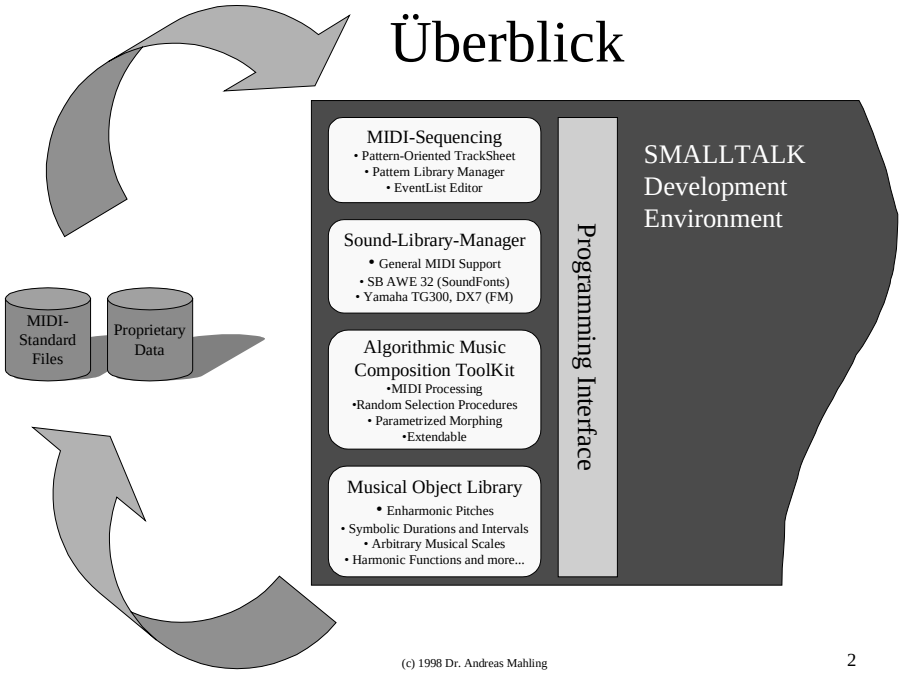
# MusicTalk
# Ein integriertes System für musikalische Komposition

Dr. Andreas Mahling
ARS NOVA Software GmbH
Rosenstr. 100
73733 Esslingen a. Neckar
mahling@arsnova.com

**Zusammenfassung**

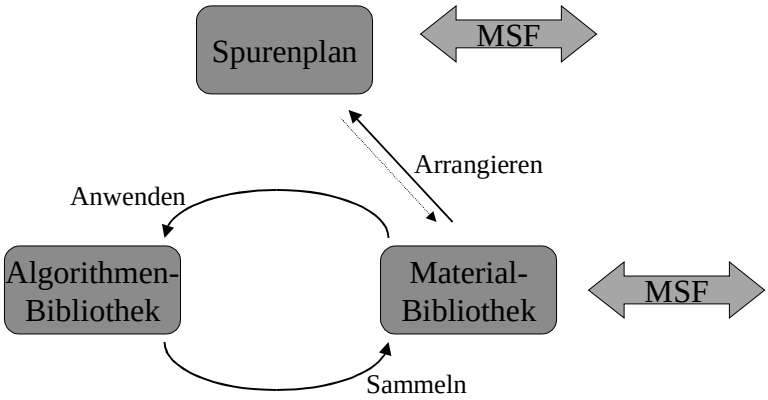
*Der folgende Beitrag beschreibt ein Software-System für die computergestützte musikalische Komposition. Neben den notwendigen Funktionalitäten wie Sequencing und Klangfarbenverwaltung verfügt die hier vorgestellte Kompositionsumgebung über eine Bibliothek musikalischer Objekte (Wissensbasis) und einen Baukasten für Algorithmische Komposition. Im Zentrum dieses Beitrags steht ein Baustein dieses Baukastens, der die Erstellung von Kompositionsprogrammen mit Hilfe einer ausschließlich dialogorientierten Benutzerschnittstelle erlaubt und dadurch die Komplexität von um musikalische Konstrukte erweiterten Programmiersprachen vermeidet.*

## Überblick




(c) 1998 Dr. Andreas Mahling

# Arbeiten mit MusicTalk (I)



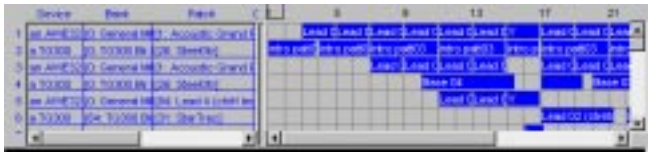

(c) 1998 Dr. Andreas Mahling


# Arbeiten mit MusicTalk (II)

| | |
|---|---|
| 1. Ebene | Anwenden vordefinierter Kompositionsalgorithmen |
| 2. Ebene | Erweiterung der Algorithmenbibliothek<br>Erweiterung des "Event-Generator" |
| 3. Ebene | Erweiterung der Klassenbibliothek musikalischer Objekte |

(c) 1998 Dr. Andreas Mahling

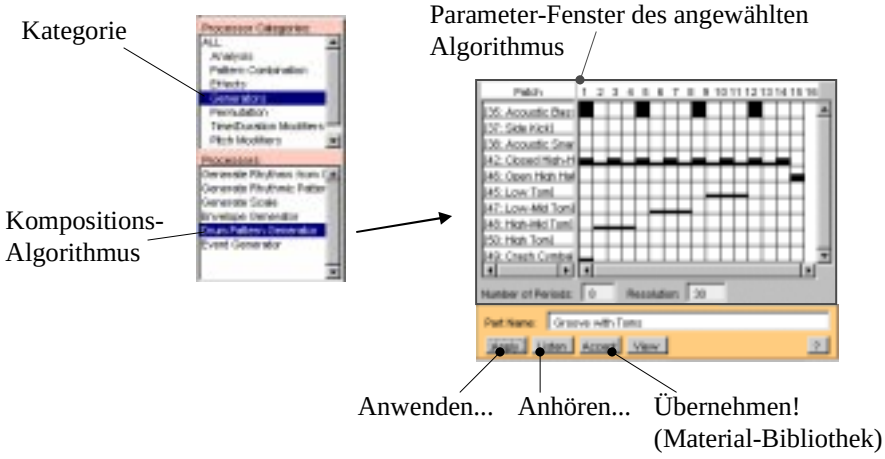
# Spurenplan




- Spurparameter:
  - Bank, Patch
  - Channel
  - Volume
  - Status
  - Panorama
  - Reverb, Chorus, Expression
  - Pitch Bend Range
- Arrangement-Operationen:
  - Duplicate, Cut
  - Drag & Drop
  - Split, Transpose, ...
  - Copy to Material Library
  - ...

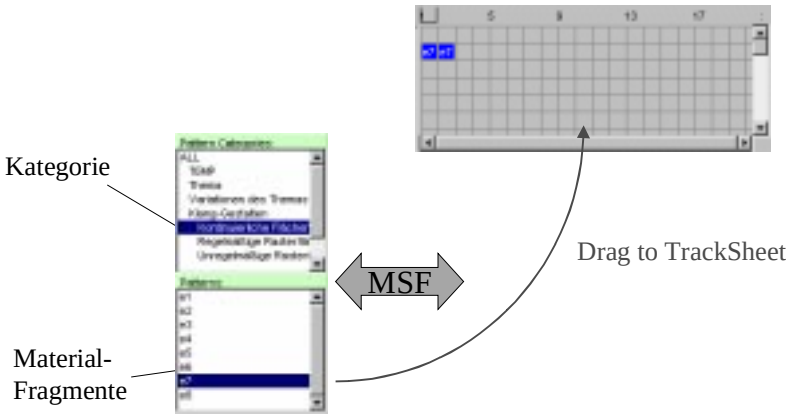
(c) 1998 Dr. Andreas Mahling


# Algorithmen-Bibliothek




(c) 1998 Dr. Andreas Mahling

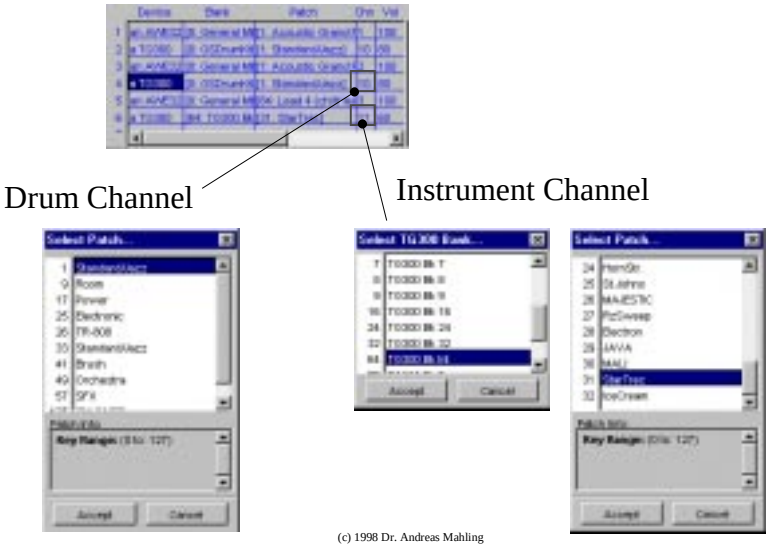
# Material-Bibliothek




(c) 1998 Dr. Andreas Mahling



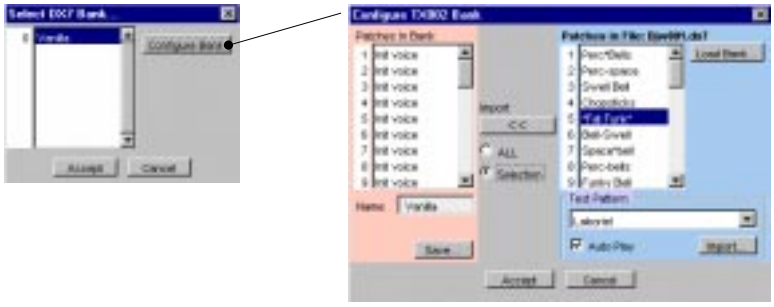
# Klangfarben-Verwaltung (TG300)




(c) 1998 Dr. Andreas Mahling

# Klangfarben-Verwaltung (DX7)

- Überträgt SysEx Dumps
  - vom/zum Gerät
  - von/zu Speichermedium
- Vorhörfunktion



(c) 1998 Dr. Andreas Mahling



# Klangfarben-Verwaltung (AWE32)

- Importiert .sbk Dateien direkt via Windows MCI: Daher kein manuelles Update von Bank-/Patch-Listen für eigene Klänge notwendig!



(c) 1998 Dr. Andreas Mahling

# Musik-Klassenbibliothek

- Erzeugen einer Ton-Skala:

```
| newScale |
  newScale := MTEqualTemperedScale
    name: 'equal-tempered-chromatic'
    lowestPitch: 16.3516          "8.1758"
    octaveRatio: 1200
    numberOfOctaves: 10.
newScale register
```

- Enharm. korrekte Transposition einer Tonhöhe:

```
(MTEnharmonicPitch newNamed: #d4)
  transposeBy:
    (MTInterval newNamed: #second option: #augmented)
```

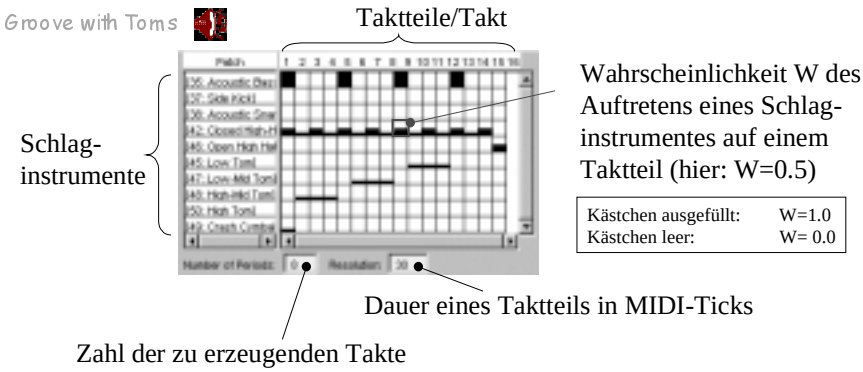
---

- Klasse
- Nachricht



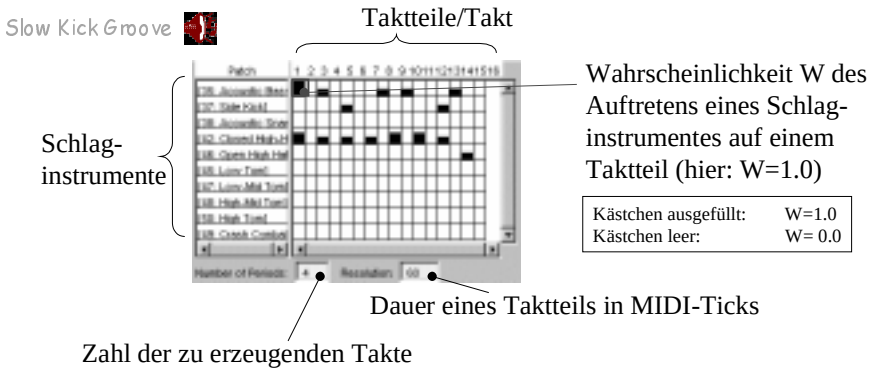
# Beispiele zur Algorithmen-Bibliothek:
## 1. "Drum Pattern Generator"

**Beispiel 1:**



# Beispiele zur Algorithmen-Bibliothek:
## 1. “Drum Pattern Generator”

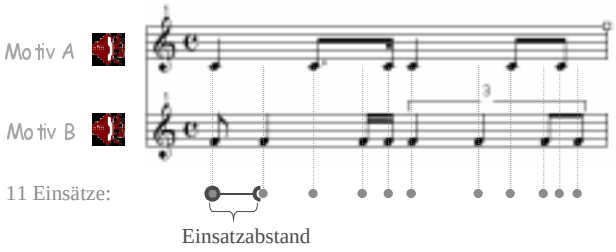
**Beispiel 2:**



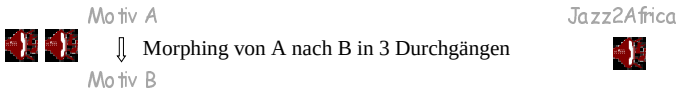

(c) 1998 Dr. Andreas Mahling



# Beispiele zur Algorithmen-Bibliothek:
## 2. “Morphing”

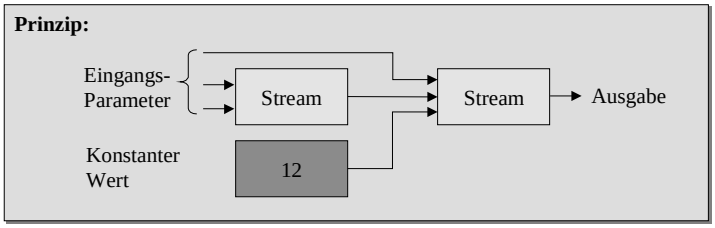



1.) Zufällige Auswahl eines Einsatzes aus den kumulierten Einsätzen von A und B.
2.) Ersetzen des durch den Einsatzabstand überstrichenen Bereichs von A mit B.
3.) Wiederholung von 1. und 2. solange noch nicht ausgewählte Einsätze vorhanden sind.

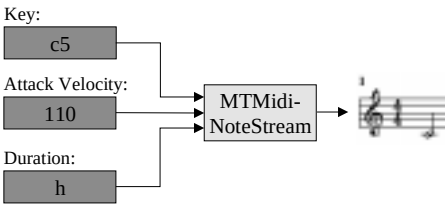



(c) 1998 Dr. Andreas Mahling

# Beispiele zur Algorithmen-Bibliothek:
## 3. “Event Generator”




**Beispiel 1:**

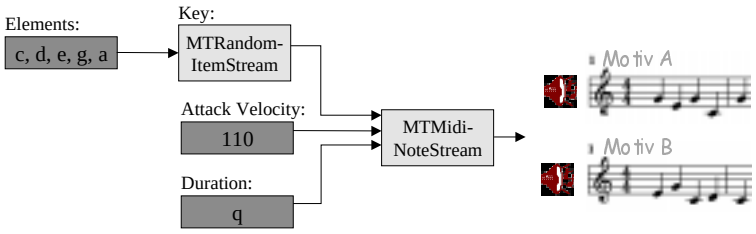



(c) 1998 Dr. Andreas Mahling



# Beispiele zur Algorithmen-Bibliothek:
## 3. “Event Generator”

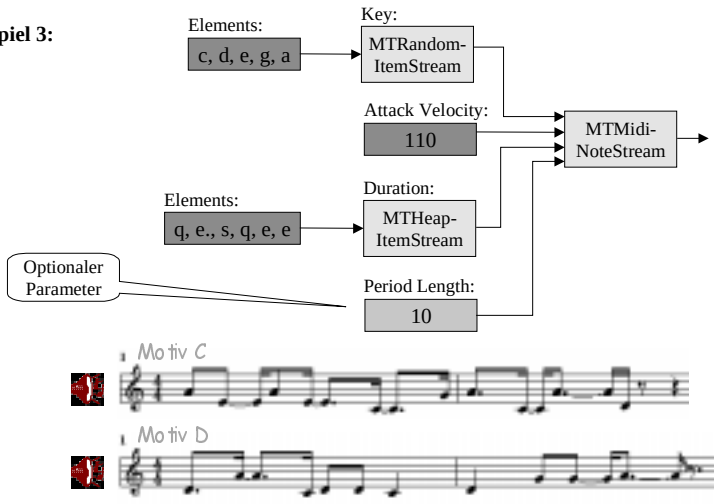
**Beispiel 2:**




(c) 1998 Dr. Andreas Mahling

# Beispiele zur Algorithmen-Bibliothek:
## 3. “Event Generator”

**Beispiel 3:**

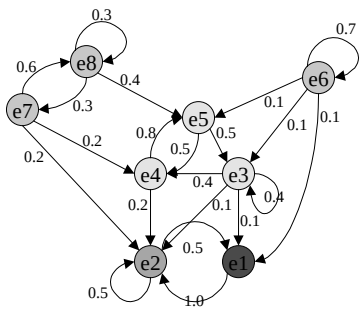


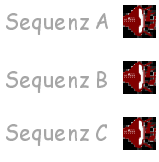

(c) 1998 Dr. Andreas Mahling
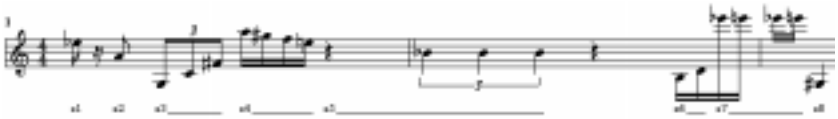


# Beispiele zur Algorithmen-Bibliothek:
## 3. “Event Generator”

**Beispiel 4:**
**(Übergangsnetze)**






(c) 1998 Dr. Andreas Mahling

# Beispiele zur Algorithmen-Bibliothek:
## 3. “Event Generator”

- Rhythmen mit zusammenhängenden Triolen
- Sich überlagernde Klangflächen wechselnder Simmenzahl
- Melismatische Bewegung mit Sekunden (Terzen und Quarten) und variierender Agogik.
- Unregelmäßiger Wechsel zwischen kurzen und langen Tönen / kleinen Intervallen und Oktavsprüngen

(c) 1998 Dr. Andreas Mahling