Bibliotheken auszuwählen, neue Bibliotheken zusammenzustellen und zu speichern.

Der Sequenz-Editor erlaubt das Zusammenstellen neuer Sequenzen aus vorhandenen Segmenten, deren Parameter frei änderbar sind (Zeit und Funktion der Interpolation eines Segments zum folgenden etc.).

Der BPF-Editor dient zum Manipulieren der *break point functions*. Alle Werte für Frequenzen, Amplituden, Phasen, Transpositionen usw. werden als solche BPFs gespeichert.

Die Segmente aus denen die neuen Sequenzen zusammengesetzt werden, wurden während der Analyse in Bibliotheken zusammengefaßt. Man kann also zuerst mit den mitgelieferten Bibliotheken arbeiten und aus den bestehenden Segmenten neue Klänge zusammensetzen oder man bedient sich der Analyseprogramme, um eigene Klänge als Ausgangspunkt für neue Bibliotheken zu verwenden.

AddAn ist ein Programm zur "Analyse additive" und **ResAn** ist ein Programm zur Analyse von Resonanzmodellen.

Da das Programmpaket sehr umfangreich ist und mehrere Synthesemodelle und deren Analysen beinhaltet, unterteile ich den Artikel in mehrere Blöcke.

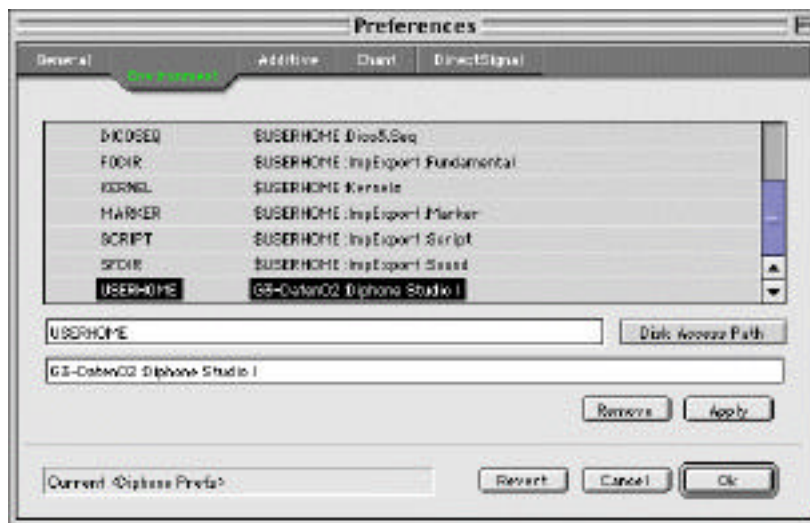
Der in dieser Ausgabe erscheinende erste Teil wird auf die Möglichkeiten der Additiven Analyse mit dem Programm AddAn, auf die Kreation eigener Bibliotheken und die Manipulation von additiven Segmenten in Diphone eingehen.

Spätere Teile werden die Chant-Synthese, resonante Filter und die Analyse resonanter Modelle behandeln. (Kommentare und eventuelle Wünsche bitte per mail).

Systemeinstellungen

Da die gesamte Dateistruktur von Diphone auf Unix basiert, ist es wichtig, in den Preferences des Programms festzulegen, wo sich Klänge, Analysefiles und andere Daten auf der Festplatte befinden. Innerhalb des Programmordners gibt es drei wichtige Unterordner: Container, Dico&Seq und ImpExport. Im Ordner Container werden die Bibliotheken gespeichert, in Dico&Seq befinden sich wichtige Files für Chant und man speichert dort ebenfalls die neu geschaffenen Sequenzen; in ImpExport gibt es weitere Unterordner, in denen sich die Ausgangsklänge befinden (Sound) oder in welche während der Analyse Files gespeichert werden (Fundamental und Partial)

Möchte man einen anderen Ort auf der Festplatte für diese Files nutzen (außerhalb des Programmordners), setzt man die Variable USERHOME in Preferences/Environnement auf den neuen Ordner.



AddAn — die additive Analyse

In Diphone werden nicht die Klänge selbst bearbeitet sondern Parameter, die die Klänge repräsentieren, wie in vielen anderen Analyse-Synthese Methoden auch.

Bei der additiven Analyse-Resynthese werden Klänge als eine Summe von Sinusschwingungen dargestellt, deren Parameter die Frequenzen, Amplituden und Phasen der Teiltöne (partials) sind. Während einer additiven Analyse werden also diese

Parameter ermittelt. Zusätzlich wird die Fundamentalfrequenz (`fundamental` oder F_0) berechnet, jene Frequenz, die in harmonischen Klängen der ersten Harmonischen entspricht und im zeitlichen Verlauf des Klanges variieren kann. Diese Syntheseart funktioniert am besten für harmonische Klänge, die einen geringen Geräuschanteil haben. Im weiteren Verlauf sehen wir aber auch Möglichkeiten, weniger harmonische Klänge zu analysieren.

Um einen Klang in Diphone synthetisieren zu können, greift man auf eine Bibliothek vorhandener Segmente zurück, die die Daten der Frequenzen, Amplituden und Phasen enthalten. Für die Schaffung einer solchen Bibliothek analysiert man einen vorhandenen Klang mit dem Programm **AddAn** und speichert die Daten der additiven Analyse in selbige. Der erste Schritt ist die Berechnung der Fundamentalfrequenz, danach werden die Frequenzen, Amplituden und Phasen der Teiltöne ermittelt und schließlich ist eine überprüfende Synthese dieser Daten möglich, um die Qualität der Analyse hören zu können.

Die drei Etappen speichern verschiedene Datenfiles auf die Festplatte: ausgehend von einem Klang "Beispiel.aiff" wird ein Datenfile für die Fundamentalfrequenz "Beispiel.F0" in den Ordner :ImpExport:Fundamental abgelegt, ein File mit den Daten der Teiltöne "Beispiel.ADD" in den Ordner :ImpExport:Partial und der resynthetisierte Klang "Beispiel.synth.AIFF" in den Ordner :ImpExport:Sound gespeichert.

Fourier-Analyse des Spektrums (FFT)

Sowohl während der Suche nach der Fundamentalfrequenz, als auch bei der Ermittlung der vorhandenen Teiltöne wird der Klang zuerst mittels einer Fourier-Analyse untersucht. Selbst wenn für einen Teil der Leser die folgende Beschreibung der FFT bekannt sein dürfte, gehe ich darauf doch etwas ausführlicher ein, da deren Verständnis eine wichtige Voraussetzung für gute Analyseergebnisse darstellt und sich immer wieder viele Fragen darauf beziehen. (Und vielleicht sind sie manchem Leser ja auch bei der Parameterauswahl in anderen Programmen wie z.B. AudioSculpt oder SoundHack hilfreich).

Ich beschreibe die FFT und deren Parameter `Windowsize`, `FFT-Size` und `Windowstep` aber nicht wissenschaftlich-mathematisch sondern eher musikerfreundlich-schematisch.

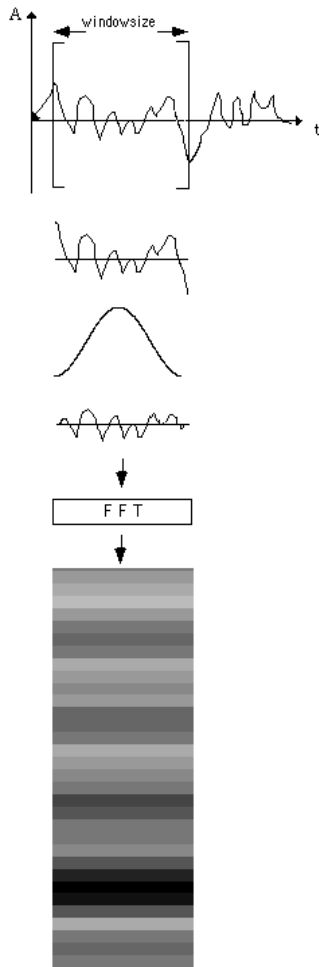
Da eine einzelne FFT-Analyse ein statisches Spektrum liefert, das keinen zeitlichen Verlauf beinhaltet, wäre die Analyse des Gesamtklages mit einer einzigen FFT nicht die Repräsentation des Klages selbst, sondern eine Art von "Gesamtdurchschnitt" der enthaltenen Energien.

Um die Spektrumsänderungen im zeitlichen Verlauf zu erhalten, wird der Klang in kleine Ausschnitte zerteilt, die jeweils unabhängig vom Rest des Klages analysiert werden und jeweils ein FFT-Spektrum (`short-time-spectrum`) ergeben. Die Gesamtanalyse des Klages ist dann eine Abfolge solcher "short-time-spectra". Die Länge eines solchen

zeitlichen Ausschnitts wird mit dem Parameter `window_size` festgelegt und entweder in Sekunden oder in Anzahl von Samples gemessen. Da die Berechnung der FFT dieses Klangausschnittes wie gesagt eine statische Abbildung der im Klangausschnitt enthaltenen Energien ist, würde eine im Ausschnitt enthaltene Melodie nicht als Abfolge von Tönen sondern als "Akkord" erscheinen.

Werte für eine günstige `window_size` hängen also vom zu analysierenden Klang ab : enthält er schnelle Variationen, wird man eine kleinere `window_size` wählen (z.B. 500 — 1000 Samples = 11.3 — 22.6 ms), damit nicht mehrere akustischen Ereignisse, die den Gesamtklang formen, in ein `window` fallen. Entwickelt sich der Klang langsamer, können größere Werte (z.B. 2000 - 4000 Samples) gewählt werden. Da die `window_size` vor Beginn der Analyse festgelegt wird und für alle folgenden "short-time-spectra" konstant bleibt, ist damit zu rechnen, daß deren Start- und Endpunkte nicht auf Nulldurchgänge des Klanges fallen und somit der gewählte Ausschnitt durch einen abrupten Beginn bzw. ein abruptes Ende einen Klick bekommt, der die Analyse verfälschen würde. (Bild 1.2, zweite Linie). Deshalb ist es nötig, den Klangausschnitt ein- und auszufaden. Dies geschieht durch Multiplikation mit einer sogenannten `windowfunction` (hamming, hanning, blackman, kaiser etc.)

Dieser ein- und ausgefadete Klangausschnitt wird nun mittels der FFT analysiert. Die FFT selbst repräsentiert einen Samplebuffer, dessen Größe (`FFT-size`) mindestens gleich groß wie der Wert der `window_size` und immer eine Potenz von 2 sein muß. (z.B. `window_size` = 1000, `FFT-size` = 1024 oder 2048 oder 4096 etc.). Der Buffer der `FFT-size` kann nicht kleiner als die `window_size` sein, sonst würde der ein- und ausgefadete Klangausschnitt ja wiederum beschnitten und hätte wieder Klicks. Die `FFT-size` kann aber größer als die `window_size` sein: in diesem Falle werden die restlichen Samples mit Nullen aufgefüllt (zero-padding).



Die FFT-size wird zwar in Anzahl von Samples gemessen, hat aber mit der zeitlichen Auflösung der Analyse nichts zu tun. (Dies ist immer wieder ein Punkt für Mißverständnisse.) Die FFT-size bestimmt die frequenzielle Auflösung der Analyse und somit seine Qualität der Abbildung der Frequenzen. Dies kann man sich folgendermaßen vorstellen: das gesamte Frequenzspektrum von 0 Hz bis zur Hälfte der Samplefrequenz (also 22050 Hz bei einer Samplefrequenz von 44100 Hz) wird in eine Anzahl gleich großer Frequenzbänder (frequency-bins) unterteilt. In jedem dieser Bänder wird die vorhandene Energie gesucht. Um eine qualitativ gute Analyse zu erreichen, möchte man möglichst viele, schmale Bänder, damit die Energien einzelner Teiltöne in unterschiedliche Bänder fallen und somit voneinander unterschieden werden können.

Die Anzahl dieser Bänder (und somit ihre Bandbreite) hängt von der FFT-size ab.

Mit einer FFT-size von 1024 werden 512 Bänder zwischen 0 Hz und 22050 Hz berechnet, die Bandbreite jedes Bandes ist also $22050 / 512 = 43.07$ Hz. Mit einer FFT-size von 4096 werden 2048 Bänder mit jeweils 10.77 Hz Bandbreite berechnet. Eine größere FFT-size ergibt also eine größere Anzahl schmalere Bänder und somit eine bessere Auflösung der analysierten Frequenzen. Fallen nämlich mehrere Teiltöne in ein und dasselbe Band, werden sie "gemittelt" mit allen Nachteilen wie beating oder Auslöschung.

Eine "short-time-analysis" enthält also ein vorbestimmtes

Raster von Frequenzbändern für die die aktuellen Amplituden und Phasen gesucht werden.

Zur Veranschaulichung der Resynthese kann man sich vorstellen, daß jedes dieser Frequenzbänder durch einen Sinusgenerator wiedergegeben wird, dessen Frequenz auf die Zentralfrequenz des Bandes gestimmt ist und durch die aktuelle Phase des Bandes "feingestimmt" wird. Die Amplitude des Generators folgt der analysierten Amplitude des Bandes. Diese Daten werden von "short-time-analysis" zu "short-time-analysis" interpoliert und ergeben damit wieder ein kontinuierliches Signal.

Als eine "Eselsbrücke" kann man sich also merken, daß für die zeitliche Auflösung der Analyse die `window size` zuständig ist (je schneller das Signal sich verändert, um so kleinere Fenster sollten verwendet werden) und für die Qualität der frequentiellen Auflösung die FFT-size zuständig ist.

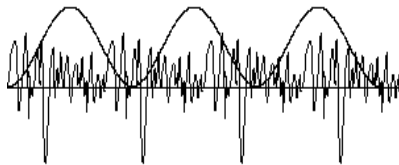
Die `window size` bestimmt aber noch einen anderen Faktor: die tiefste analysierbare Frequenz. Selbige muß nämlich mindestens drei Perioden innerhalb der `window size` haben, um analysiert werden zu können. Dies ist keine ganz strikte Grenze - besser sind sogar vier bis fünf Perioden, um eine Frequenz sicher bestimmen zu können.

Variiert der zu analysierende Klang zwischen 200 und 500 Hz, ist die längste Periode $1/200$ s = 0.005 s. Um 4 Perioden im Analysefenster zu haben, sollte die `window size` also mindestens $0.005 * 4 = 0.02$ Sekunden lang sein.

Man sieht hier eine Schwierigkeit für schnell variierende Klänge mit tiefen Frequenzanteilen. Um die zeitliche Variation des Klanges zu analysieren, möchte man eine kleine `window size` wählen, um die tiefen Frequenzen analysieren zu können, braucht man jedoch eine größere `window size`. In diesen Fällen kann man nur durch Ausprobieren den besten Kompromiß finden.

In Programmen, wo `window size` und `FFT-size` nicht einzeln wählbar sind (z.B. SoundHack, Sounddesigner, Protools Audiosuite), steht man immer vor einem Kompromiß. Mit kleinen Werten erreicht man eine gute zeitliche Auflösung, erlangt dadurch aber sehr breite Frequenzbänder der FFT und damit eine schlechte Resynthese der Frequenzen. Mit großen Werten erhält man eine gute frequenzielle Auflösung, büßt damit aber in der zeitlichen Auflösung ein (viele Klangereignisse fallen in ein und dasselbe Analysefenster und ihre Energien werden gemittelt).

Der nächste Parameter ist der `window step` oder auch `step size`.

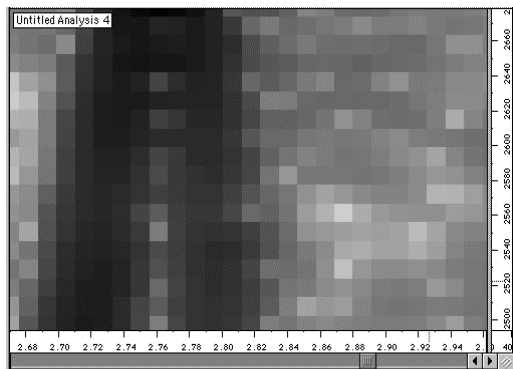


Ausfaden beeinflusst wird.

Würde eine Analyse (die Länge einer `window size`) der vorhergehenden direkt folgen, könnten Teile des Klanges durch das Ein- und Ausfaden nicht analysiert werden.

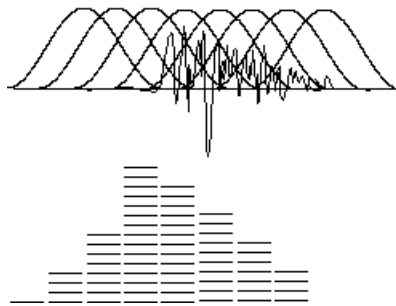
Deshalb beginnt die folgende Analyse nach einer zeitlichen Verschiebung, die kleiner als die `window size` ist und die Analysewindows überlappen sich. Typische Werte für die `step size`

sind $1/4$ oder $1/8$ der `window size`. Jedes Analysewindow liefert ein `short-time-spectrum`, wovon zeitlich nur die Dauer der `step size` verwendet wird, dann folgt schon die nächste Analyse. Dies hat den Vorteil, daß jeder Zeitpunkt des Klanges ein mal mehr oder weniger im Mittelbereich der `window function` erscheint und seine Energie nicht maßgeblich durch das Ein- und

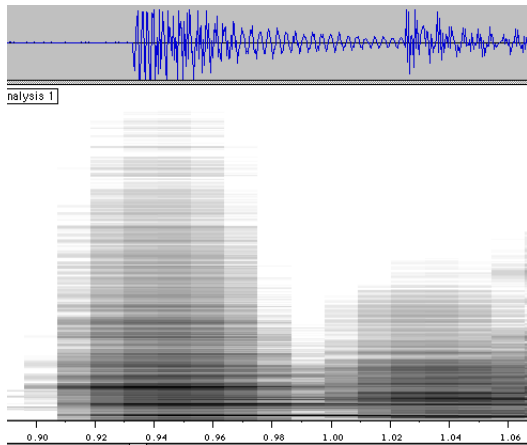


Zoomen wir in das Sonogramm in Audiosculpt, sieht man kleine Kästchen, die in der zeitlichen Aufeinanderfolge die `short-time-spectre` zeigen, deren Länge der `step size` entspricht. Auf der Frequenzachse entsprechen die Kästchen den `frequency-bins` (hier ca. 10 Hz breit, da eine `FFT-size` von 4096 verwendet wurde).

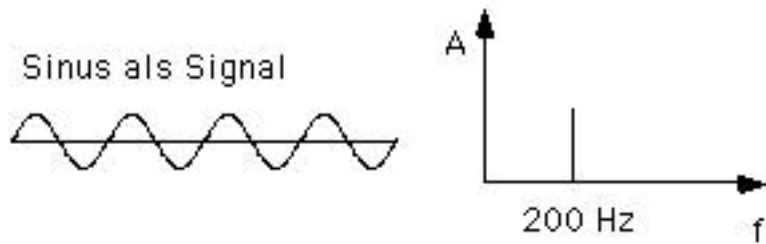
Der Nachteil der überlappenden Analysefenster liegt in der Tatsache, daß jeder Zeitpunkt im Klang mehrfach analysiert wird (z.B. vier mal bei einer `stepsize` von $1/4$ der `window`size). Betrachten wir diesen Effekt am Beispiel der Analyse eines Attacks (Klangereignis mit viel Energie im gesamten Spektrum).



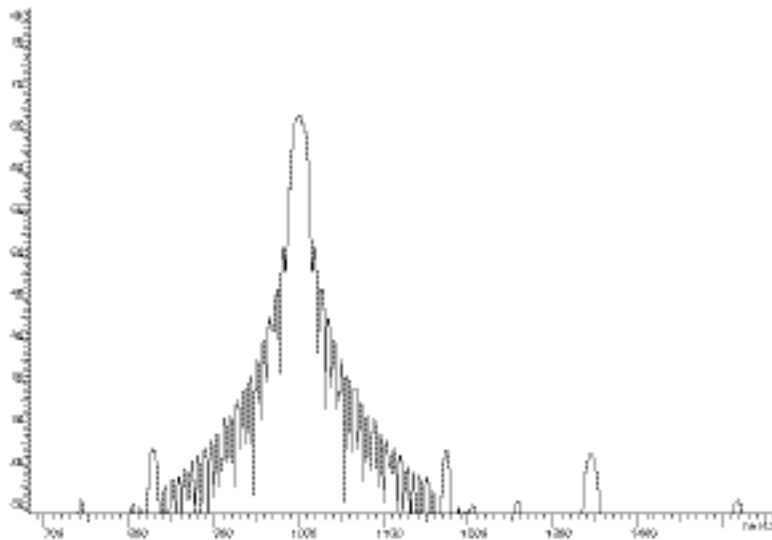
Das erste Analysefenster "berührt" den Attack nur wenig und durch das Ausfaden bleibt kaum Energie übrig: das resultierende Spektrum hat kaum Energie. Die darauffolgenden Analysefenster werden mehr und mehr den Attack in ihrem Zentrum haben und demzufolge mehr und mehr hohe Frequenzen finden. (Im linken Beispiel hat das vierte Fenster den Attack im Zentrum). Im weiteren Verlauf wird die Energie der hohen Frequenzen wieder abnehmen. Da das zum jeweiligen Analysefenster gehörende `short-time-spektrum` zeitlich am Anfang des Fensters liegt, wird darin schon Energie abgebildet, die eigentlich erst später erscheint. Dadurch ergibt sich der Effekt einer Art von "spektralem fade-in" der hohen Frequenzen.



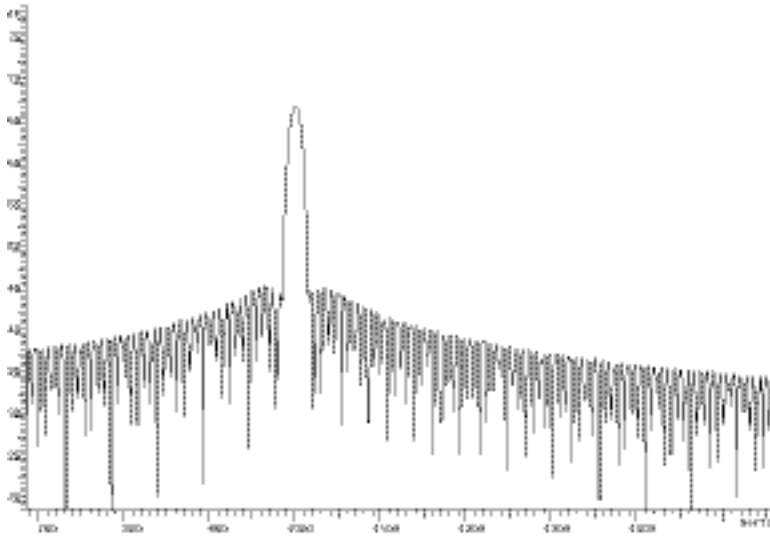
Der vierte zu wählende Parameter für die Analyse ist die `windowfunction`. Verschiedene Programme bieten eine Auswahl von Fensterfunktionen an, wobei `hamming`, `hanning`, `blackman` und `kaiser` häufig vertreten sind. Ihr Einfluß auf die Qualität der Analyse ist subtil aber nicht zu unterschätzen. Theoretisch ist die spektrale Abbildung einer Sinusschwingung eine Linie im Spektrum.



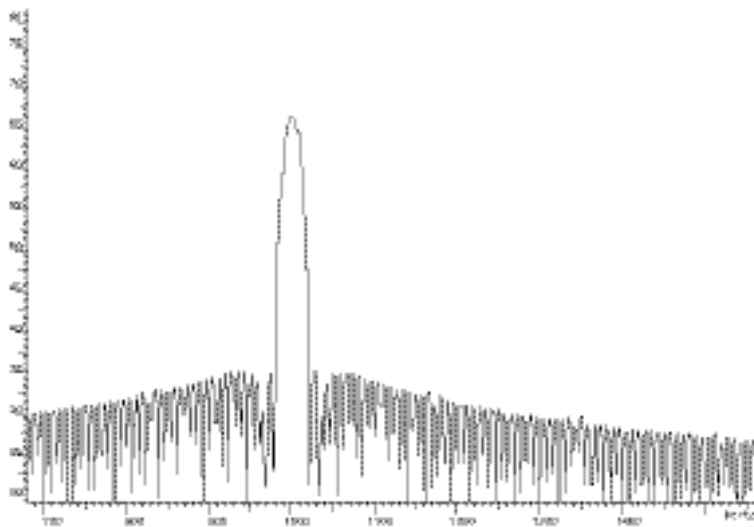
Wird dieses Signal nun während der Analyse mit einer Windowfunktion multipliziert, kommt es zur Multiplikation zweier Signale (ähnlich wie bei einer Ringmodulation). Das resultierende Spektrum ergibt sich aus einer Convolution des Signalspektrums mit dem Spektrum der Windowfunktion. Die qualitativen Unterschiede der einzelnen Windowfunktionen liegt also darin, wie ihr Spektrum beschaffen ist. Um dies zu veranschaulichen wurde ein Sinus mit einer Frequenz von 1000 Hz mit den drei typischsten Windowfunktionen multipliziert. Die Energie des Sinus wird dabei also nicht als eine einfache Linie abgebildet.



Ein hanning-window erzeugt einen ziemlich breiten zentralen peak und einige Seitenbänder

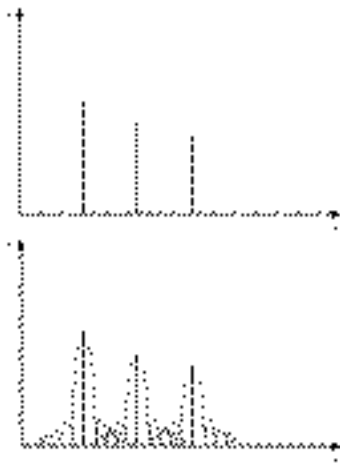


Die Frequenzdefinition des zentralen peaks vom hamming-fenster ist besser. Es erzeugt aber wesentlich weiter auslaufende Seitenbänder.



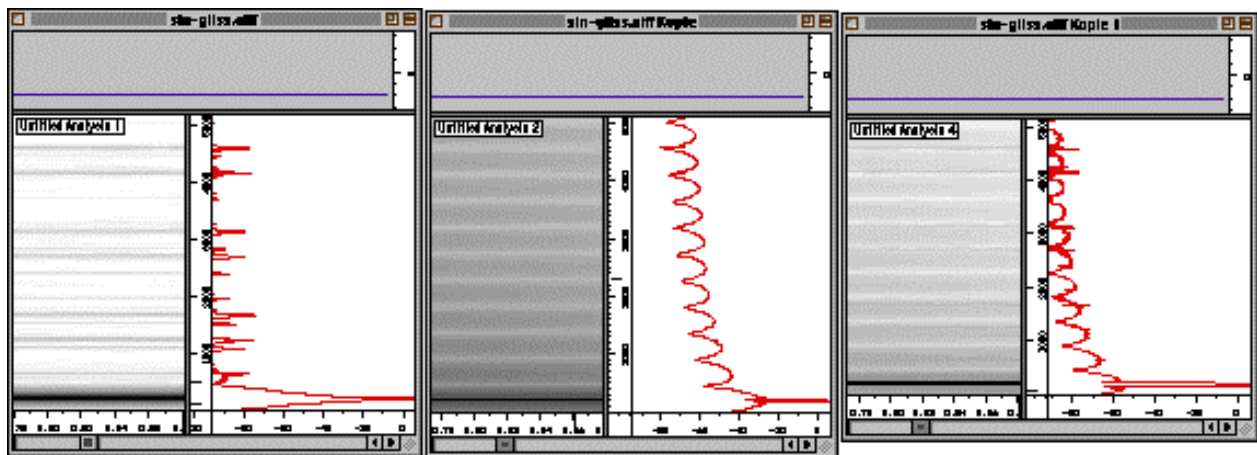
Das Spektrum des blackman-fensters hat ebenfalls eine gute Definition der Frequenz. Die Seitenbänder haben eine geringere Amplitude und sind im Frequenzbereich nicht so ausladend wie jene des hamming-fensters.

Dies war nun lediglich die Multiplikation einer einzelnen Sinusfrequenz mit den drei gängigsten window-funktionen. Bei der Analyse eines komplexen Klages geschieht diese Energieverteilung nun für jeden einzelnen Teilton.



Wenn der komplexe Klang drei Teiltöne enthält, wäre sein theoretisches Spektrum lediglich eine Folge von drei Linien. Da bei der Analyse diese drei Frequenzen aber mit dem Spektrum der Windowfunktion multipliziert werden, ergibt sich folgende Energieverteilung.

Dies sieht man auch im Sonagramm. Ein Sinus wurde mit den window-funktionen hanning, hamming und blackman analysiert (von links nach rechts):

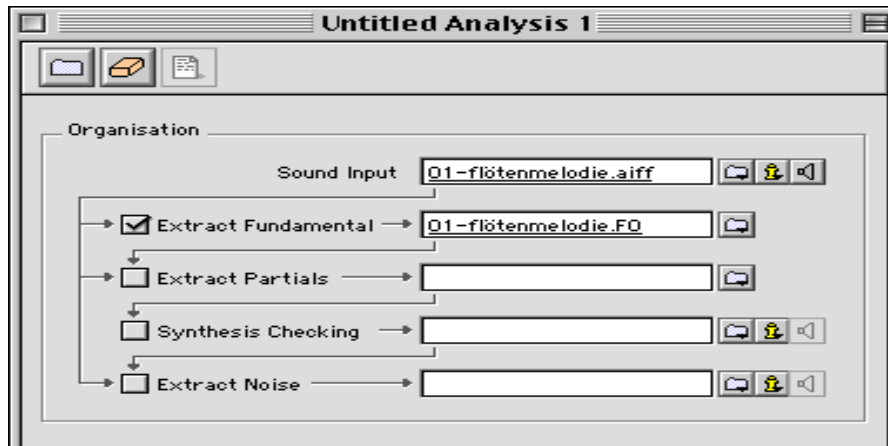


Wobei das hanning-fenster klar sichtbare "Teiltöne" produziert, ergeben die anderen beiden Funktionen eher einen lautereren (hamming) und leiseren (blackman) "Geräuschteppig". Der Signal-Rausch-Abstand zwischen zentralem peak und Seitenbändern ist aber groß genug, daß sie nicht hörbar werden. Lediglich bei aufeinanderfolgenden Bearbeitungen, die jeweils eine Analyse/Resynthese durchlaufen, können die Nebeneffekte der Fensterfunktionen störend werden.

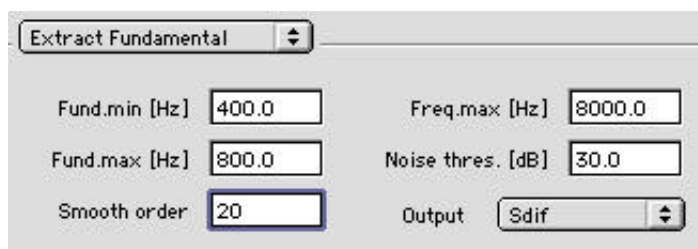
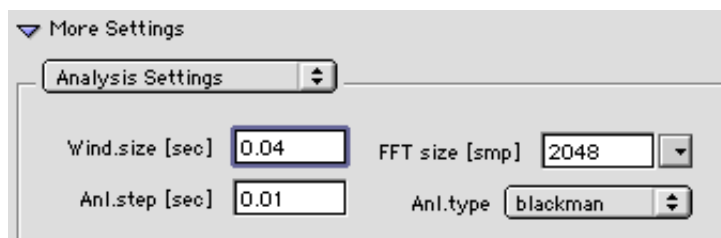
Analyse der Fundamentalfrequenz (fundamental)

Nach diesem Abriß über die Auswirkungen der vier Parameter kommen wir zurück zur Analyse in Diphone. Zuerst wird die Fundamentalfrequenz (F0) berechnet.

Befindet sich der zu analysierende Klang im Ordner : ImpExport : Sound, kann man ihn in **AddAn** durch klicken auf das Ordnersymbol in der Zeile "Sound Input" auswählen und durch klicken auf den Lautsprecher anhören.
(Beispiel 01-flötenmelodie)



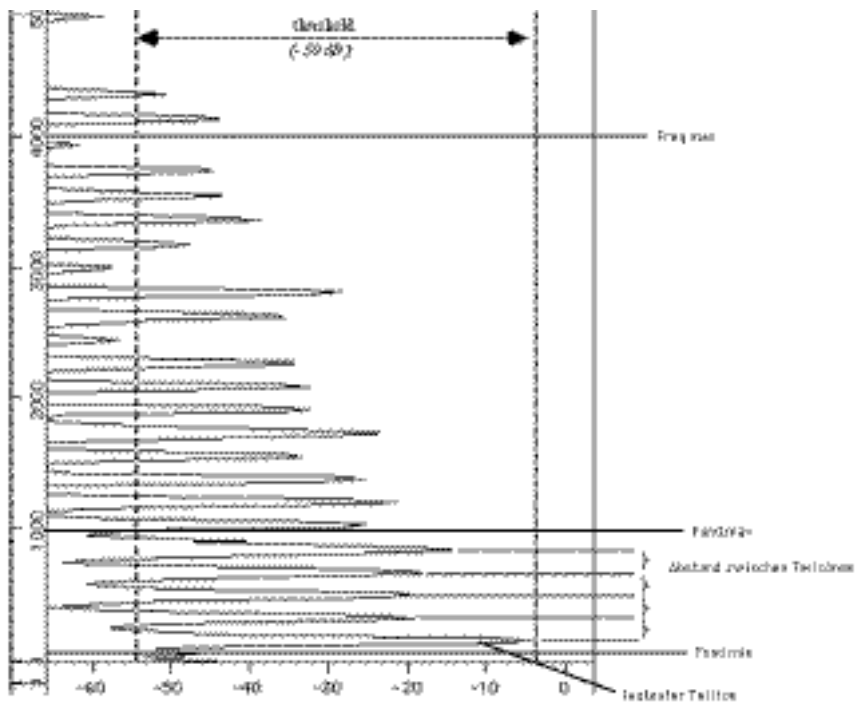
Zuerst wählen wir lediglich "Extract Fundamental".
Im unteren Bereich des Fensters (More Settings) werden die Werte für die Analyse eingestellt. `window_size` und `window_step` (hier Analysestep) werden in Diphone in Sekunden angegeben.



Im Folgenden werden die Werte für die Analyse der Fundamentalfrequenz eingestellt. Während dieser Analyse werden die Abstände der gefundenen Energiebänder ermittelt und untersucht, ob sie sich auf ein kleinstes gemeinsames Vielfaches beziehen. Um gute Ergebnisse zu erhalten und auch um den Rechenaufwand zu limitieren, begrenzt man den Suchbereich (ansonsten würden alle "frequency-bins" innerhalb einer Analyse miteinander verglichen)

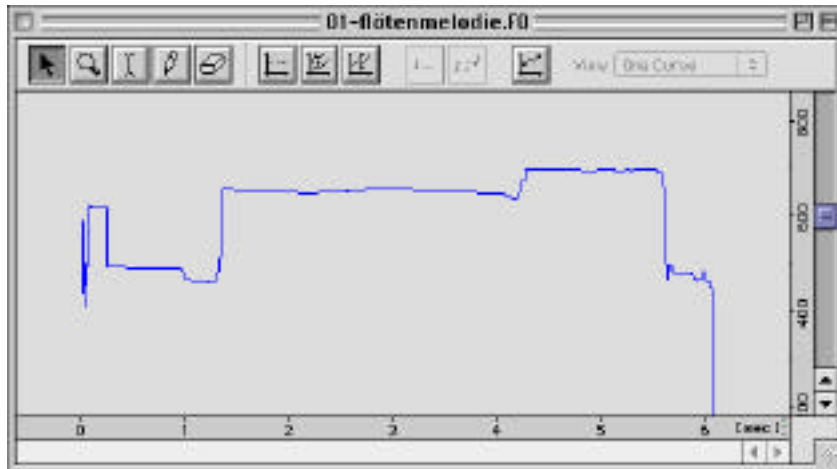
Fund.min und Fund.max bestimmen Grenzwerte, zwischen denen nach der F0 gesucht werden soll. Noise threshold ist ein relativer Amplitudenabstand zum lautesten Teilton. Alle Energien, die leiser als dieser Schwellwert sind, werden während der Suche nach harmonischen Vielfachen vernachlässigt. Bei Klängen mit großem Geräuschanteil wird man also einen geringeren (!) Noise.thresh. wählen, um nur die lautesten Energien in die Analyse einzubeziehen.

Freq.max bestimmt die obere Grenze des Suchbereichs. Hat man 4 bis 5 Vielfache einer Grundfrequenz gefunden, kann man davon ausgehen, daß die richtige F0 gefunden wurde. Es ist also nicht nötig, bis ans Ende des Spektrums (22050 Hz) weiterzusuchen. Smooth order (Werte zwischen 3 und 30) glättet nach der Analyse den Verlauf der Fundamentalfrequenz, um kleine Schwankungen zu nivellieren.

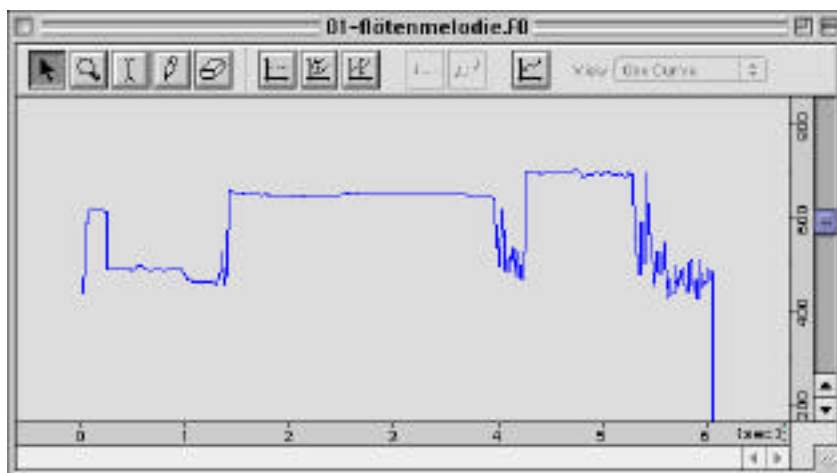


Das output format ist Sdif, ein binäres file. Um die Daten ansehen zu können, oder auch in anderen Programmen weiterzuverwendetn, ist ebenfalls die Möglichkeit der Ausgabe als ASCII-file möglich.

Selbiges File kann dann z.B. in Diphone geöffnet und betrachtet werden. Für unsere Flötenmelodie ergibt sich folgendes Ergebnis mit den oben erwähnten Einstellungen:



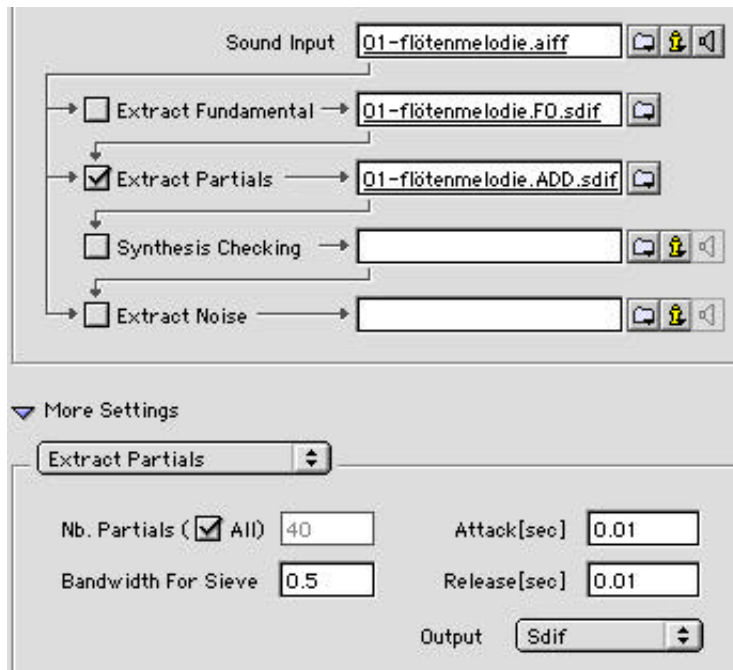
Wir sehen Unregelmäßigkeiten zu Beginn und am Ende des Klanges, diese fallen aber nicht weiter ins Gewicht. Hat man schlechte Parameter für die Analyse der F0 gewählt, sieht man dies an unregelmäßigen Sprüngen des Frequenzverlaufs:



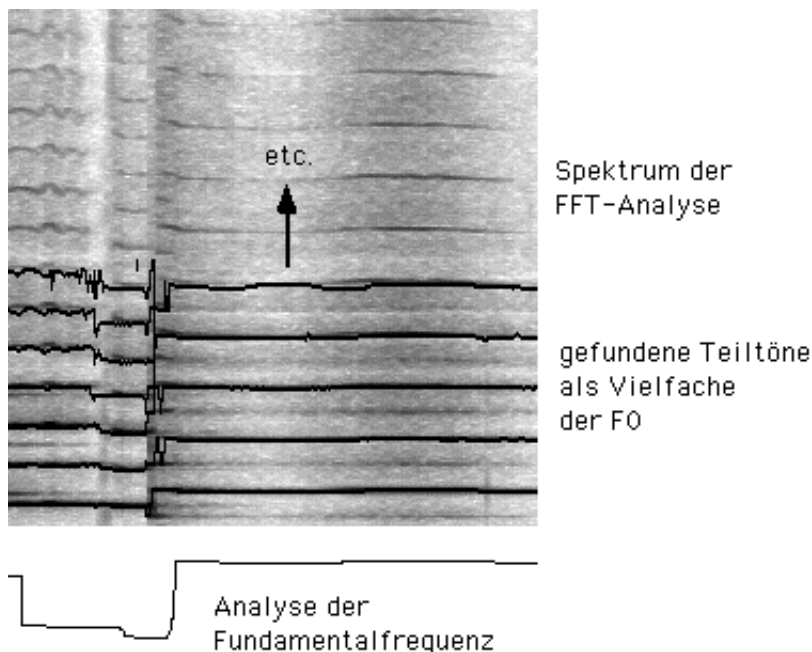
Um im weiteren Verlauf die Daten der F0 verwenden zu können, muß das output-format unbedingt auf Sdif eingestellt sein. (ASCII wurde hier also nur zur visuellen Überprüfung verwendet)

Reduzierung auf harmonische Bestandteile des Klanges (partials)

Der nächste Schritt ist die Analyse der Teiltöne.



Dieser Analyseschritt wird die berechnete Fundamentalfrequenz als Basis nehmen, um in den "short-time-spectra" der FFT Energieverläufe zu extrahieren, die Vielfache der Fundamentalfrequenz sind, also den Teiltönen des Klanges entsprechen.

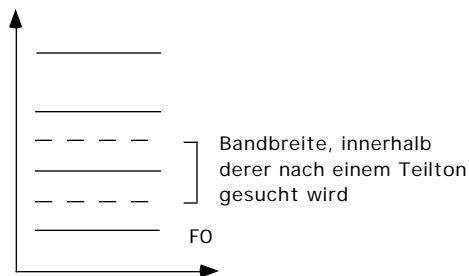


An diesem Punkt sieht man die Bedeutung einer guten Analyse der Fundamentalfrequenz. Würde sie unnatürliche Sprünge aufweisen, würden auch die Frequenzen der Teiltöne diese Fehler enthalten.

An der obigen Analyse sieht man außerdem, daß der Geräuschanteil des Klanges, der bei unserem Beispiel durch die Luft in der Flöte verursacht wird, nicht analysiert werden kann. (weiter unten werde ich ein Beispiel geben, wie auch der Geräuschanteil besser wiedergegeben werden kann).

Die Parameter für Attack und Release ermöglichen das Ein- und Ausblenden von Teiltönen in dem Moment, wo sie im Spektrum auftauchen bzw. aufhören. Dadurch verhindert man Klicks.

Bandwith for Sieve bestimmt einen Frequenzbereich für jeden Teilton, der sich um das exakte Vielfache der Fundamentalfrequenz befindet und somit Abweichungen vom strikten harmonischen Gefüge eines Klanges erlaubt. Ein Wert von 0.5 reicht vom Mittelwert zwischen zwei Teiltönen bis zum Mittelwert zwischen den folgenden beiden Teiltönen.



Resynthese und Extract Noise

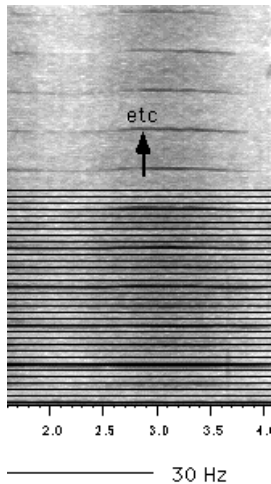
Synthesis Checking ist die Resynthese der gerade analysierten Teiltöne und gibt somit die Möglichkeit, die Qualität der Analyse zu beurteilen, bevor man anfängt, die Segmentierung der Daten vorzunehmen und Bibliotheken (container) zu berechnen. (Beispielklang 02-floetenmelodie.synth).

Hat man "Extract Noise" ausgewählt, wird ein Klang "Beispiel.noise.AIFF" erzeugt, der die Subtraktion des Ausgangsklages minus des resynthetisierten Klages ist. Damit erhält man alle Klanganteile, die nicht ins harmonische Raster fielen. (Beispielklang 03-floetenmelodie.noise).

Selbstverständlich kann man auch alle Analyseschritte in einem Durchlauf berechnen lassen. Zuerst nur die Fundamentalfrequenz zu berechnen und ihren Verlauf anzusehen hat den Vorteil, daß man eine eventuell schlechte F_0 -Analyse schon erkennt, bevor man die weiteren Analyseschritte berechnet.

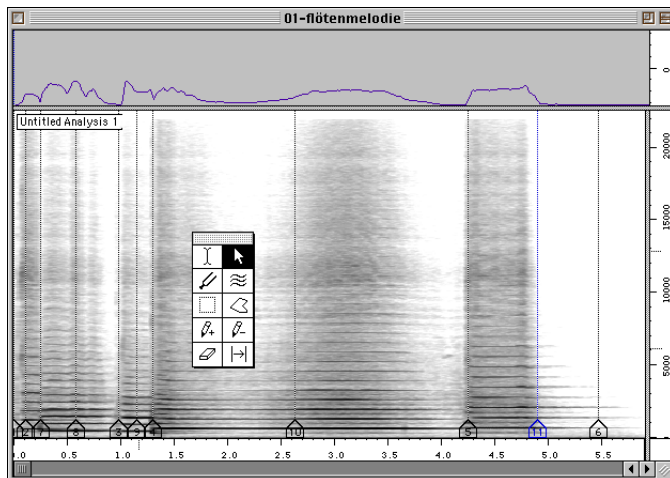
Verbesserung der Analyse vom Geräuschanteil

Wie weiter oben erwähnt, war die Analyse des Geräuschanteils der Flöte durch die Reduzierung auf die pur harmonischen Klanganteile nicht möglich. Gibt man bei der Analyse eine "künstliche" Fundamentalfrequenz von z.B. 30 Hz, werden während der Analyse der Teiltöne alle Vielfachen von 30 Hz ermittelt und somit während der resynthese eine viel größere Anzahl von Sinustönen verwendet. Da sich diese nun nicht ausschließlich auf den Vielfachen der reellen Fundamentalfrequenz befinden, sondern auch zwischen den "echten" Teiltönen, werden Energien, die dem Geräuschanteil des Klanges zuzuordnen sind trotzdem wiedergegeben. (Beispielklang: 04-foetenmelodie.synth.30Hz)

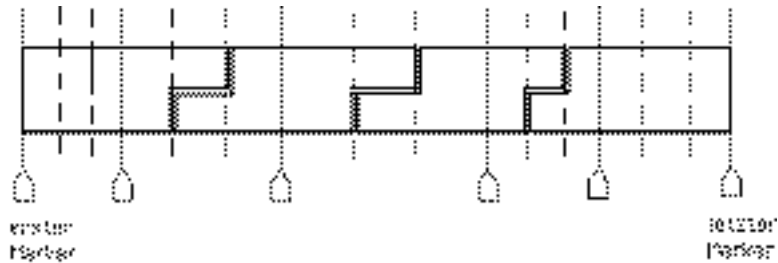


Segmentierung des Ausgangsklanges (segments)

Zur Segmentierung kann man sich des Programms Audiosculpt bedienen.



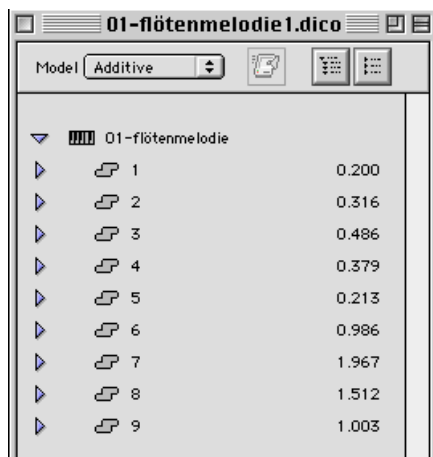
Durch ctrl-klick mit dem Pfeil ins Sonagramm kann man Marker setzen, deren Zeitpunkte über `File:Export:Markers` in ein Textfile geschrieben werden (in den Diphone-Ordner `:ImpExport:Marker`). Jeder Marker steht für die Mitte eines Segmentes.



Berechnung der Bibliothek, die die Segmente enthält

Ausgehend von diesem Marker-file kann man in Diphone ein Script erzeugen (Menü: `PlugIns>Create Script from Markers`) und danach den container berechnen (Menü: `PlugIns>Create Dictionary from Script`)

Nun befindet sich im Ordner `container` eine Bibliothek, deren Segmente die Analysedaten enthalten und in neuen Sequenzen verwendet werden können.



Das Bild zeigt einen Screenshot der Diphone-Software. Die Titelleiste zeigt den Dateinamen `01-flötenmelodie1.dico`. Darunter befindet sich ein Menü für das Modell, das auf `Additive` eingestellt ist. Die Hauptansicht zeigt eine Liste von Segmenten für die Datei `01-flötenmelodie`. Jedes Segment ist durch ein Dreieck und eine Uhrzeit markiert.

Segment	Zeitpunkt
1	0.200
2	0.316
3	0.486
4	0.379
5	0.213
6	0.986
7	1.967
8	1.512
9	1.003

Öffnet man diesen Container in Diphone kann man die Segmente in eine neue Sequenz herüberziehen und bearbeiten.

Die Möglichkeiten der Klanggestaltung mit diesen Segmenten wird im folgenden Teil des Artikels beschrieben.