4 bach: automated composer's helper

4.1 bach

bach: automated composer's helper[1] in der Folge kurz: *bach* ist eine Sammlung von *Patches* und *Externals* für MAX. Obwohl *bach* „nur“ als eines von unzähligen *Libraries* für MAX in den Weiten des Internets zu finden ist, stellt es ganz besondere Möglichkeiten zur Verfügung, denen ich hier ein eigenes Kapitel widmen möchte. Die Entdeckung von *bach* hat mich bewogen, über die Thematik der *Notation in der computergestützten algorithmischen Komposition* zu schreiben.

bach ist „the missing Link“ in MAX und bringt die ansonsten abstrakte Darstellung von musikalischen Informationen in Zahlen oder *Listen* in ein Notenbild. Für dieses Vorhaben sind viele neue Objects notwendig und in bewundernswerter Weise von Andrea Agostini[2] und Daniele Ghisi[3] auch dafür entwickelt worden.

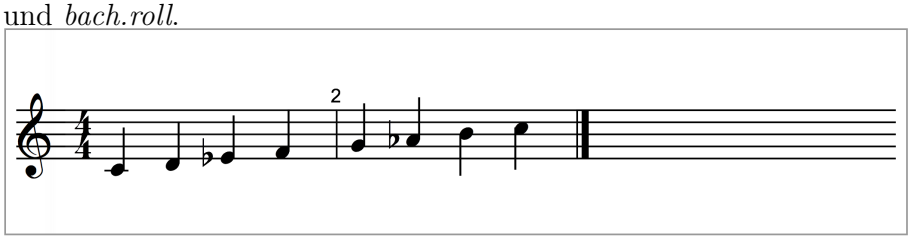
Viele der Errungenschaften aus *bach* sind dennoch nicht unbekannt. Seit Jahren dient OpenMusic (OM)[4] von Ircam als Vorbild dafür, wie Zahlenlisten

[1] http://www.bachproject.net (Jänner 2015).
[2] Andrea Agostini ist Komponist und lebt in Italien und Frankreich.
[3] Daniele Ghisi ist Mathematiker und Komponist aus Italien.
http://www.danieleghisi.com/biography/ (Jänner 2015).
Er erhielt 2012 für *bach* die *AFIM-Jeune Chercheur*- und *A. Piccialli*-Preise.
[4] http://repmus.ircam.fr/openmusic/home (Jänner 2015).

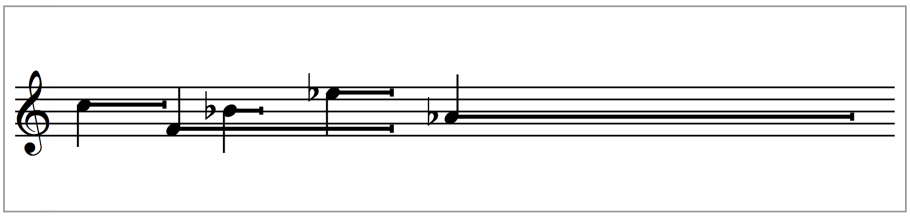
ansprechend als Notation repräsentiert werden.
Das entscheidend Neue mit *bach* sind die Möglichkeiten in *Echtzeit* innerhalb der MAX Programmierumgebung.
Es soll nicht verschwiegen werden, dass eine verwandte *Library* für MAX nämlich MAXSCORE[1] zeitlich schon vor *bach* im „Rennen" war.
MAXSCORE wurde von NICK DIDKOVSKY[2] in der *Java Music Specification Language (JMSL)*[3] gemeinsam mit GEORG HAJDU[4] entwickelt.

Zumindest für diese Arbeit fällt die Entscheidung für *bach* aus. *bach* ist Meister im Umgang mit *Listen* und liefert eine mehr als komplett wirkende Sammlung von überaus nützlichen *Externals* und *Abstractions*, welche auch für alle anderen Bereiche im Umgang mit MAX nützlich sind, mit.

Wesentlich für die Darstellung der Notation sind die OBJECTS *bach.score* und *bach.roll*.



bach.score liefert die erwartete Notation am Bildschirm.



bach.roll präsentiert das Darzustellende in einer *proportionalen*[5] Notation.

[1] http://www.computermusicnotation.com (Jänner 2015).
[2] NICK DIDKOVSKY ist Gitarrist, Komponist und Programmierer aus Amerika.
http://www.didkovsky.com (Jänner 2015).
[3] http://www.algomusic.com (Jänner 2015).
[4] GEORG HAJDU ist Professor für Multimedia-Komposition an der *Hochschule für Musik und Theater* in Hamburg.
http://georghajdu.de (Jänner 2015).
[5] Bei der *proportionalen Notation* handelt es sich um eine horizontale Platzverteilung, die jeder Note genau den ihrer Dauer entsprechenden Platz zuweist.

Zwei neue Datentypen mussten eingeführt werden. Zum einen die *Rationale Zahl* mit dem OBJECT *bach.ratnum* und zum anderen die *llll*s.
Rationale Zahlen sind unumgänglich, wenn es sich um *relative* Notenwerte handelt, und sehr nützlich, wenn Verhältnisse oder proportionale Berechnungen zu bewältigen sind.

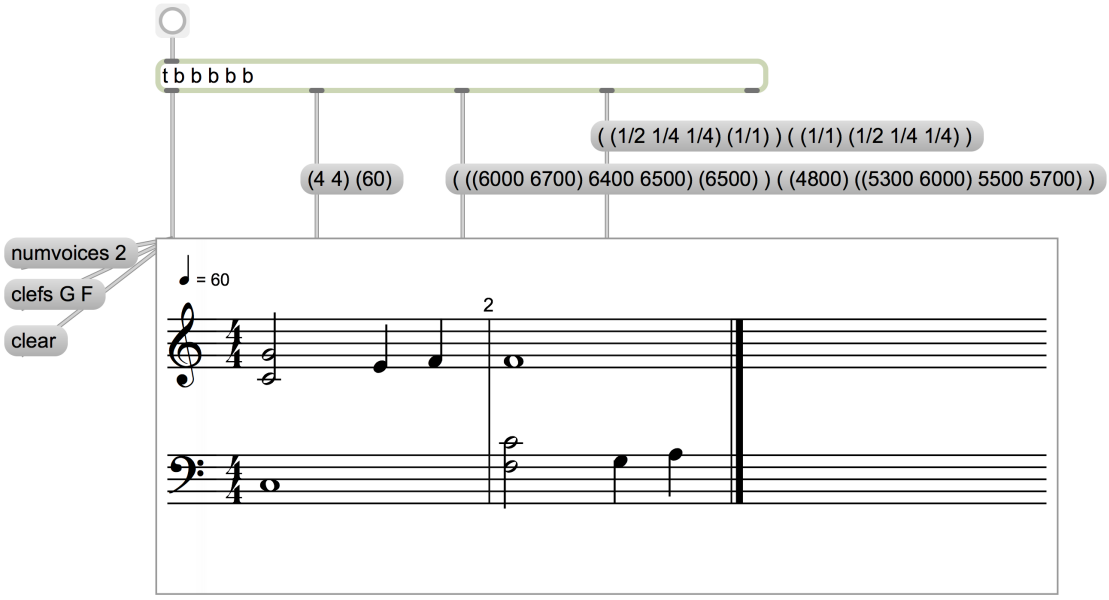
1/2 1/4 1/8 1/16 oder 16/15

bach.ratnum

((6700 (6000 7200) 6500))

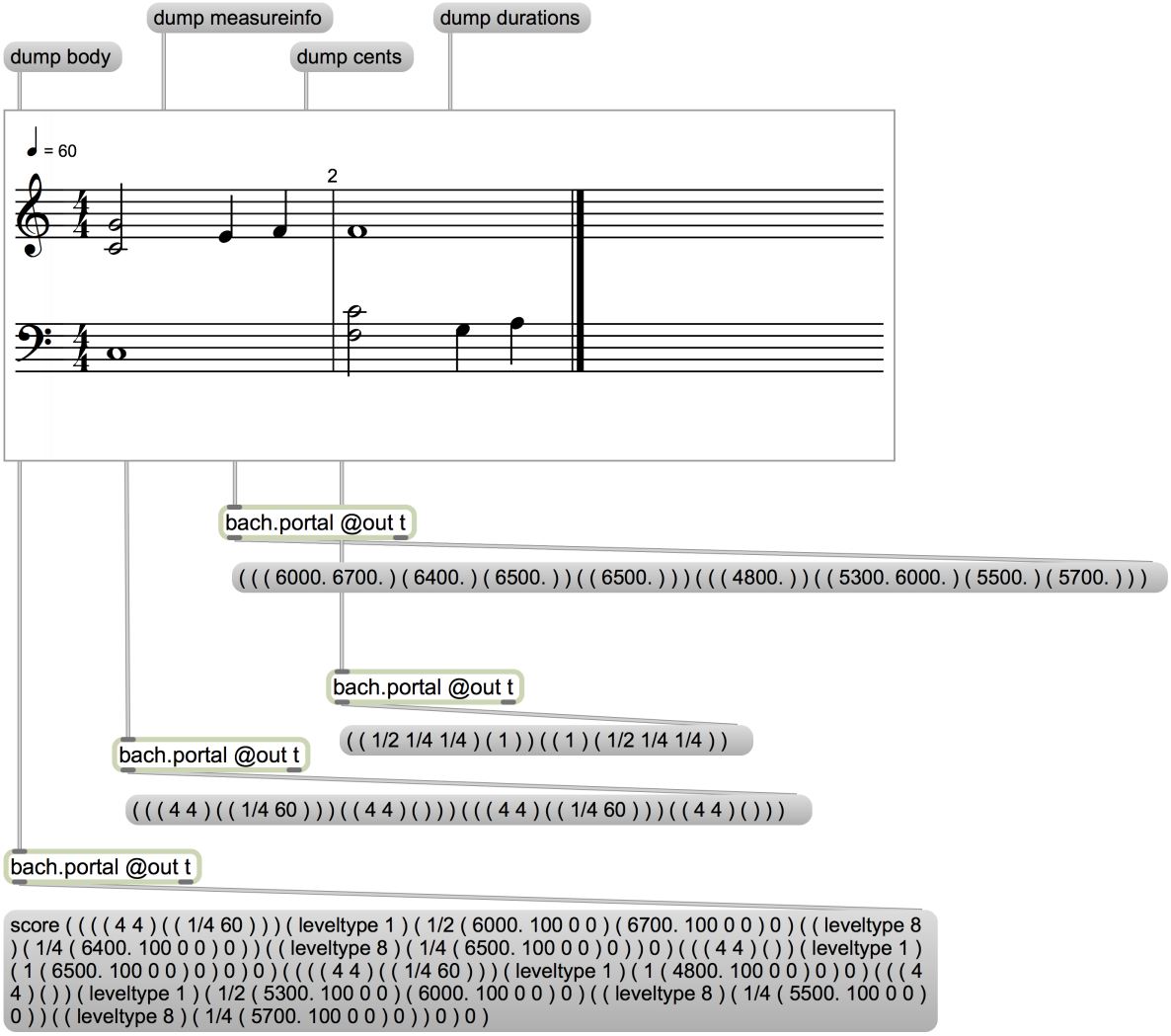
*llll*s sind *Lisp-like linked lists* und für das Kombinieren und Verschachteln von *Listen* – wie wir es schon von LISP kennen – zuständig.

Diese neuen Datenformate in MAX bringen die komplexe *Listen*-Berechnung auf einen Höhepunkt. Ganze Partituren werden damit als verschachtelte *Listen* repräsentiert.

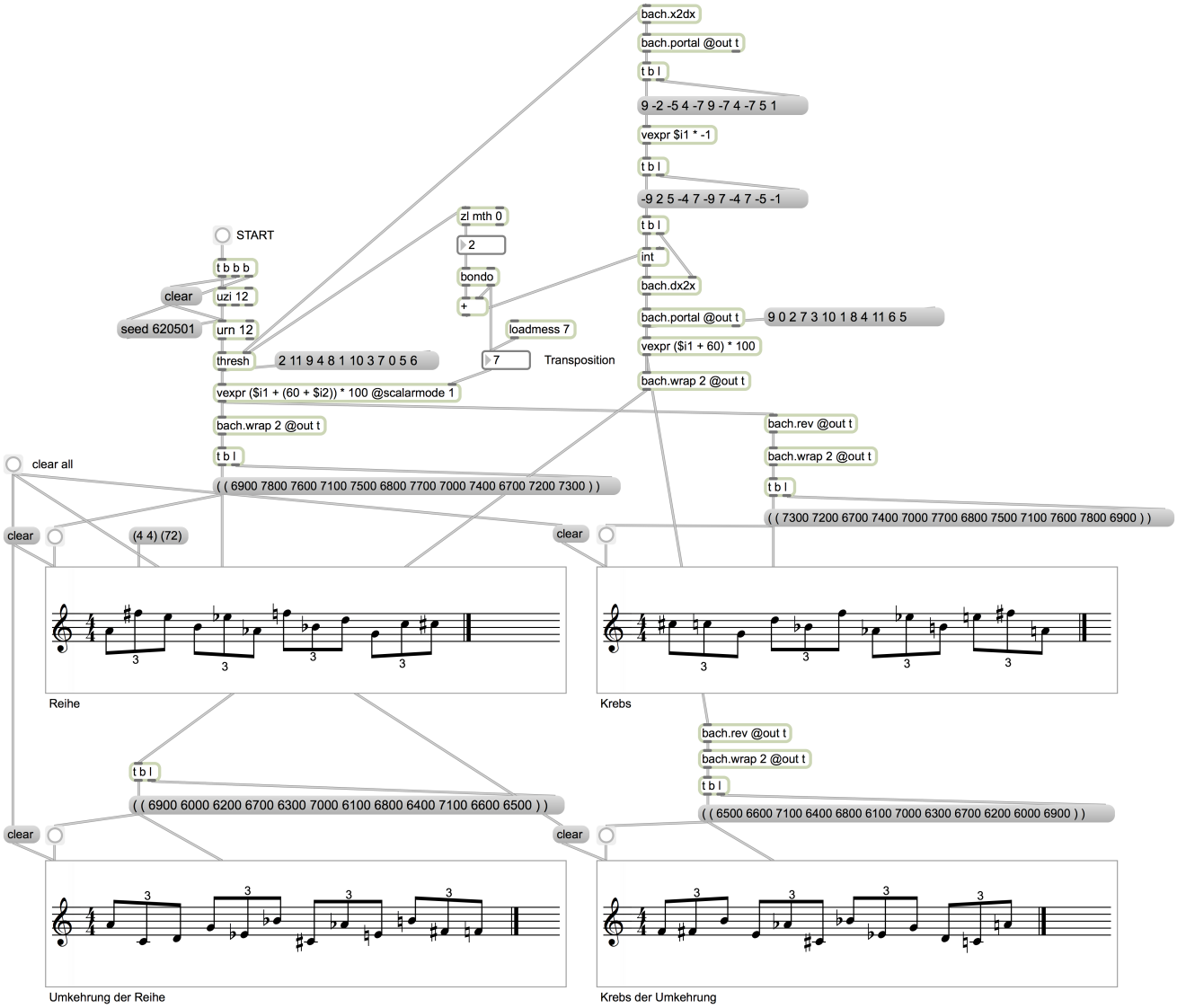


Hier wird auch deutlich, warum die *Listen*-Operationen und das „Jonglieren“ mit ihnen im Umgang mit *bach* innerhalb MAX so wesentlich sind.

Inhalte aus *bach.score* können mit *dump-messages* bei den zuständigen *outlets* ausgelesen und gegebenenfalls weiter verarbeitet werden.

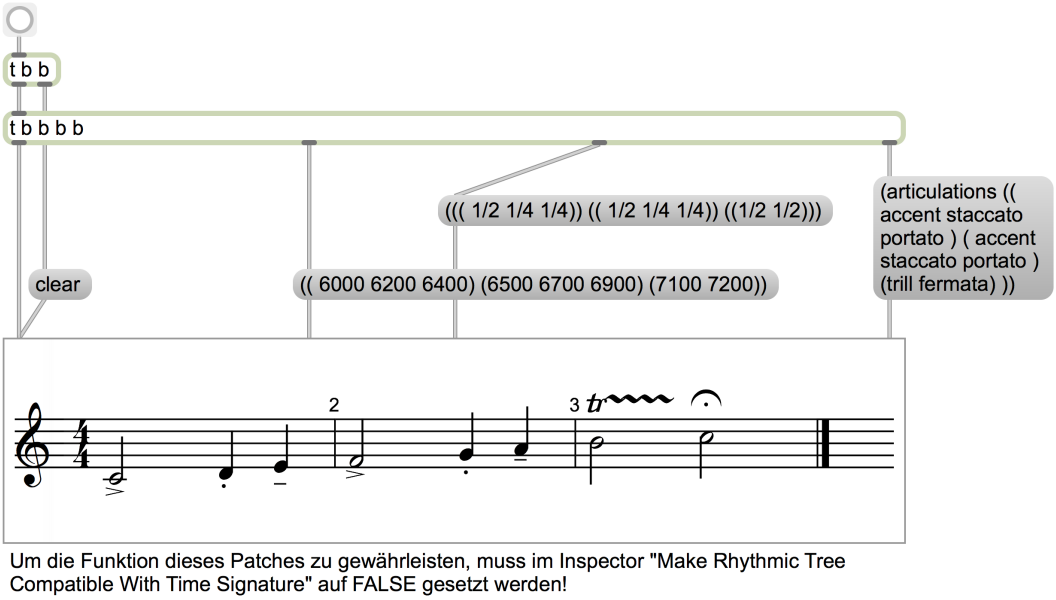


Um Berechnungen durchführbar und überschaubar zu halten, sind spezielle *bach*-Objects mit ihren genau definierten *Messages* großteils notwendig oder zumindest sehr nützlich.



In diesem Beispiel wird eine *Zwölftonreihe* von der *Generierung* bis zu den üblichen *Transformationen* wie *Transposition, Krebs* und *Umkehrung* sowie *Krebs der Umkehrung* gezeigt.

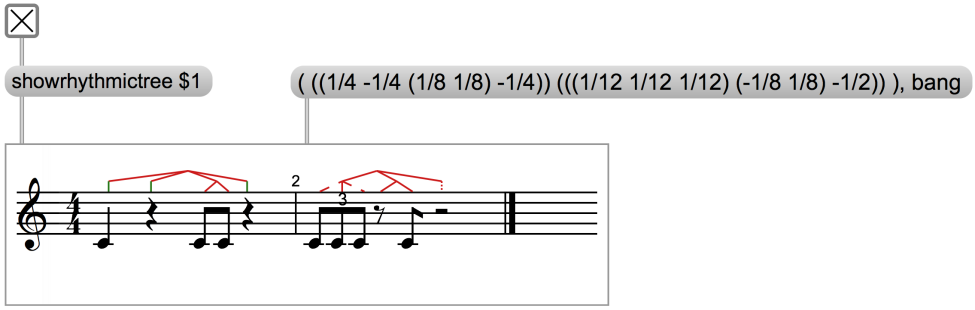
bach versteht sich auch ganz gut mit Artikulationszeichen.



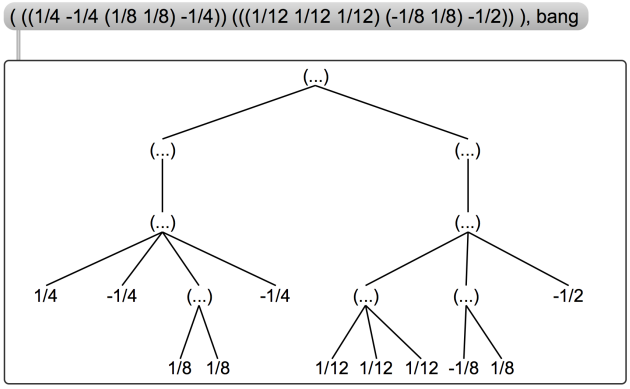
Leider sind derzeit (*bach* Version 0.7.7b) noch keine Legato-Bögen möglich. Zur Verfügung stehen:
staccato, staccatissimo, fermata, portato, accent, accent+staccato, accent+portato, portato+staccato, martellato, martellato+staccato, left hand pizzicato, trill, gruppetto, upward mordent, downward mordent, double mordent, upward bowing, downward bowing, 3-slash tremolo, 2-slash tremolo, 1-slash tremolo.

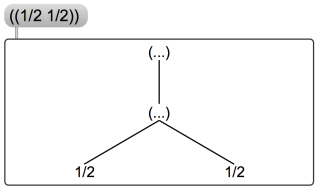
Wie in den Beispielen zu sehen ist, werden die Tonhöhen in *Midicents*[1] und die Notenwerte in Bruchzahlen angegeben. Für Pausen wird ein "-" vor die Bruchzahl gesetzt.

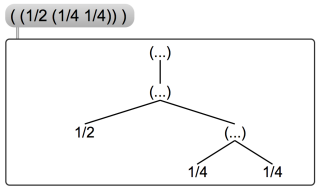
[1]Hier ist die Midi-Notennummer auch in der Cent-Einteilung in 1/100 stel-Schritten zu verstehen.

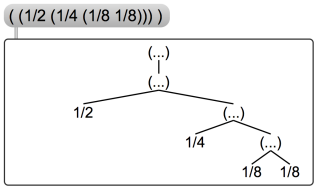


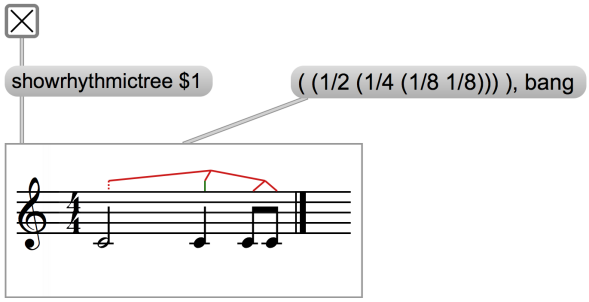
Bei *bach* wurde an sehr sehr vieles gedacht, und viele Raffinessen zur grafischen Darstellung wurden beigegeben. Beispielsweise das OBJECT *bach.tree* zum Darstellen oder auch Editieren der *llll*s.



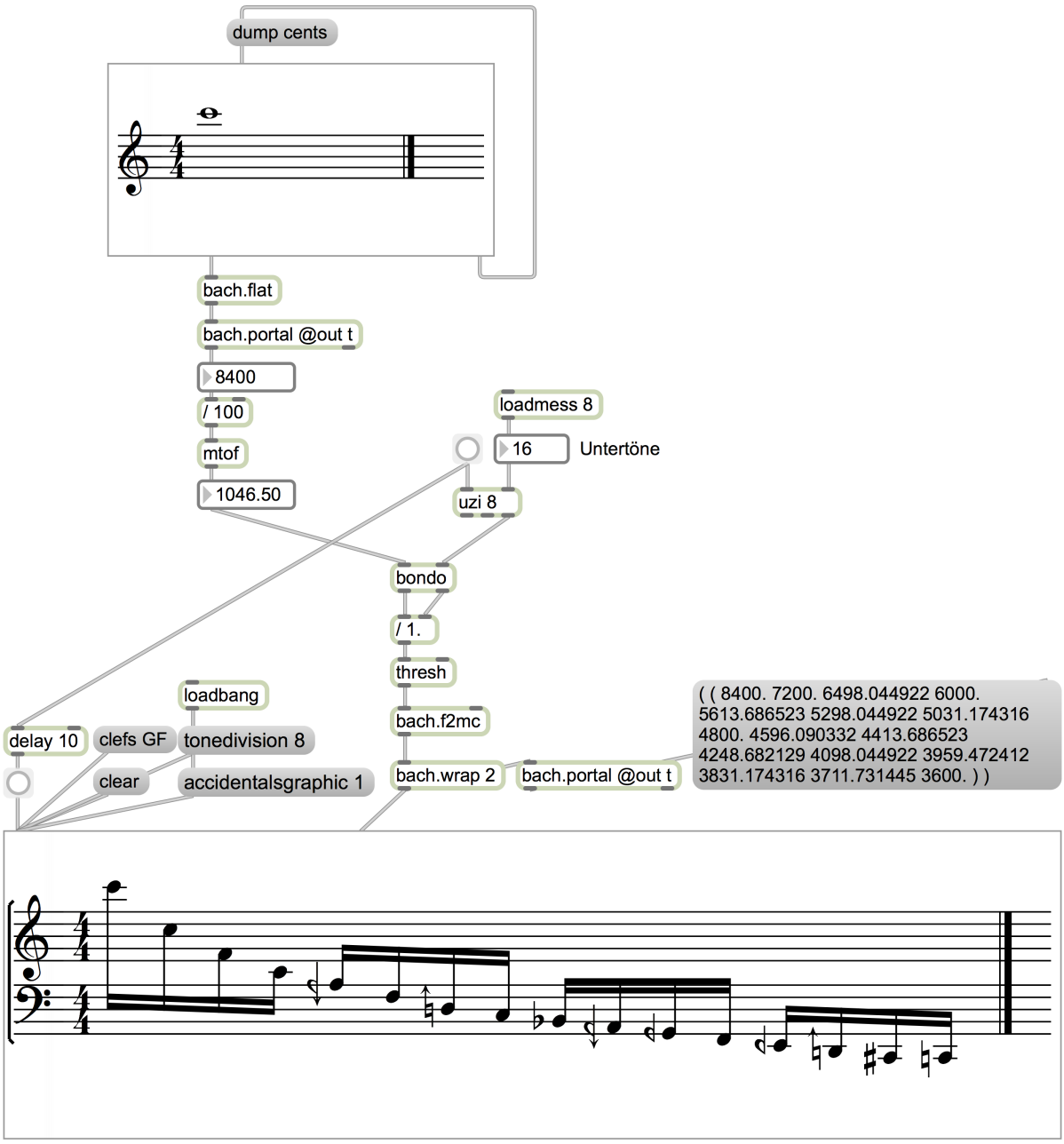








Tonhöhen werden bei Bedarf auch in Mikrointervallen berechnet und in üblicher mikrotonaler Notenschreibweise angezeigt.



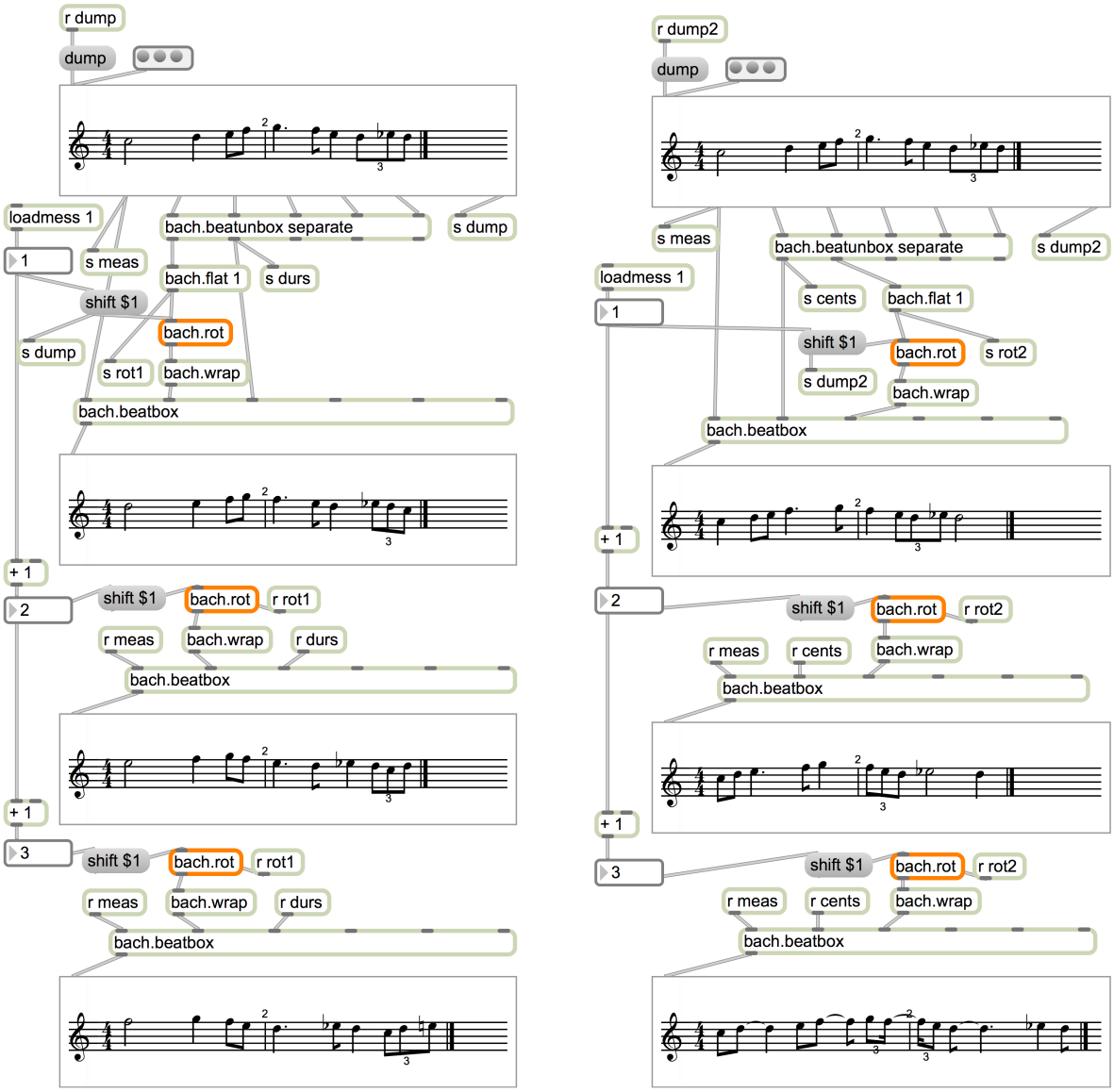
Stellvertretend für eine Fülle spezieller *Listen*-manipulierender *bach*-OBJECTS möchte ich drei vorstellen:

bach.rot bach.restrot bach.chordrot

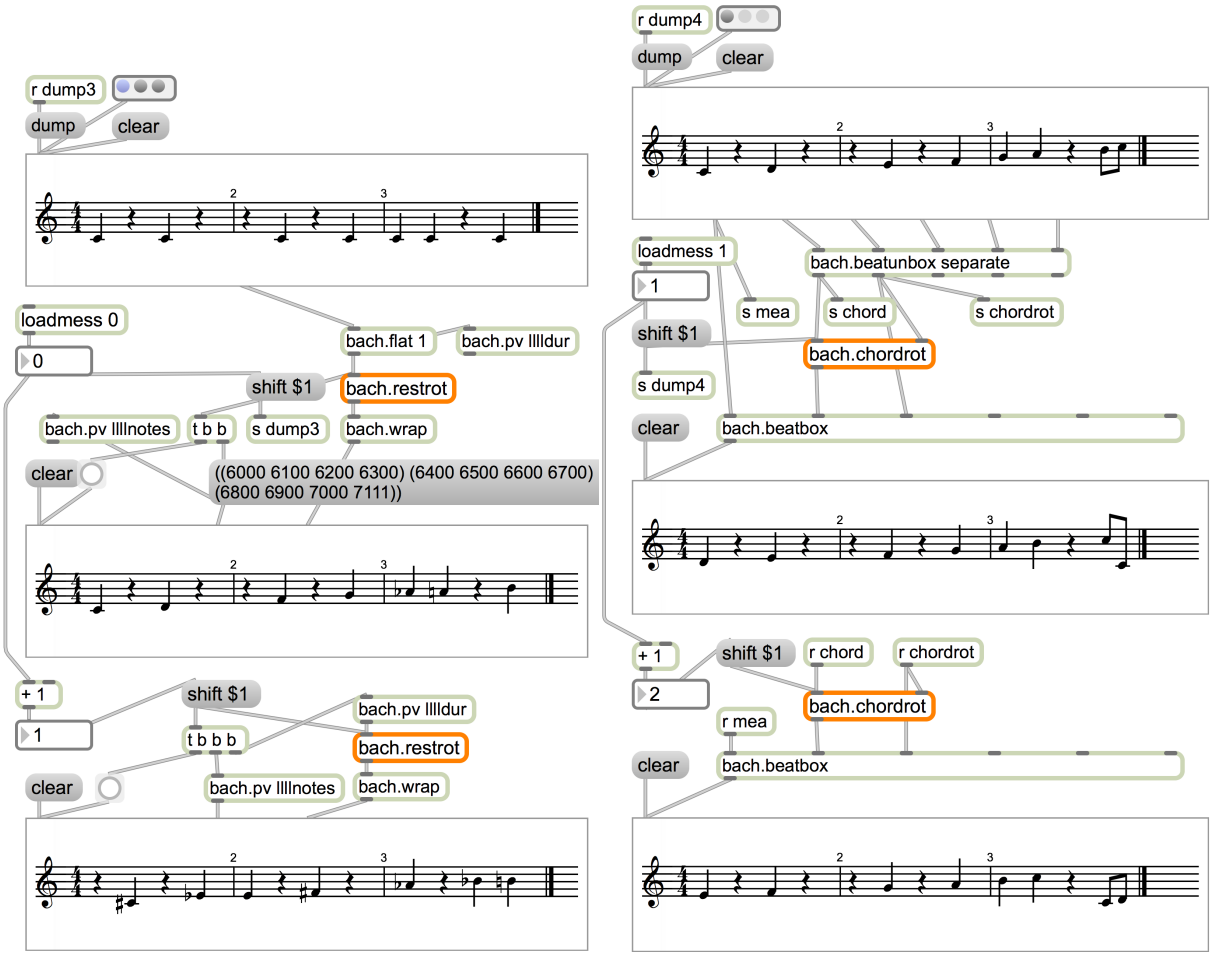
bach.rot führt eine Rotation von *llll*s mit ihren Sub-*Listen* aus.

bach.restrot rotiert die negativen Vorzeichen – also die Pausen – durch die *Liste*.

bach.chordrot rotiert nicht negative Elemente – also alles, was nicht Pause ist.

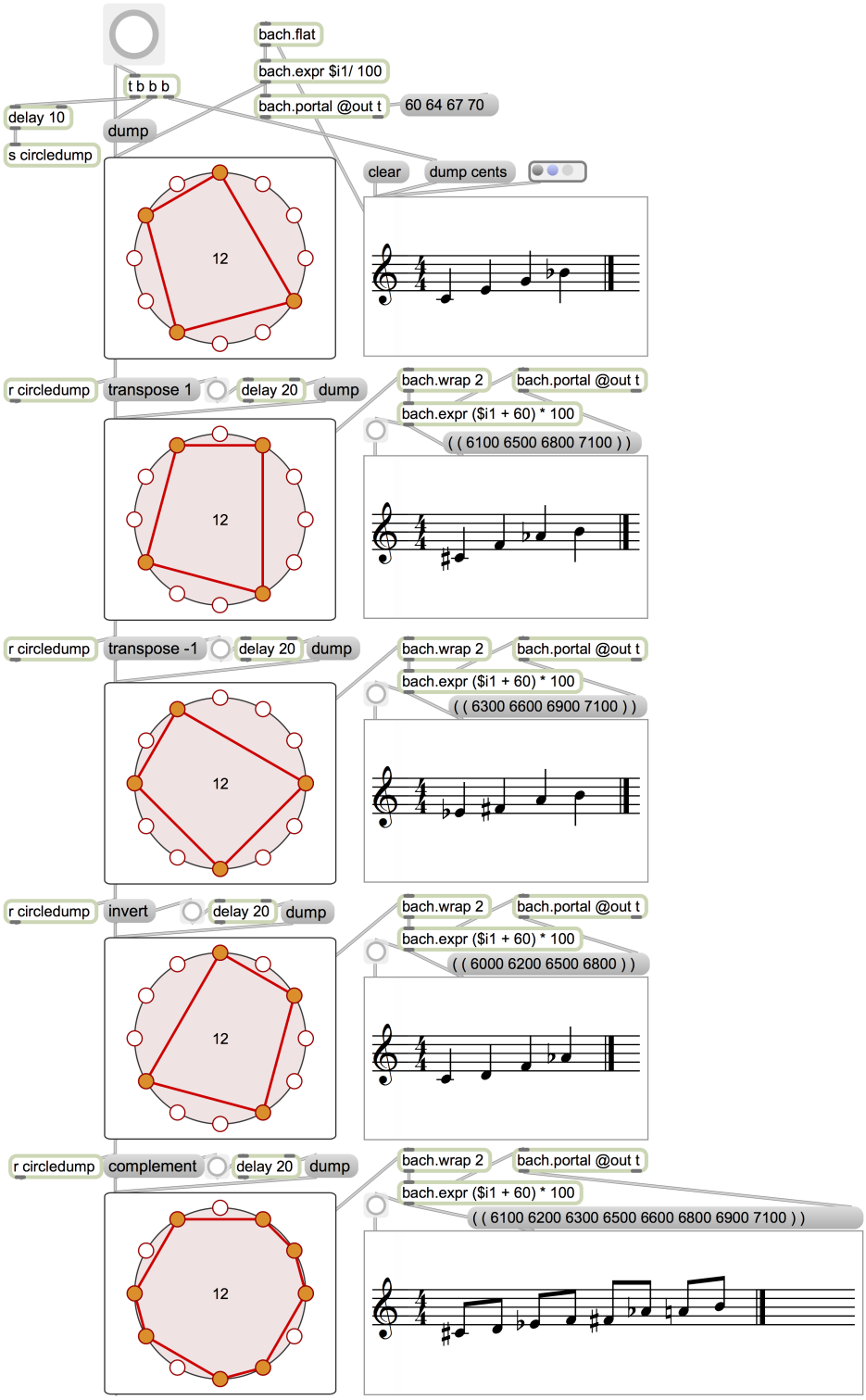


Links im Beispiel werden die Tonhöhen und rechts die Notenwerte rotiert.



In diesem Beispiel werden links die Pausen und rechts die Nicht-Pausen rotiert.

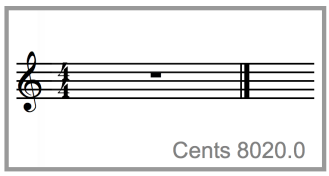
User-Interface-OBJECTS können zur Veranschaulichung dienen, wenn es um Themen der *Pitch-Class-Theory* geht.



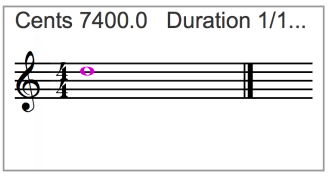
Die Notation in *bach* erfolgt entweder in *Echtzeit* durch Generierung und Umwandlung von *Listen* beziehungsweise *Listen* von *Listen* oder durch direkte Eingabe in die *bach.score*- oder *bach.roll*-OBJECTS.

Bei direkter Eingabe stehen viele Tastaturkommandos oder Mausaktionen – ähnlich jener, die in üblichen Notations-Programmen, beispielsweise bei FINALE[1] oder SIBELIUS[2], gefunden werden – zur Verfügung.

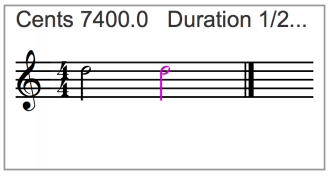
bach.score-Eingabe



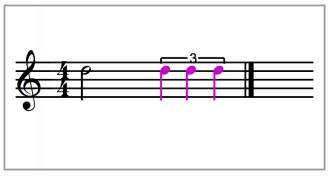
Durch *Cmd+Klick* (bei OSX) oder *Ctrl+Klick* (bei Windows) in das *bach.score*-OBJECT wird eine ganze Pause eingefügt.



Durch *Cmd+Klick* auf die Pause wird diese in eine Note umgewandelt (oder auch umgekehrt).

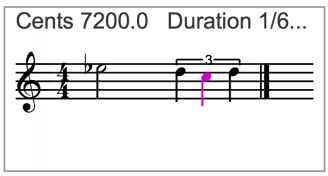


Durch Anwählen oder Selektieren einer Note (es ändert sich dabei die Farbe der Note) und Ausführen eines *Cmd+2*-Befehls werden die Notenwerte aufgeteilt. Das funktioniert bis *Cmd+9*.

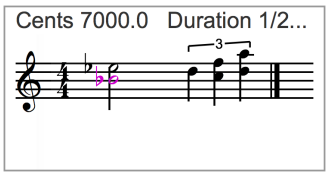


Hier das Resultat aus Anwählen der zweiten Note und Ausführen eines *Cmd+3*-Befehls.

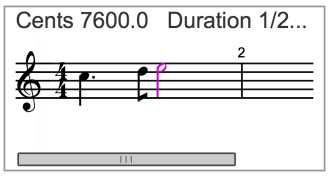
[1] https://www.finalemusic.com (Jänner 2015).
[2] http://www.sibelius.at (Jänner 2015).



Durch Anwählen von Noten und Betätigen der Pfeiltasten werden diese entsprechend auf- oder abwärts transponiert.

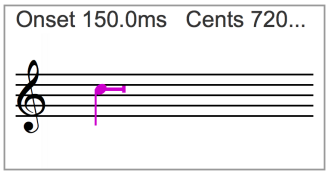


Durch Anwählen von Noten, Betätigen der *Alt*-Taste und Ziehen mit der Maus werden Noten vervielfacht und transponiert.

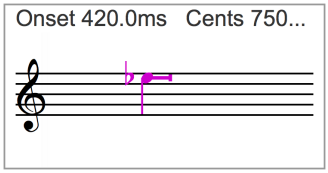


Durch einen *Ctrl+Klick* gelangt man in den *linear editing mode*. Es erscheint ein *pitch cursor*, der mit den Pfeiltasten bewegt werden kann. Notenwerte werden mit den Zahlentasten hinzugefügt. Eine 1/4tel-Note wird mit der Taste 5 eingefügt – so, wie wir es von FINALE kennen. (Eine 1/2-Note mit 6, eine 1/8tel-Note mit 4 und so weiter) Mit dem Punkt "." wird punktiert und mit der t-Taste wird eine Ligatur erzeugt. Mit der ESC-Taste gelangt man wieder in den normalen Editier-Modus zurück.

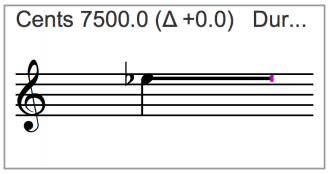
bach.roll-Eingabe



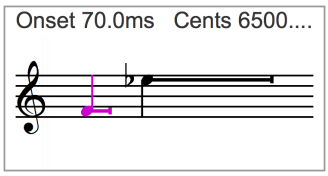
In *bach.roll* wird mit *Cmd+Klick* an derselben Stelle eine Note positioniert.



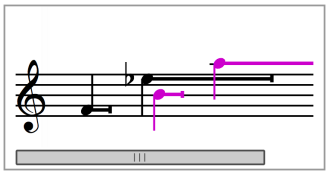
Durch Anwählen und Ziehen der Note mit der Maus wird ihre zeitliche Position oder/und ihre Tonhöhe geändert.



Durch Anwählen des Längenbalkens und durch Ziehen mit der Maus wird die Notenlänge verändert.

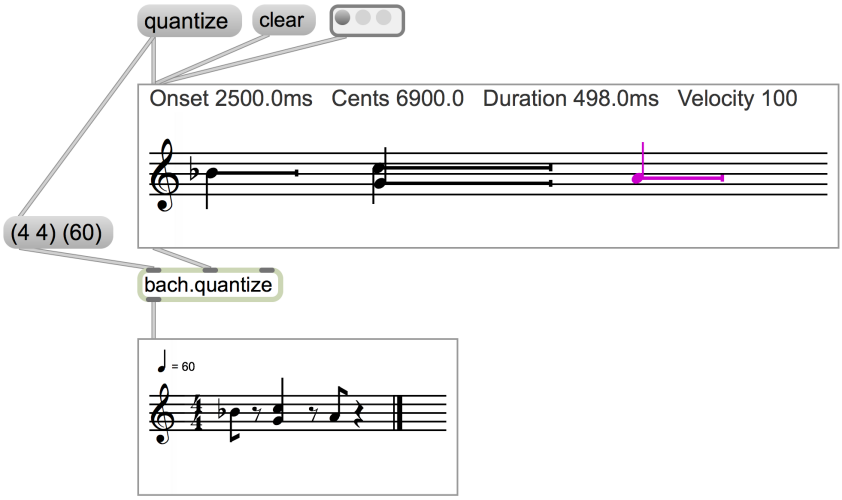


Durch *Cmd+Klick* werden Noten hinzugefügt.

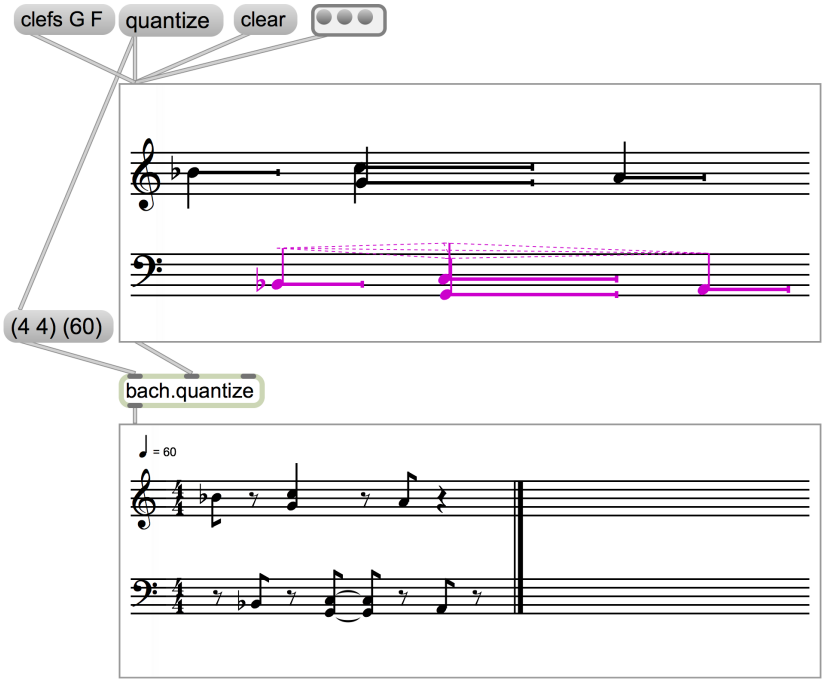


Durch Anwählen von Noten + Ziehen mit der *Alt*-Taste werden Noten oder Notengruppen bewegt und vervielfältigt.

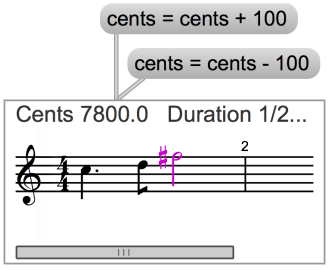
Die proportionale Darstellung in *bach.roll* wird mit dem *bach.quantize*-Object in eine traditionelle Darstellung in *bach.score* umgewandelt.

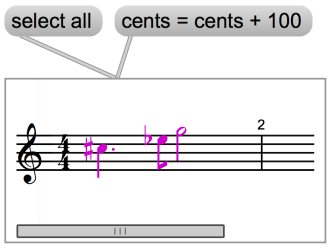


Mit der Taste G können Noten zu Gruppen zusammengefasst und gemeinsam bewegt oder manipuliert werden.



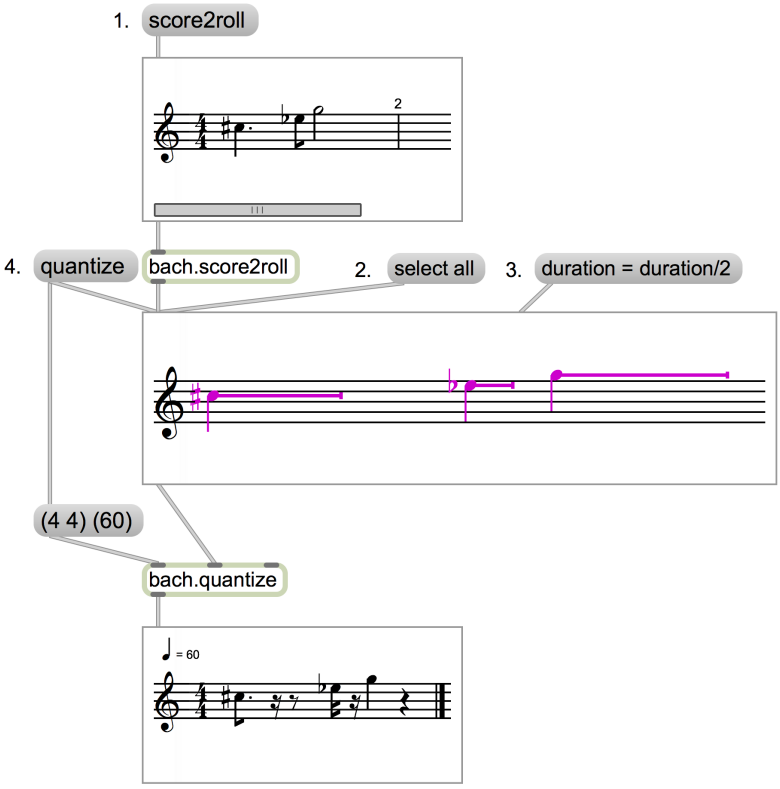
Noten werden mit der Maus oder mit differenzierten *Messages* angewählt und mit Tastatur- und/oder Maus-Kommandos oder wiederum mit speziellen *Messages* manipuliert.



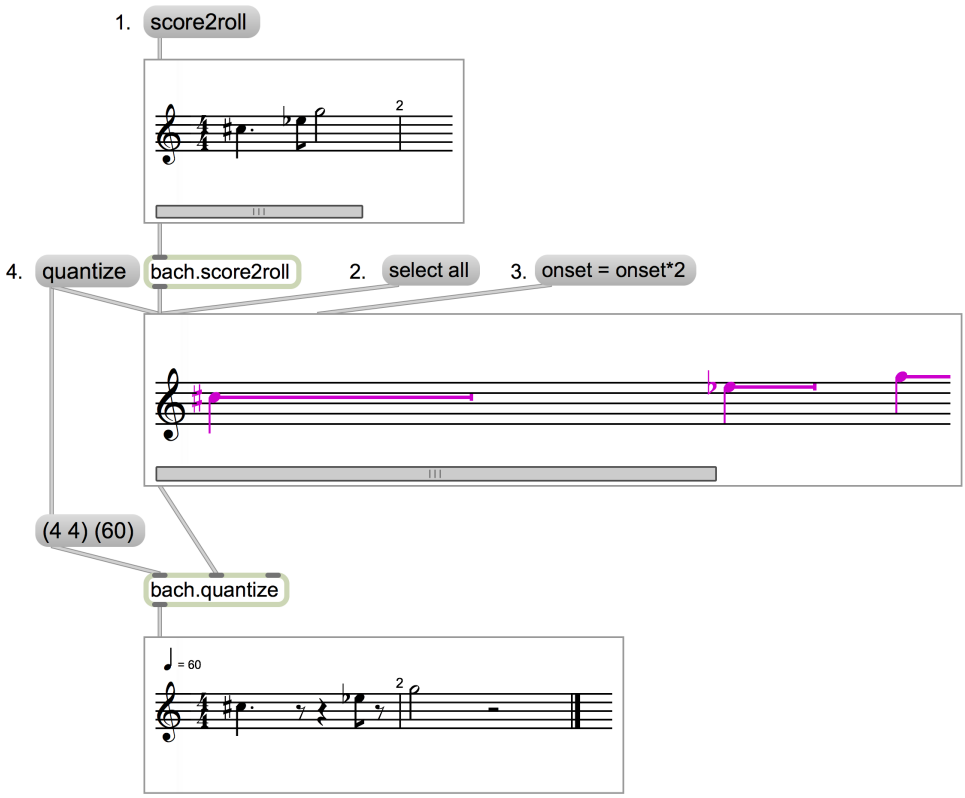


Bei den nächsten beiden Beispielen wird einmal die *duration* der Noten halbiert und einmal die *onset*-Zeit verdoppelt.

Halbieren der Duration

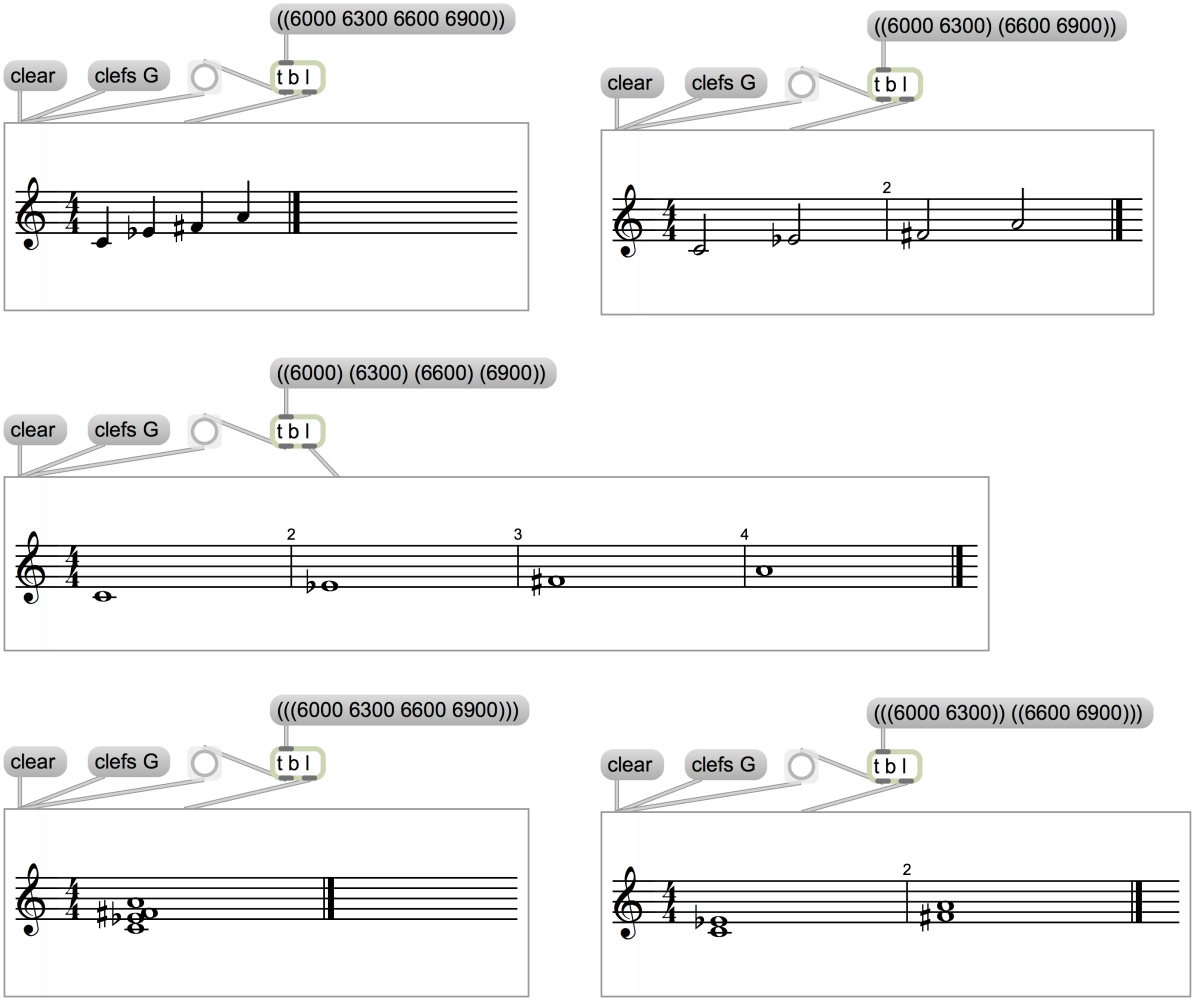


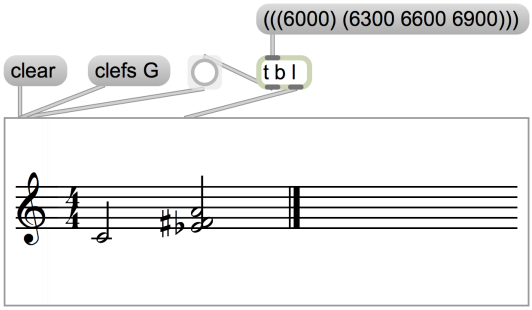
Verdoppeln der Onset-Zeit

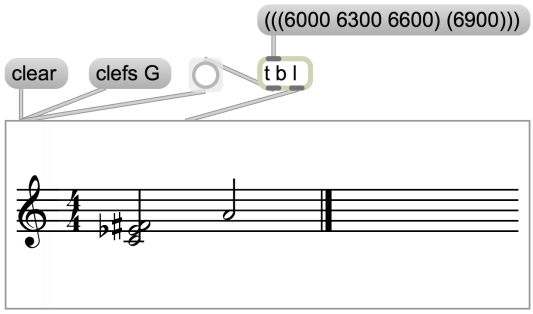


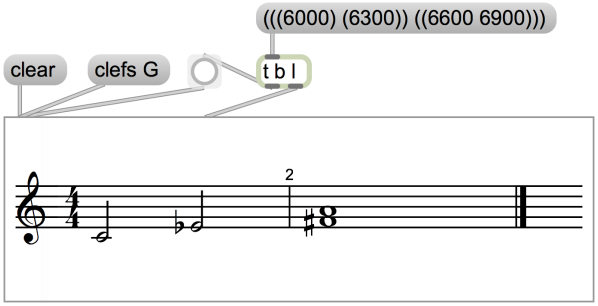
Klammern

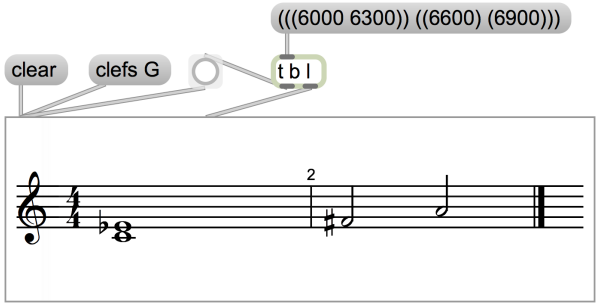
Wesentlich bei allen *bach*-„Operationen“ ist die Beschaffenheit der *Listen* – im Besonderen die gezielte Platzierung der Klammern.

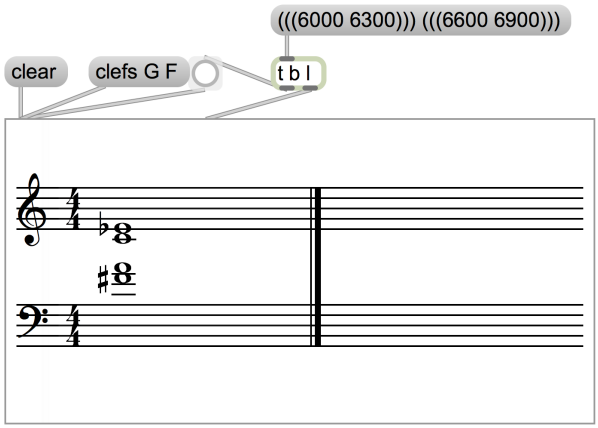


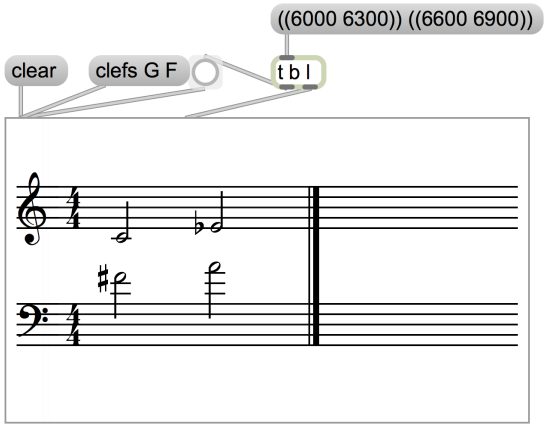

(((6000) (6300 6600 6900)))
clear
clefs G
t b l

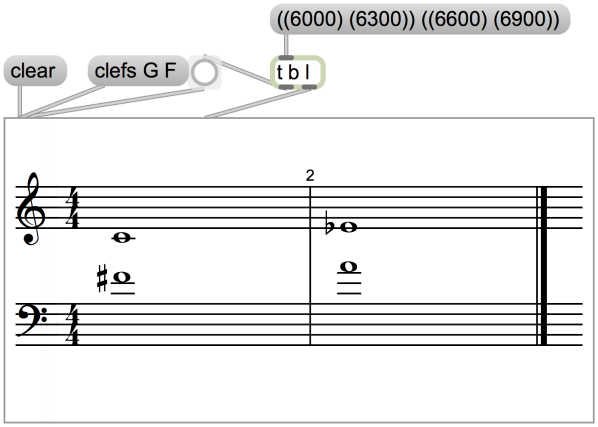

(((6000 6300 6600) (6900)))
clear
clefs G
t b l

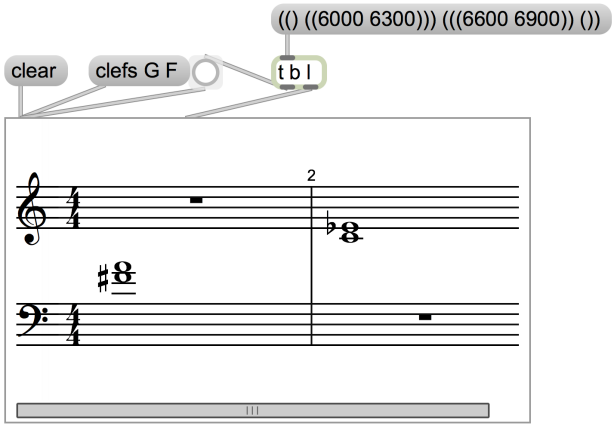

(((6000) (6300)) ((6600 6900)))
clear
clefs G
t b l
2

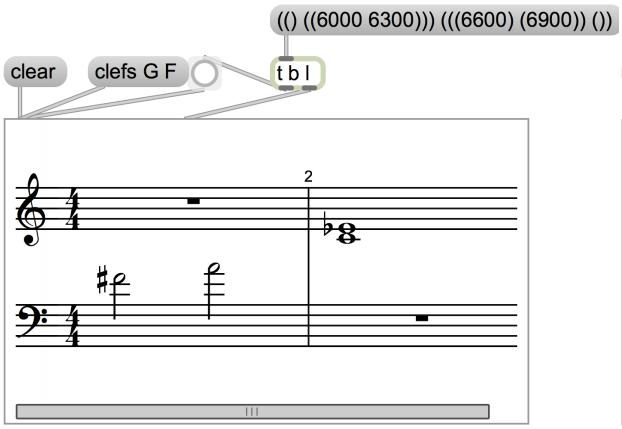

(((6000 6300)) ((6600) (6900)))
clear
clefs G
t b l
2

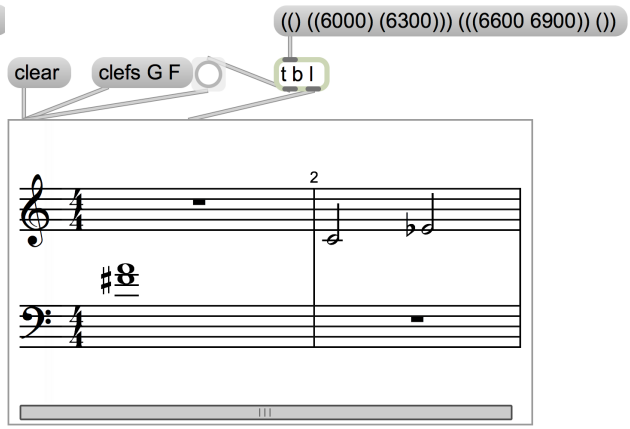

(((6000 6300))) (((6600 6900)))
clear
clefs G F
t b l

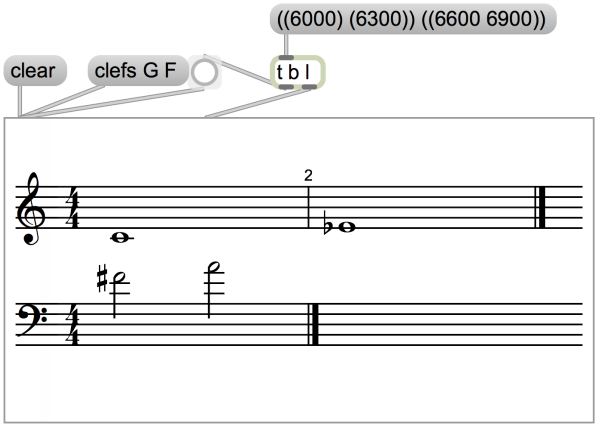

((6000 6300)) ((6600 6900))
clear
clefs G F
t b l


((6000) (6300)) ((6600) (6900))
clear
clefs G F
t b l
2

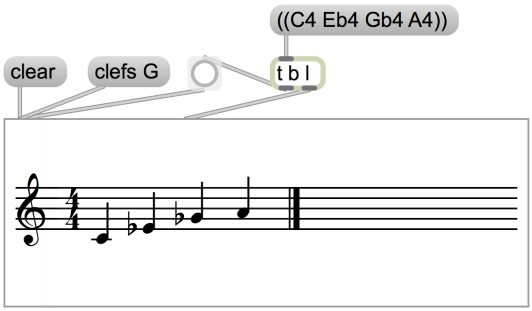
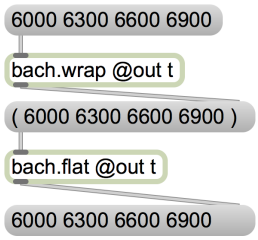

(() ((6000 6300))) (((6600 6900)) ())
clear
clefs G F
t b l
2


(() ((6000 6300))) (((6600) (6900)) ())
clear
clefs G F
t b l
2


(() ((6000) (6300))) (((6600 6900)) ())
clear
clefs G F
t b l
2


((6000) (6300)) ((6600 6900))
clear
clefs G F
t b l
2

bach.roll- sowie *bach.score*-Objects verstehen anstelle der Angabe in *Midicents* für die Tonhöhen auch Notennamen.



Zum Platzieren der Klammern sind die Objects *bach.wrap* und *bach.flat* eine große Erleichterung: