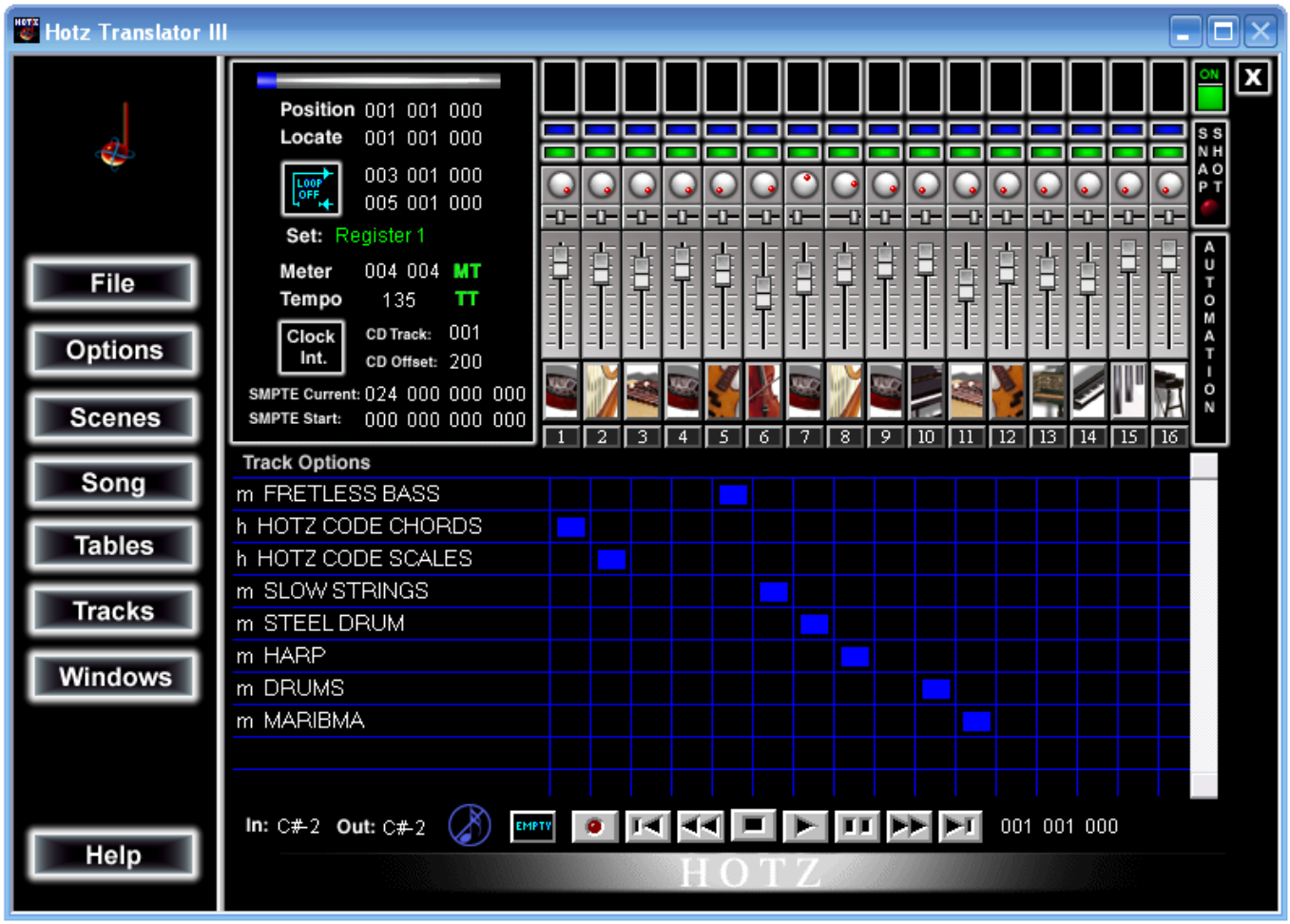
Hotz Translator III



Contents

Contents

Contents

Contents

Chapter 1: Quick Start

Installing the Hotz Translator

To install the Hotz Translator software simply insert the CD. On most computers this will Auto Launch the setup program. If this does not happen on your computer, run the setup program that is on the root of the Hotz Translator CD. Once the setup program is running, follow the simple instructions to install the program on your system.

After installing, you may run the **Translator** by double-clicking on its icon. The Hotz Translator software can also be started by double-clicking on **Hotz_Translator_III.exe** located in the **Hotz_Translator_III** directory using the Windows Explorer.

Note: *Some of the Screen Shots in this manual may show images from an older revision of the software, however the general functionality is the same.*

Recommendation:

Although you may use the computer keyboard as an input controller for the Hotz software, we recommend using an external MIDI controller such as a MIDI Keyboard, Guitar or other MIDI Controller. Why? Many computer keyboards are constructed in ways that make them respond poorly to certain simultaneous key depressions. This means that you may not always get the number of notes out that you expect. Some keyboards such as the "Maxim" made by "Kinesis" do not have this problem but are still limited to fixed velocity values.

Most sound cards have built-in MIDI interfaces that require a special cable for connection to external MIDI devices. If you are not familiar with MIDI interfaces or how to properly install and set them up on your system please refer to the manufactures documentation.

Default Example

Press play and start performing along on the computer keyboard or your MIDI controller!

The power of the Hotz technology is best understood by experience. Load several of the different song files included to broaden your initial experience.

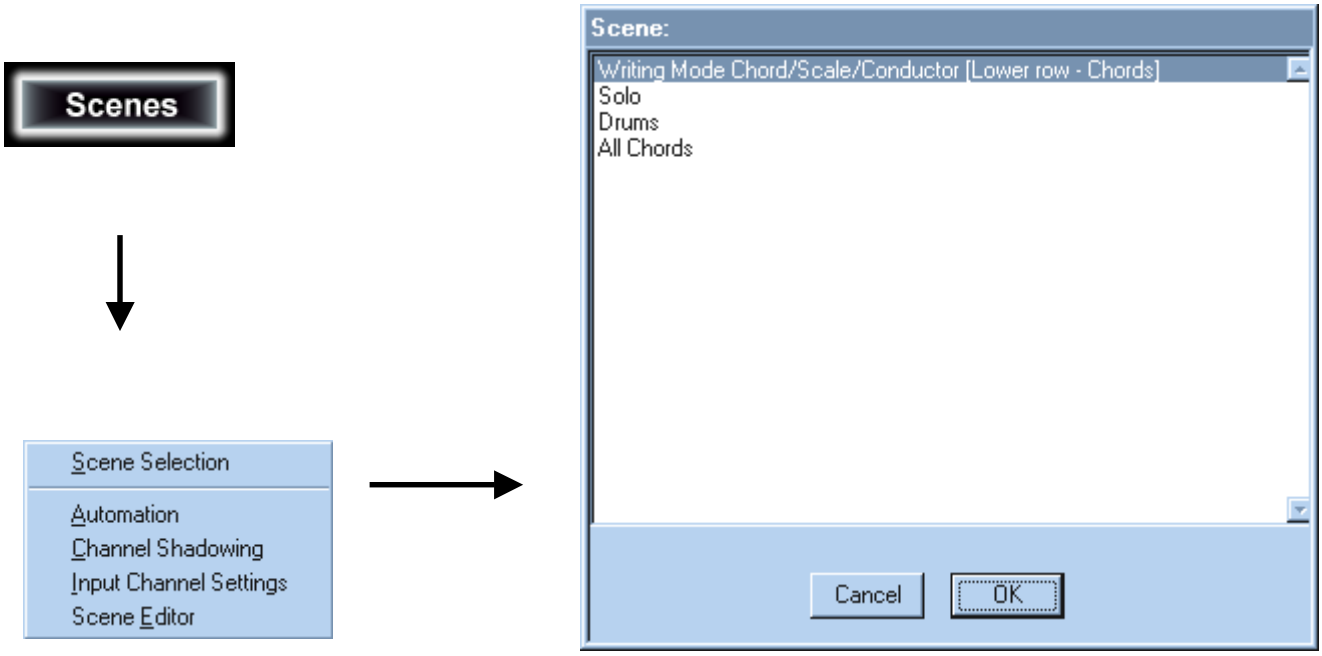
Press on the **Scenes** button and choose "**Scene Selection**". Many songs have a variety of Scenes or global setups. Choose one and experiment.

Chapter 1 Quick Start

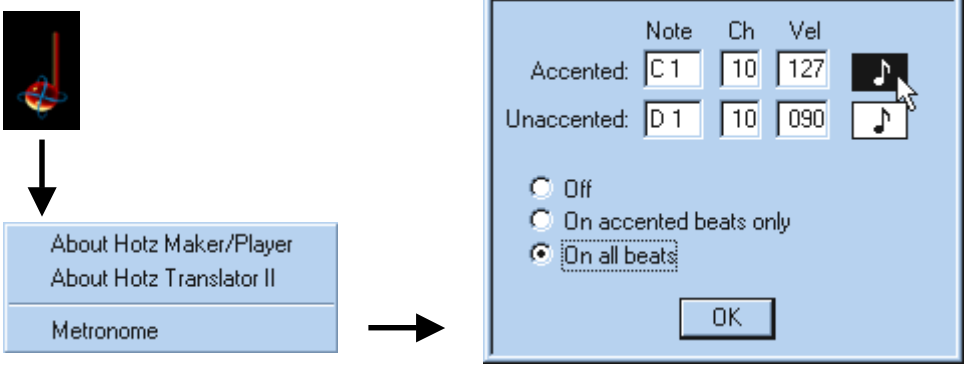
Hotz-coding a Simple Sequence

1. Load the **WritingTemplate.htz** file.

2. Make sure the current **Scene** is "**Writing Mode Chord/Scale/Conductor**".



3. Turn on the **Metronome** to use as a timing reference. (When you become more familiar with the Translator you may import a drum track or create one yourself to use as a timing reference.)

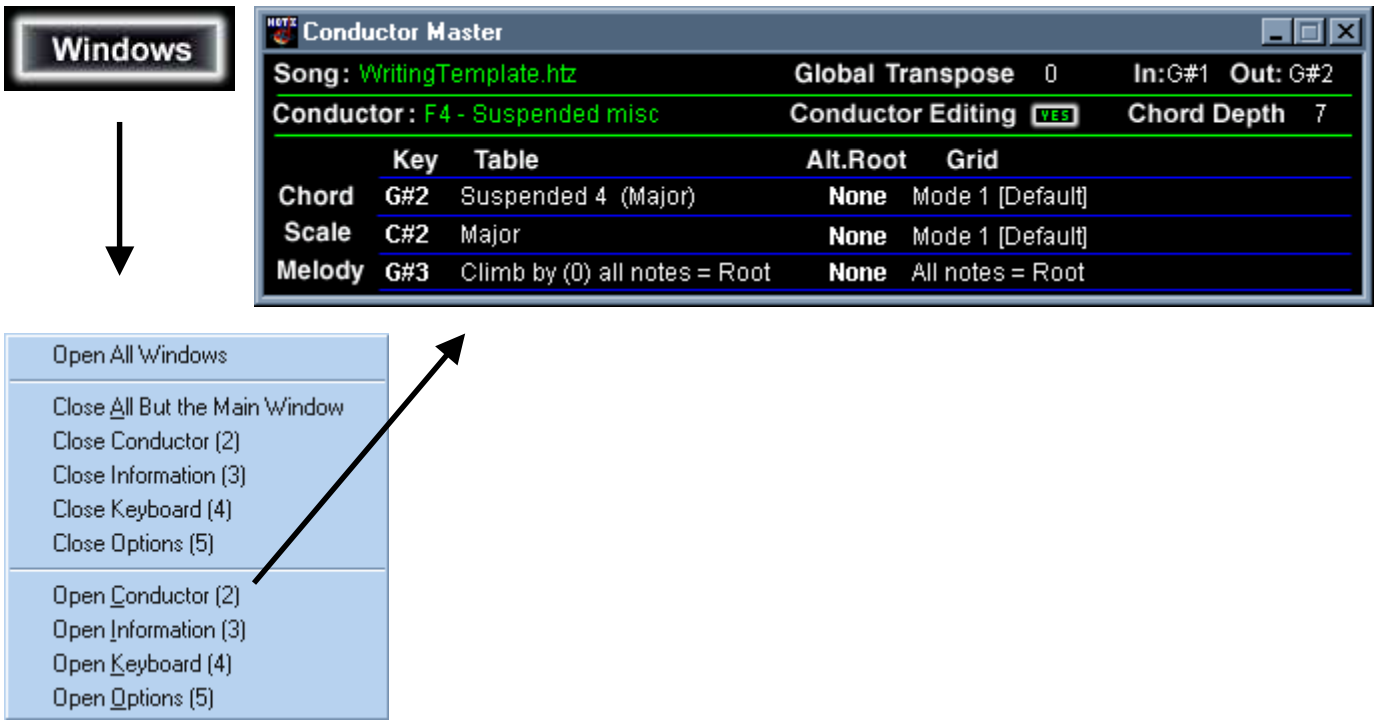


Set **Note**, **Channel** and **Velocity** to desirable values (You may test these by clicking on the note icons to the right of these values).

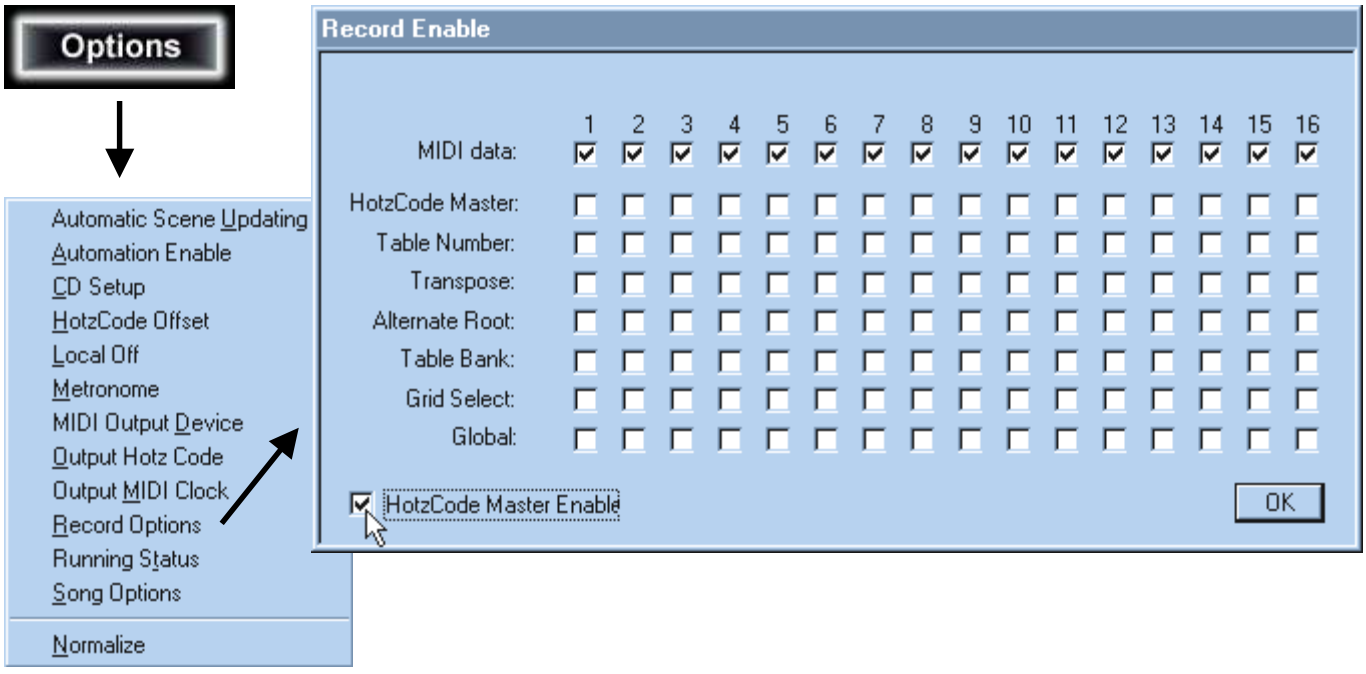
Enable **On all beats** by left-clicking.

Chapter 1 Quick Start

4. Open the **Conductor Window** if it is not already open. This will allow you to see the **Chord** and **Scale** changes as they occur.

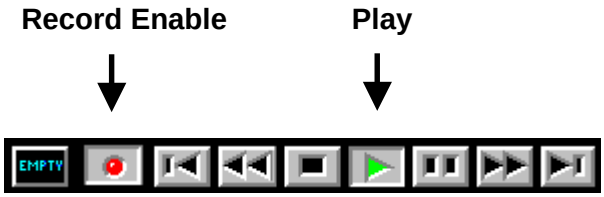


5. Enable **HotzCode** recording. Select **Options** and choose **Record Options** when the **Record Enable** dialog appears put a check by **HotzCode Master Enable.**



6. Press the number 1, 2, 3, 4, 5, 6, 7, 8 keys on the number row of your keyboard (not the numeric keypad). When the "**Writing Mode Chord/Scale/Conductor" Scene** is selected, these keys generate **Conductor** (**HotzCode**) events which cause the conductor tables assigned to these keys to switch in real-time and output a Chord(note array) based on the chord tables that are selected by these **Conductor** (**HotzCode**) events. At this point you should hear the Chords sounding. If you do not, please refer to the section on **MIDI Output**. You should also see the names of the Chords changing in the Conductor Window.

7. Every time you press one of these keys you are generating HotzCode in real-time. All you have to do now is put the Translator in Record, Hit Play and step through a series of chord changes by pressing the keys on the number row of your keyboard in any sequence you desire. Experiment with a few patterns such as (3,2,3,4,2,3), (6,4,4,5,3,3,4,2,2,3) or any other sequence you like. When you hit Stop you will have recorded your first HotzCode.



You may now play along to these changes, add over-dub after over-dub and experiment with real-time performance.

This was, of course, a very simple "first HotzCoding" experience. However, the exact same procedure may be applied to any complexity of structure changes. As you explore the depths of this technology, it will become more and more clear that by using the Hotz Translator you can flawlessly perform musical passages that seemed unattainable before.

At this point you might like to set up your own Chord and Scale changes. See the next section "*Writing with your own Chords and Scales*" to understand how this is done.

We recommend having the Translator running while you read through this manual.

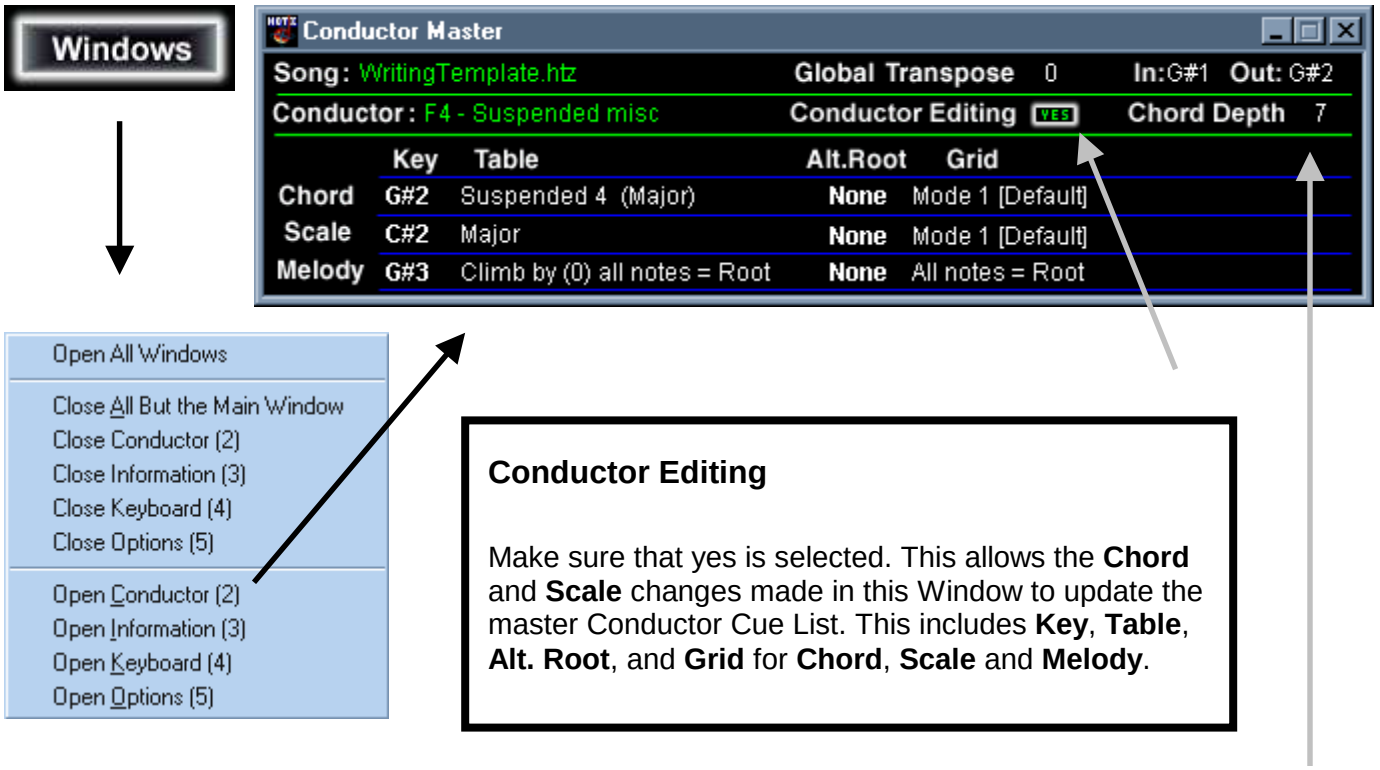
Note: The names of the Chord and Scale tables are shown with both sharps and flats. However, most of the editing screens and note displays use enharmonic equivalents. For these functions all sharps and flats are shown only as sharps. This means that Eb is displayed as D#, Ab as G#, etc. This was done to optimize processing speed and memory efficiency.

Chapter 1 Quick Start

Writing with your own Chords and Scales

Use the same procedure as the previous **Hotz-coding a Simple Sequence** with the following additions.

Open the **Conductor Window** if it is not already open. This is where you will select your own **Chords** and **Scales** for HotzCoding (conducting). Make sure that **Conductor Editing** is set to **YES**.



Conductor Editing

Make sure that yes is selected. This allows the **Chord** and **Scale** changes made in this Window to update the master Conductor Cue List. This includes **Key**, **Table**, **Alt. Root**, and **Grid** for **Chord**, **Scale** and **Melody**.

Chord Depth

When Conductor events are played you have the option of hearing the Chord structure as you send out the Conductor events. These Conductor Chords (note arrays) are normally output on channel 1. **Chord Depth** allows you to determine the number of notes used to generate the Conductor Chords.

A value of (-1) means use the values stored in the Conductor Cue List. A (0) means do not output notes. All other values are the actual number of notes that will be output.

To change the value left-click and drag. You may also single left-click to decrement the value and single right-click to increment the value.

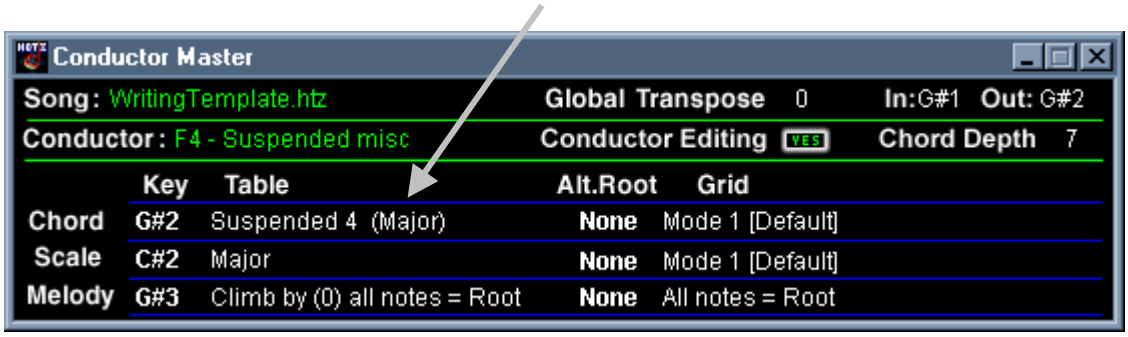
Note: This value does not affect the actual structure in any way, it is available as an audible aid and creative tool. When recording, this data is stored on a separate track from the HotzCode and may be independently turned on/off or deleted without affecting the HotzCode itself.

Chapter 1 Quick Start

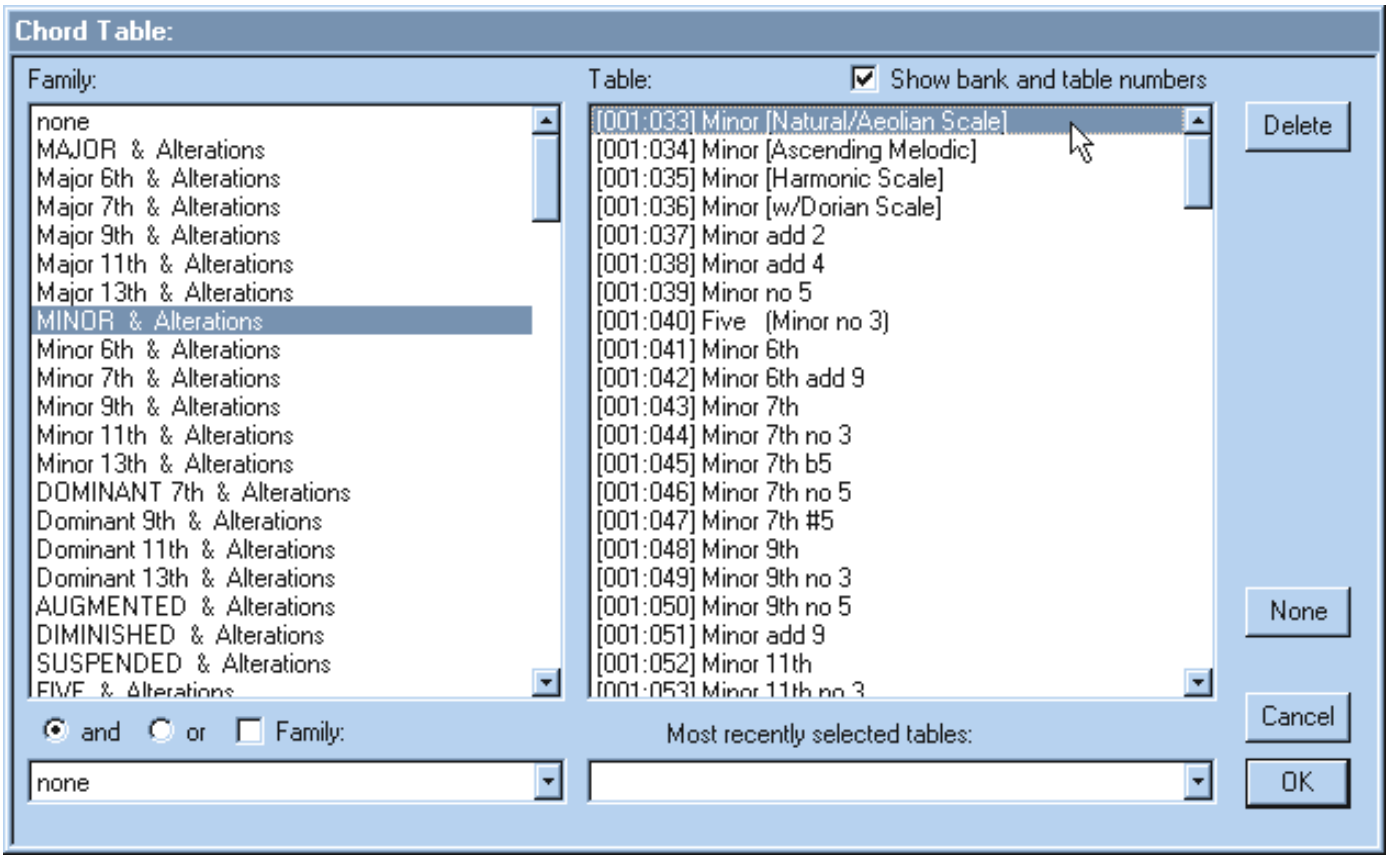
As in step 6 of **Hotz-coding a Simple Sequence** pressing the number 1, 2, 3, 4, 5, 6, 7, 8 keys on the number row of your keyboard (not the numeric keypad) causes the conductor tables, assigned to these keys, to switch in real-time and output (**HotzCode**) events. If the Chord Depth is set to anything other than "0", a Chord (note array) with also be generated.

First press the number 1 key.

Now select a different chord table by left-clicking on the table name.



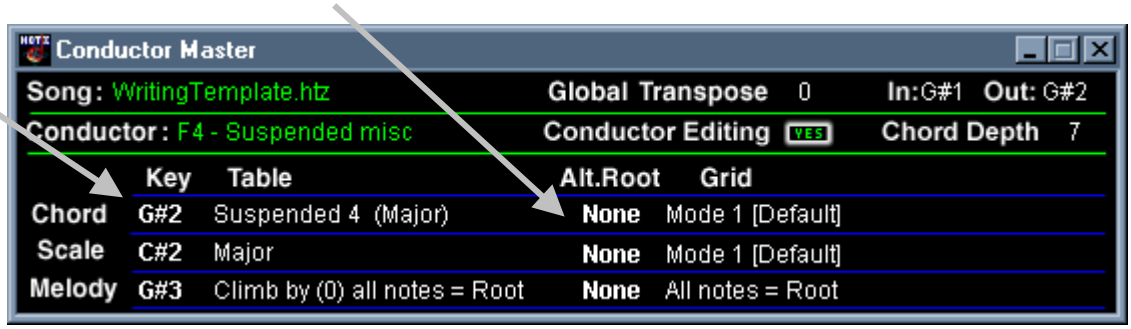
The following dialog appears. Left-click on one of the **Family** names in the left column and a series of **Table** structures that are associated with that family will appear in the column to the right. To select a Table either double-click on it or single left-click and press O.K. Now the current Chord will be the one you just selected.



Chapter 1 Quick Start

Next select a Key for the Chord you have just selected.

To change the value left-click and drag. You may also single left-click to decrement the value and single right click to increment the value. The **Alternate Root** may also be edited this way.



Press the 2 key and select a Chord table and Chord key in the same way.

Now do the same with the other number keys 3,4,5,etc. Scales are assigned the same way.

That is all there is to setting up your own changes. When you press the number keys they should now switch the Chords and Scales to those you have selected.

If they do not please verify that you have:

1. Loaded the song "WritingTemplate.htz", this is a special file already set up for writing.
2. Selected the Scene "Writing Mode Chord/Scale/Conductor".
3. Made sue that **Conductor Editing** is set to **YES**.

Each Conductor event can simultaneously output all of the data listed in the Conductor Window. However, in practice it is typical that only the **Chord Table, Chord Key, Scale Table** and **Scale Key** are used for most songs. **Grids** and **Alt. Root** are in place for more advanced HotzCoding, but please feel free to experiment.

It is possible to have extremely elaborate Conductor setups and each song can have a number of different setups. To select a different Conductor setup left click on the name of the current Conductor setup (this is just to the right of the word "**Conductor:")** and make a selection. You can create a different series of changes and have them readily available in the same song file.

Always save your song after you have set up a series of Chords and Scales. You can use these as writing templates in the future.

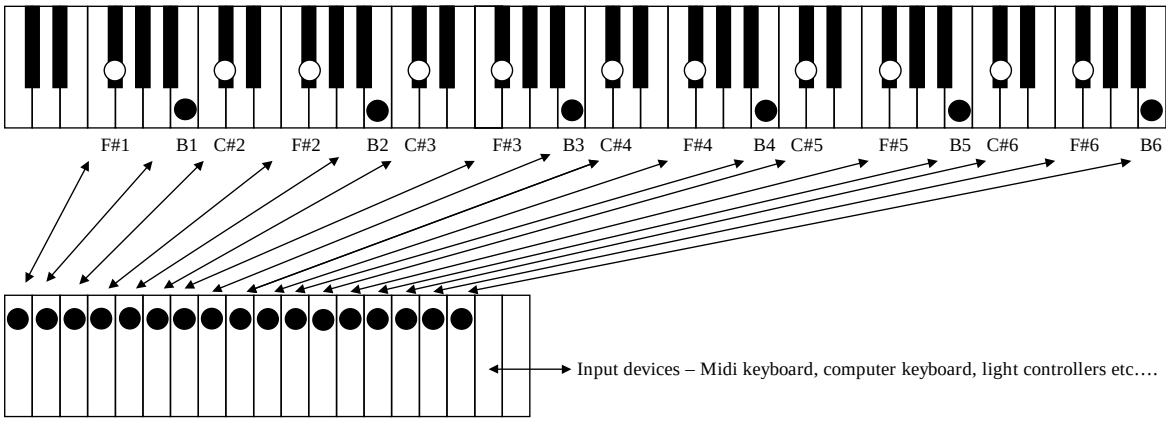
Chapter 2: Hotz Translator Overview

A New Way to Make Music

Despite the tremendous advances in technology, the process of becoming an accomplished musician**,** especially one trained in the theory and complex language of music**,** has remained the same for centuries. This usually consists of many years of constant, demanding practice of physical patterns in conjunction with the study of diverse and often confusing approaches to musical theory. Many musicians are prevented from reaching their creative potential when faced with such a difficult path. Similarly, musical instrument interfaces initially developed hundreds of years before electricity are still being used to control electronic instruments and data, oftentimes with limited success.

By integrating a unique software MIDI processing environment, and a vast database of chord/scale structures, the **Hotz Translator** provides a profound and effective computer-assisted musical instrument. The Translator makes it possible to play instantly any desired musical structure (chord or scale), in any key/octave, and any desired voicing with precision, speed, and creative intention. All without retuning or using complex and ever-changing fingerings, and without having to avoid the "out of tune" notes that are not part of the structure! In a Hotz system, the demanding tasks of musical structure memorization and physical pattern layout are handled by the computer. This leaves the user free to concentrate on the most creative and satisfying aspects of the musical experience, playing and improvising music.

Placement of the notes of a F# sus 4 chord across multiple octaves on a 6 octave MIDI keyboard.



F#1B1C#2F#2B2C#3F#3B3C#4F#4B4C#5F#5B5C#6F#6B6 etc.....

Full expansion of notes for the same 6 octaves of an F# sus 4 chord on a Hotz Translated Instrument. All desired notes appear sequentially translated for 100% efficiency, with all possible chord voicings available.

What makes the **Hotz Translator** unique and a truly musical instrument is the way in which it solves the problem of changes from one structure to another while the player is performing. This crucial process makes it possible to change a table while still sounding notes the player is sustaining from the previous table. The result is indistinguishable from what a musician does when changing from one chord or scale to another by changing fingering patterns. **The instrument is dynamically translated to follow chord and scale changes throughout your performance, so you always have the musically correct notes laid out before you.**

Chapter 2: Overview

The Hotz coding process sequences the structure/table changes without requiring the player to do anything but concentrate on playing music. Thus the player can focus completely on which notes of a particular chord or scale should be played and for how long, which is the very essence of music. Changes of structures can be under pre-encoded software control, or made in real-time by the player as part of playing musical passages (the Conductor feature adds this extremely powerful dimension to all skill levels of users). The result is realistic improvisational control impossible without decades of study and practice.

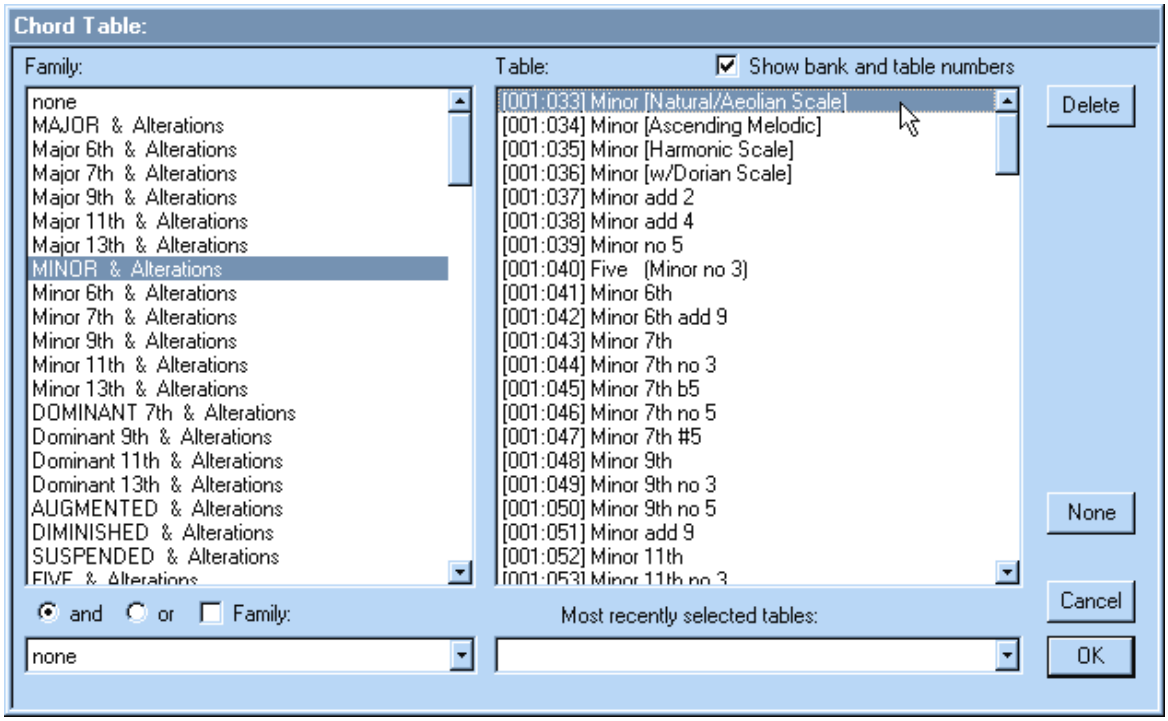
The change to a different structure/table is as flawless and transparent as if you physically had moved to a new fingering pattern, yet you can play the same notes on your keyboard and translation results in the new pattern being played. This means that a novice musician can, for example, play only the white notes of a translated MIDI keyboard, yet sound musically correct while playing any desired musical structure in any key! The structure/table control changes can be simply coded into a MIDI sequence so that the user can play along with any piece of music (MIDI files or CD tracks), no matter how complex, without sounding out of key. The player still makes the creative choices that define rhythm, voicings, and dynamics! By not having to worry about "wrong notes," a player is freed to feel the music and explore in ways that would be impossible using traditional instruments and techniques.

Some of the benefits of working with Hotz technology.

- Real-Time Super-Human Music Performance. Greatly enhanced functionality of traditional MIDI controllers and/or computer keyboards.
- Perform complex music much sooner than is typically possible and perform passages that are otherwise physically impossible..
- Immediate and practical real-time access to an immense vocabulary of chord/scale structures. Includes the most advanced musical structure knowledge base available, this empowers and inspires players to explore.
- User-expandable knowledge base, important for development, research, education, etc.
- Accelerated understanding of connections between structures, styles of music, cultures...
- Assists original and authentic expression in every conceivable style and mood of music. Simplifies the exploration and understanding of masterful compositions.
- Upward path: There is no dead end street! Hotz technology can take you as far as you wish to go as a musician. The Hotz Translator was designed from the ground-up to let the worlds most serious professional musicians explore new musical realms and perform musical feats in real-time that were simply not possible without Hotz technology.

Chapter 2: Overview

Hotz Translator Database



The Hotz Translator contains one of the most complete databases of musical structures available anywhere in a single reference. The accumulated expert knowledge base of musical structures from around the world, developed over thousands of years, is embodied in the Translator engine's database. It contains many unique structures that have not been previously classified. Many of these new and unique structures playable on a Hotz Translator instrument are physically impossible for one person to play on conventional instruments. We humbly observe that it is probably not possible for a human without computer assistance to memorize or mentally access a musical database even remotely this advanced, let alone have the training and dexterity to flawlessly perform all of these structures in real-time. The Translator database provides every possible voicing of thousands of culturally cross-referenced structures, available in any key, with any altered root notes and in any mode. [As a point of reference around 50 different chord and scale structures can be used to play the majority of music produced in Western culture.] This means that on as little as fifteen pads or input keys literally billions of voicing possibilities are available. The incredible power of having all conceivable voicings of a chord available simultaneously is extremely satisfying. A novice can access and perform with a vocabulary of musical structures that is very advanced the first time they play the Translator.

Database organization

Because of the vast size of the database a mechanism was devised to help organize and easily locate the various structures. Each of the Structures are assigned several family classes, including: **Type** (Minor 9th. chord, Dorian Scale, Harmonic Minor scale...), **Style** (Jazz, Blues, Classical...), **Geographic Origin** (Japanese, African, European, Indian...), and **Properties** (majestic, ominous, fierce, erotic...) may be associated with a structure. This system gives the user at least four different paths to a desired result, thus facilitating the process of composition, education, and exploration.

Note: Advanced users can edit and/or add new structures to the vast library already assembled.

Chapter 2: Overview

LIST OF DATABASE FAMILIES

TYPE

MAJOR & Alterations
  Major 6th & Alterations
  Major 7th & Alterations
  Major 9th & Alterations
  Major 11th & Alterations
  Major 13th & Alterations
MINOR & Alterations
  Minor 6th & Alterations
  Minor 7th & Alterations
  Minor 9th & Alterations
  Minor 11th & Alterations
  Minor 13th & Alterations
DOMINANT 7th & Alterations
  Dominant 9th & Alterations
  Dominant 11th & Alterations
  Dominant 13th & Alterations
AUGMENTED & Alterations
DIMINISHED & Alterations
SUSPENDED & Alterations
FIVE & Alterations
Intervals
Climbs
Traditional Scale Modes
Pentatonic Scales & Modes
Alternative Structures
Compound Structures
Large Structures
Chromatic

STYLE

African
Avant-garde
Blues / R&B
Celtic / Irish
Classical / Western
Ethereal
Folk / Country
Indian
Island
Jazz
Latin
Middle Eastern
Other Styles
Oriental
Rock /Pop

GEOGRAPHIC ORIGIN

Africa
America (North)
America (South & Central)
Antarctica
Arctic
Asia (East & Central)
Asia (North)
Asia (South / India)
Asia (Southeast)
Asia (Southwest)
Asia (Western Asia)
Atlantic (Islands)
Australia / New Zealand
Caribbean (Islands)
Europe
Indian Ocean (Islands)
Other Geographical
Pacific (Islands)

PROPERTIES

Note: "Properties" are not yet fully implemented in the database library.

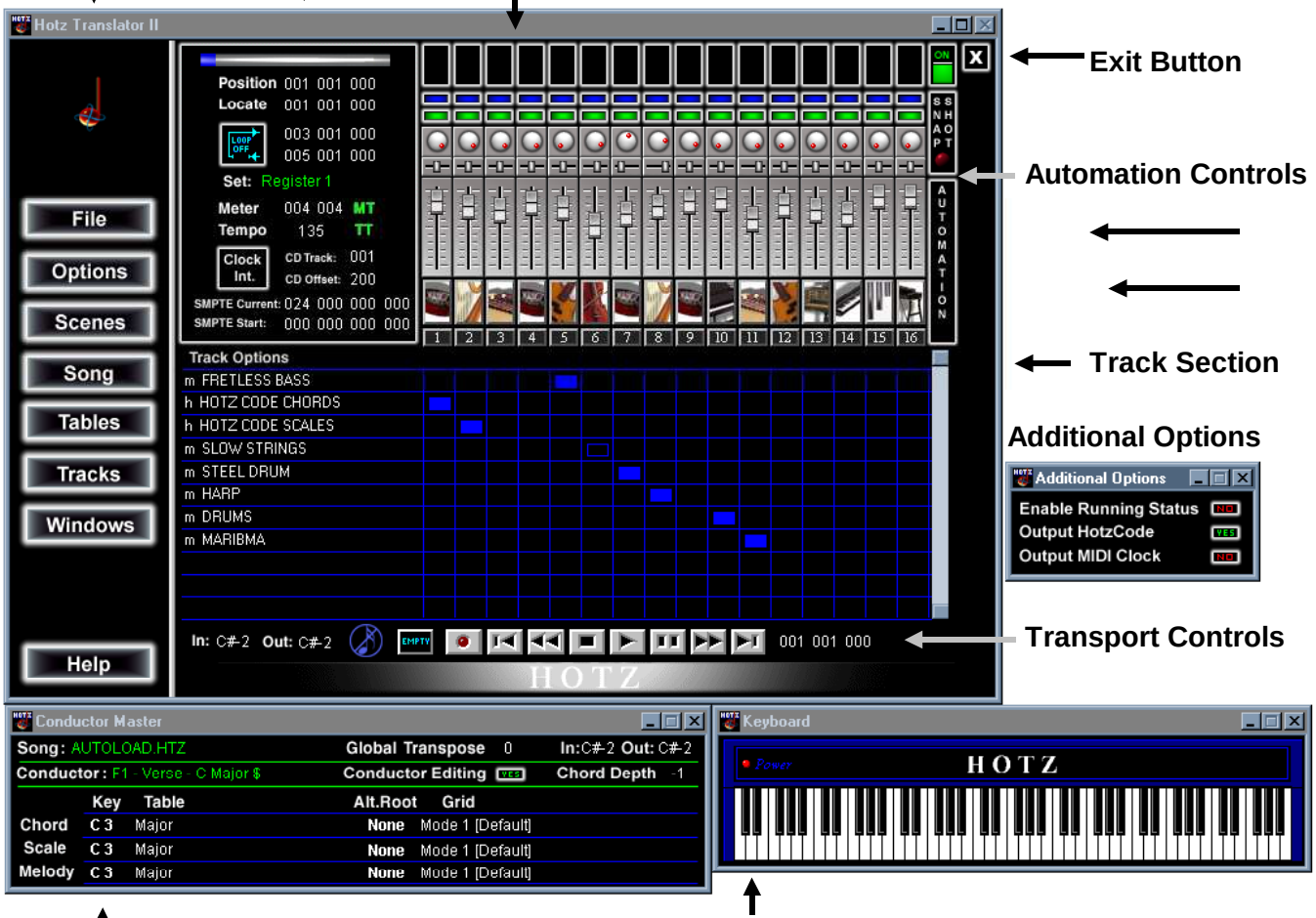
Anger
Anxiety
Beauty
Compassion
Confusion
Cosmic
Darkness
Depression
Discomfort
Erotic
Fierce
Finality
Flowing
Forceful
Forgiving
Gentle
Heavy
Heroic
Intricate
Light
Lonely
Loss
Lost
Love
Joyous
Madness
Magical
Majestic
Meditative
Motion
Mournful
Mysterious
Ominous
Painful
Peaceful
Playful
Powerful
Purity
Reflective
Restless
Romantic
Sensuous
Simple
Solemn
Sorrowful
Stressful
Sweet
Tenderness
Triumphant
Ugliness
Uncertainty
Vigor
Warlike
Weakness
Wonder
Wrath

Chapter 3: The Hotz Translator Environment

The **Hotz Translator** software loads with a default window arrangement similar to that shown below.

The Main Window is divided into:

- A **Menu Button** Section
- An area for **Location** and **Transport** Settings. These parameters are saved with each song file.
- A **Mixer** Section



Conductor Window

Keyboard Window – Displays the current activity of the translated input.

Chapter 4: Conductor Window – Detail

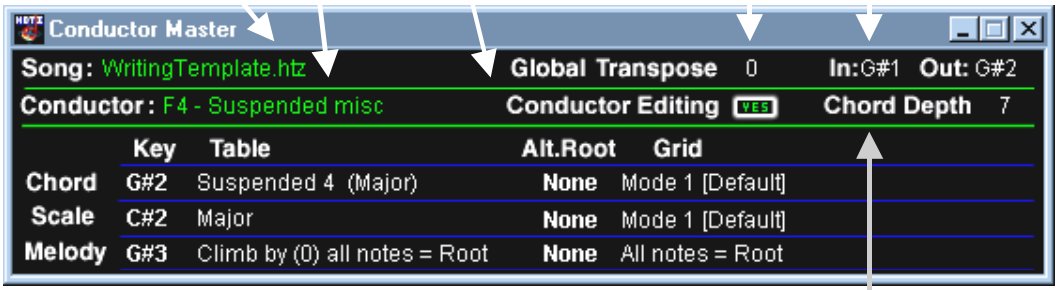
Song
Displays the Name of the Current Song

Conductor
Displays the Name of the Current Conductor. Left-click on the name to access a Conductor Bank selector.

Global Transpose
Displays Global Transpose changes. Left-click and drag to change value.

Conductor Editing
Allows changes made in this Window to update the master Conductor Cue List.

In: Out:
Original Incoming and Translated output Values of the last Translated Note event.

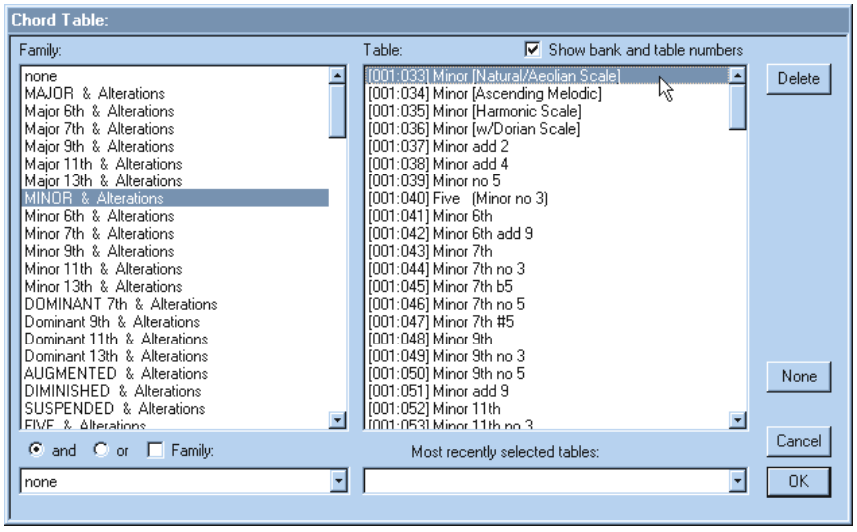


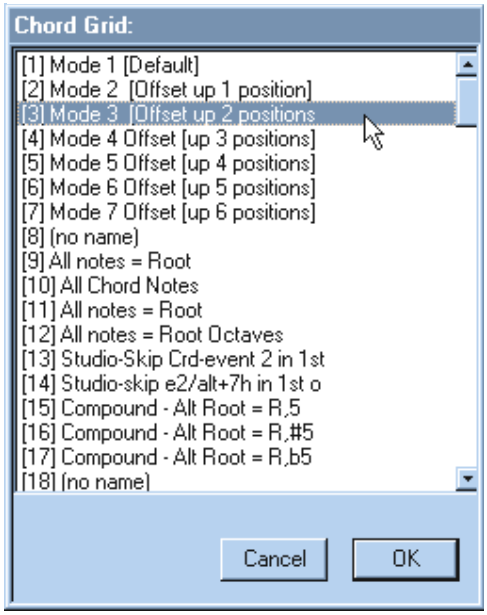
Key
Edit for each table type by clicking and dragging

Table
Click on a table name to bring up the table selector.

Alt. Root
Edit by clicking and dragging

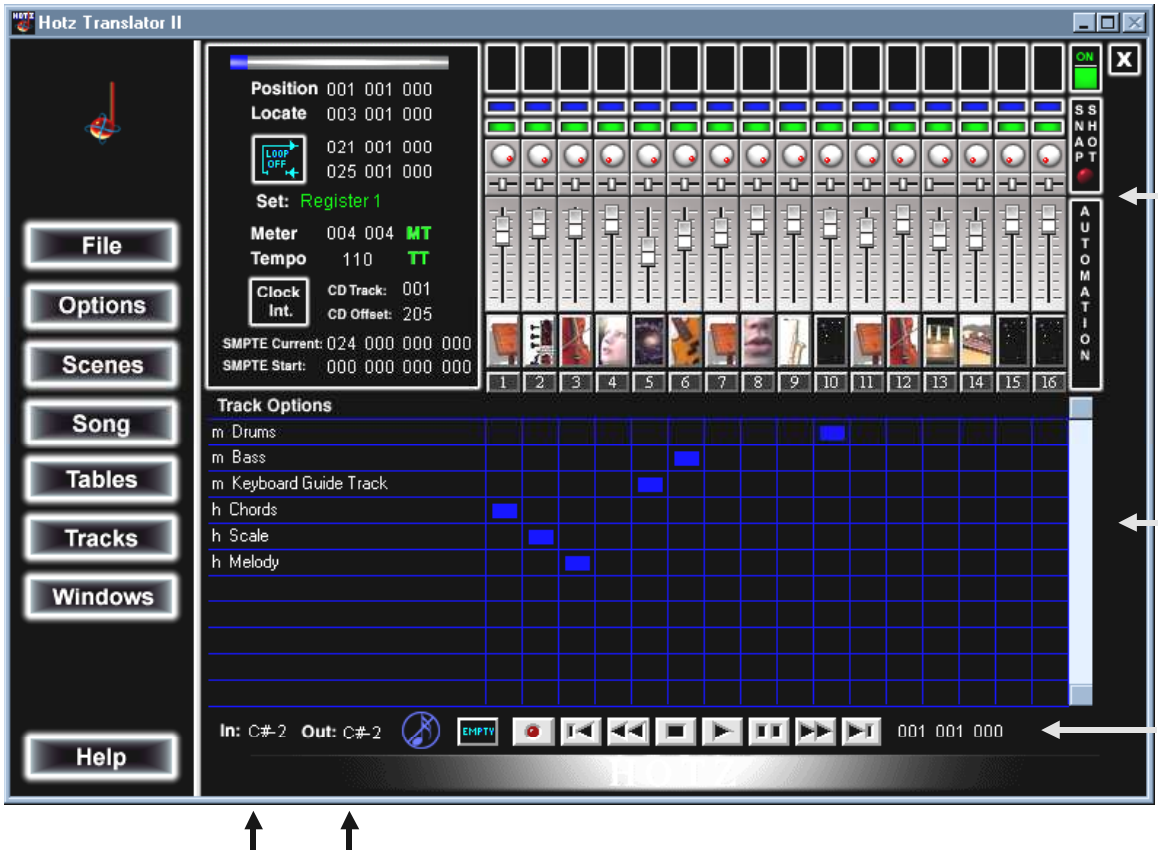
Chord Depth
Displays and allows editing of the number of notes used to generate the Conductor Chords (note arrays). A value of (-1) means use the values stored in the Cue List. (0); means do not output notes. All other values are shown as actual array depth.





Chapter 5: Main Window – Detail

- **Menu Buttons**
- **Location** and **Transport** Settings
- **Mixer** Section



Exit Button

Automation Controls

Track Section

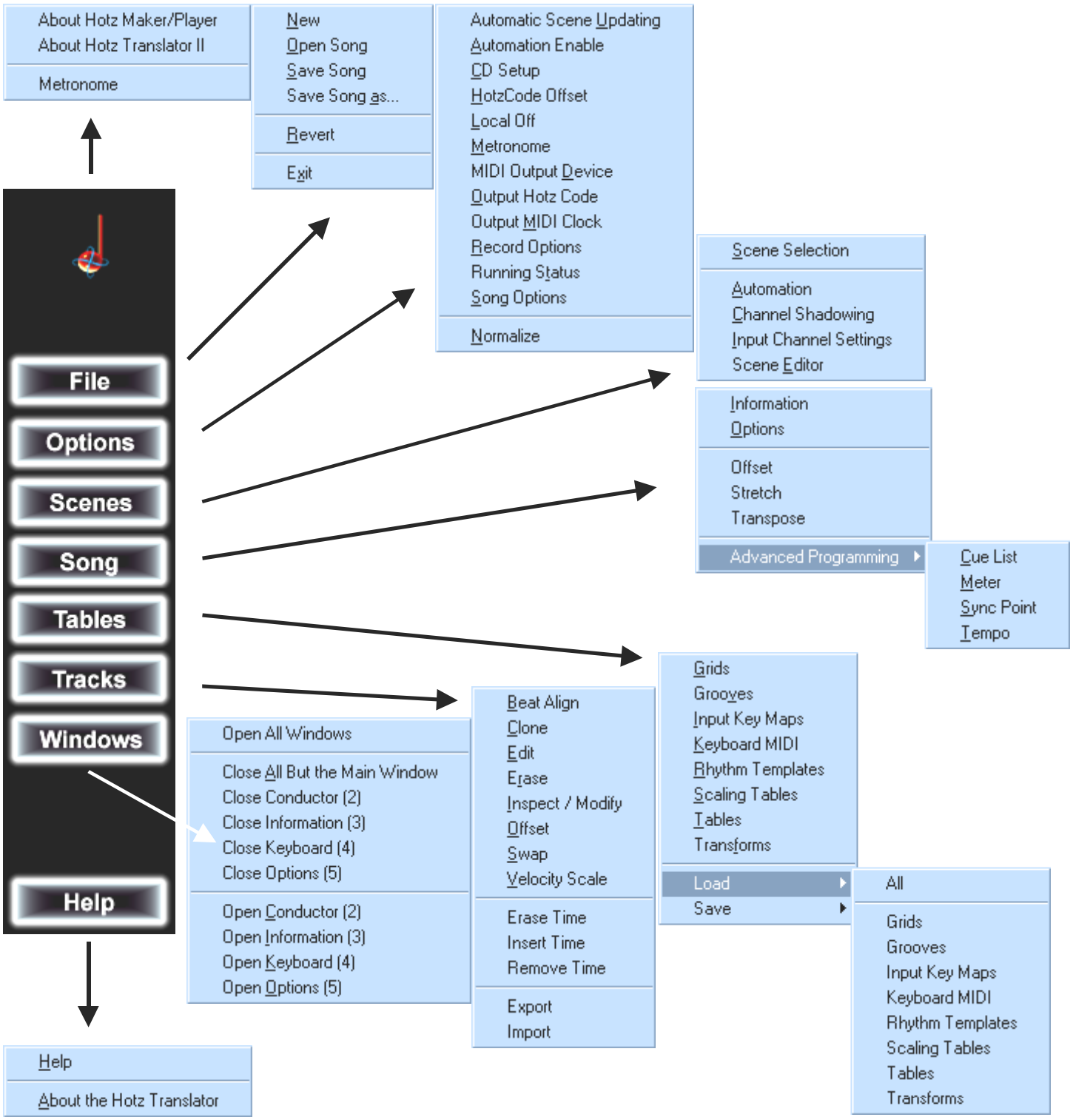
Transport Controls

Last Translated Note Data
(**In:**) Displays the value of the last incoming note event (**Out:**). Displays the value it was **Translated** to.

Chapter 5: Main Window - Detail

Menu Buttons

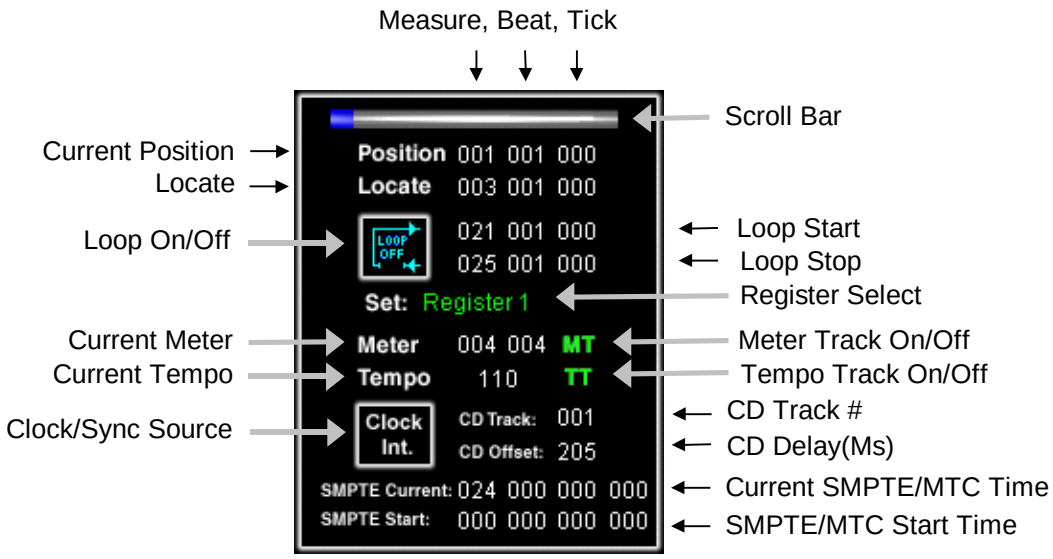
Left-clicking on any of the items below brings up its' corresponding pop-up menu.



Chapter 5: Main Window - Detail

Location and Transport Settings

On most of the number items below, left-click and drag to edit in fast steps, single left-click to decrement by 1 and single right-click to increment by 1.



Scroll Bar

The Song Location **Scroll Bar** is used for quickly scrolling to a new location, and can be used while transport is active. Left-click and drag to move.

Position

The **Position** indicator shows the current location within the song in terms of Measure number, Beat number, and tick number.

Locate

Locate can be used to automatically start anywhere within the song when **Play** is activated. **Locate** is stored in the **Registers**.

Loop

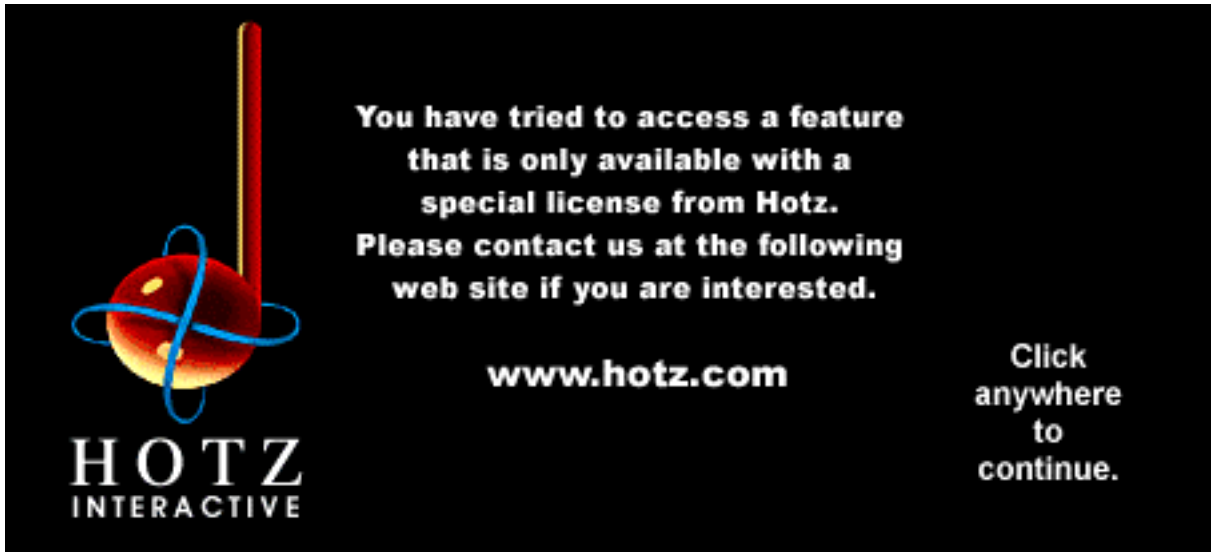
Click on the **Loop** button to activate the global song loop. Set the **Start** and **End** point for looping in the same way as all other number values. The song will loop within the set parameters until transport is stopped. This feature is extremely useful for recording patterns, sections, or punching in. Locate and Loop values are stored in the **Registers**.

Chapter 5: Main Window - Detail

Clock/Sync Source

Clock/Sync source is used to set the clock to **Internal**, **MIDI Clock**, **CD**, or **Time Code - MTC**. Click on the icon to change the clock source, then left-click on your selection. **Please note: In the standard release of the Hotz Translator certain CD features are not available. You can still play along with pre-coded CD's, but you will not be able to adjust CD parameters.**

The following message will appear when one of these features is selected.



CD Track Number and CD Delay/Offset

CD Track Number and **CD Offset** are displayed next to the sync icon. Click on either number to set desired value. When coding a CD song, set the song's track number and then the CD Offset. The default delay value for CD Offset is 200 milliseconds, which you will need to customize for each CD track.

Note for CD Coders: Make sure that your systems' global CD sync is set up and working correctly before setting the CD offset and HotzCoding a CD. Users considering Hotz-coding CD's should contact Hotz Corp to obtain a license.

The Translator may be slave synced to another external sequencer that is capable of outputting MIDI clock with bar pointers by selecting MIDI Clock as the Clock source. Output MIDI Clock in the Options Menu allows external sequencers to sync to the Translator.

Registers

The **Registers** feature allows you to divide your song or tracks into sections you can then access more efficiently. Select the **Register** you want and then set the **Locate** as well as the **Loop** start and end. The values for each of these are stored with the register and the registers are stored in the song file.

Tempo

Set the base **Tempo** by clicking on the tempo number area and dragging up to increase value, down to decrease value; or click to select then type in desired value. *Note: Tempo Map and Meter Map features are implemented (see Song Commands section for further information). It is possible to create a tempo track by playing a reference tap-tempo track to any musical performance. This feature allows users to sync to previous recordings that may have many tempo variations.*

Chapter 5: Main Window - Detail

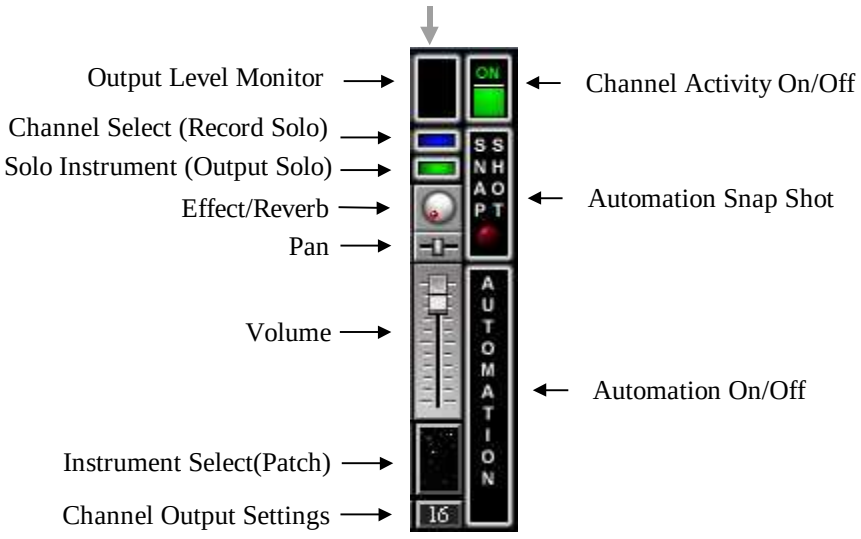
Mixer Section

The Main window includes a **Mixer Section** with controls for activity, automation, channel setting, pan, patch, reverb, solo, and volume.

The figure above shows output activity on several channels. Notice the various pan, volume, and aux/reverb settings. Play your computer keyboard or MIDI keyboard and you should see activity in the monitor areas. The best way to understand the **Mixer** (or any other Translator window), is to play with the controls until you can predict the results of specific actions.

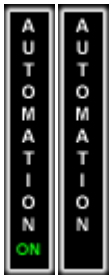
*It is important to realize that a **Mixer Channel** is not to be confused with a MIDI channel*. Although the software defaults to all Mixer channel assignments corresponding with MIDI channels, it is easy to re-route a Mixer channel's output to any MIDI channel. However, beginning users should avoid confusing situations by leaving the Mixer channel outputs at default values.

Mixer Channel 16



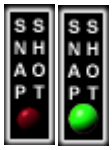
Chapter 5: Main Window - Detail

Automation On/Off



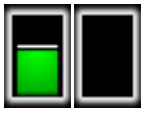
Left-click to enable or disable **Automation**. Automation data is Volume, Pan, Patch and Effect data that is stored in tracks. When enabled, any automation data will override the current scene data of the same type. You may also create automation data by moving any of the Mixer controls mentioned as Automation data types, or you may use the Automation Snapshot to take a picture of the current state of the mixer and store the data onto a track. The Automation On/Off state is stored with each scene as well as any cell mute states of any automation tracks.

Automation Snapshot



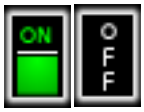
Left-click to take a snapshot of all current mixer settings and record them to a track. Automation data includes Volume, Pan, Patch and Effect. For **Automation Snapshot** to function, **Automation On/Off** must be set to on. The recording that is generated will appear as the last track.

Channel Activity Monitor



This shows the output activity on each channel.

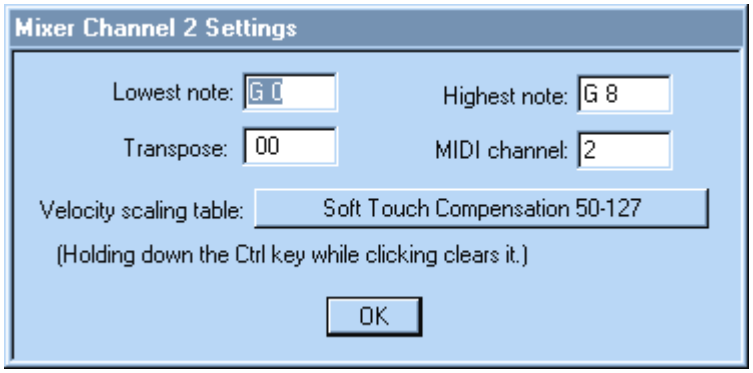
Channel Activity On/Off



Left-click to enable or disable channel activity. On slower systems, turning channel activity off can help improve graphic response.

Channel Output Settings

Click on a **Mixer Channel Number** icon to set parameters for the selected Mixer channel and its associated Tracks.



Chapter 5: Main Window - Detail

When the **Channel Settings'** window opens you may edit the following:

Lowest Note

Lowest Note is used to filter out notes that would be too low for a particular sound.

Highest Note

Highest Note is used to filter out notes that would be too high for a particular sound.

Transpose

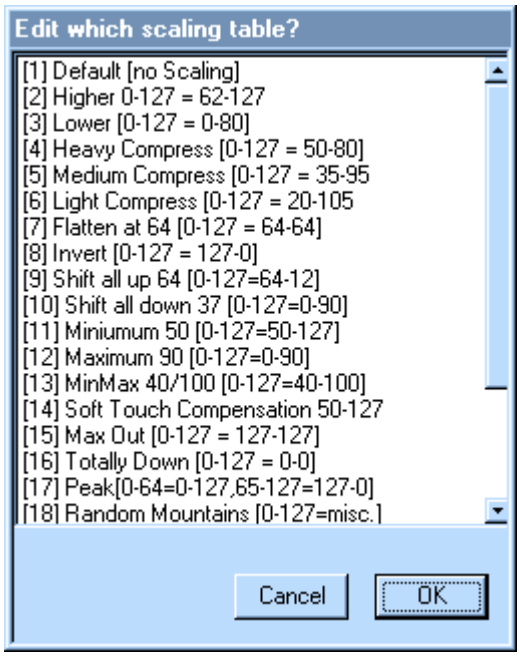
Transpose in semitones. This is usually set at octave increments of 12. Example: -24, -12, 0, 12, 24.

MIDI output channel

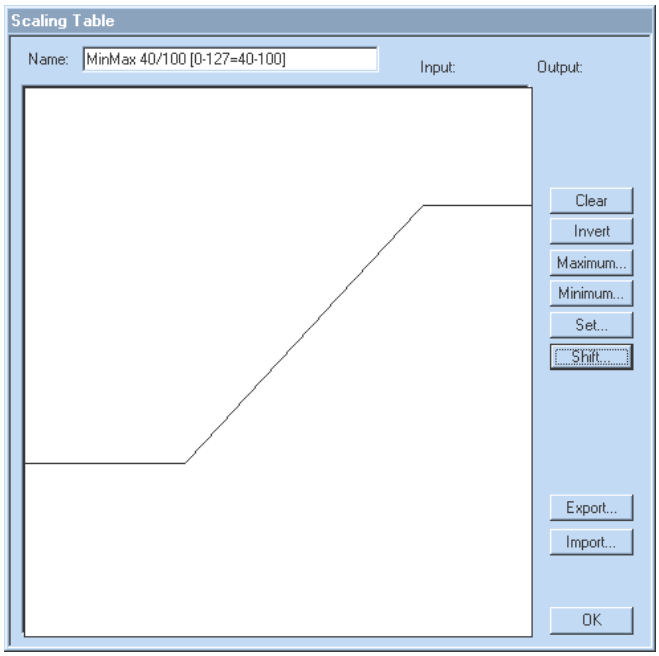
MIDI output channel for the selected Mixer channel and its' associated Tracks. Normally this should be the same as the mixer channel.

Velocity Scaling Table

Velocity Scaling Table (which allows you to customize each incoming channel's velocity mapping table—see *Advanced Features* section). Select a **Velocity** table by clicking on the icon and choosing one of the available tables from the pop-up menu. Note that a Velocity table assignment may be cleared by pressing the Control key while clicking on the icon.



Scaling Table Example



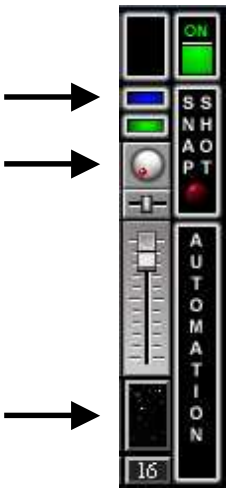
Chapter 5: Main Window - Detail

Channel Select (Input/Record Solo)

The **Channel Select (Record Solo) OFF/ON** switch is one of the most important controls in the **Mixer** window. Click on the blue switch to activate Solo Recording; it will become Red, indicating that all input data on all channels is being routed to that channel during recording. The **Record Enable** switch on the Transport must be on (**red**) for any recording to take place.

Effect/Reverb

Effect/Reverb knobs are used to add **Reverb** to mixer channels.

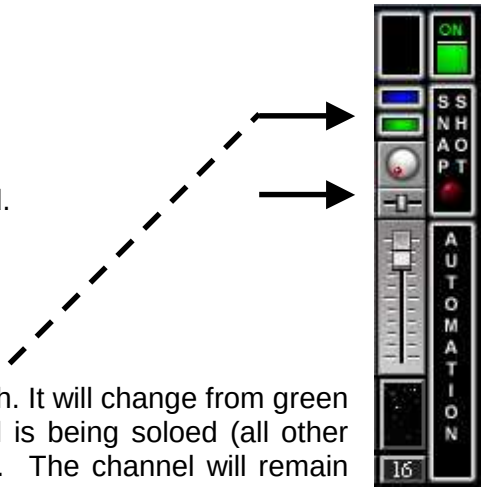


Instrument Select (Patch)

Click on the instrument **Patch** icon to change the instrument assignment for a particular **Mixer** channel. A list of General MIDI instrument assignments pops up. Select a new sound by double-clicking on it or clicking once on it and selecting OK. Those who do not have General MIDI compatible modules should refer to their instrument's sound list and choose the corresponding number from the pop-up list. The number is the program change that will be sent to the sound module. **Note: Some synthesizers list their sounds with 0 as the first sound location. Therefore, you may need to subtract one from all numbers you choose from the Translator so the two lists correlate properly.**

Pan

PAN is used to determine the left-right placement in the stereo field.

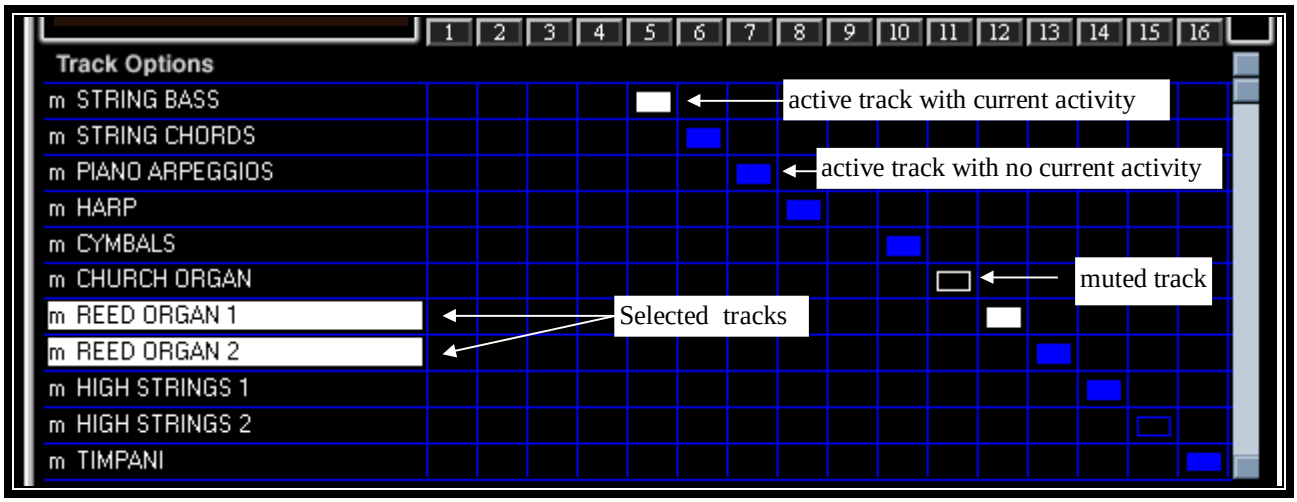


Solo Instrument

To solo a channel, left-click on the **Solo Instrument OFF/ON** switch. It will change from green to a white "solo" label, indicating that only that particular channel is being soloed (all other tracks are temporarily muted, excepting any other soloed tracks). The channel will remain soloed until it is left-clicked again.

Chapter 5: Main Window - Detail

Track Grid



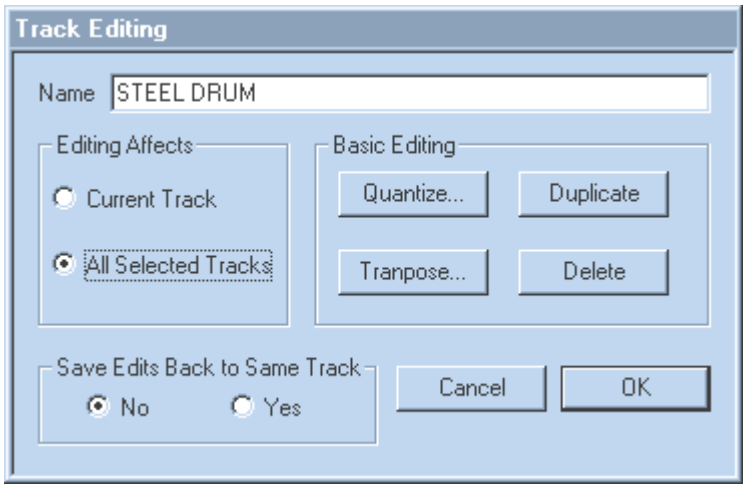
The **Track Grid** window represents each MIDI channel as a vertical column and each track as a horizontal row on a grid.

Basic Editing

One or more tracks must be selected on the Track window for most of the commands to function.

For additional **Track Editing** options refer to the **Track Menu** which provides additional commands for manipulating track data. (See the **Track Menu** section.)

Basic editing can be done by double-clicking on the Track Name.



Delete

Removes the track data from memory.

Chapter 5: Main Window - Detail

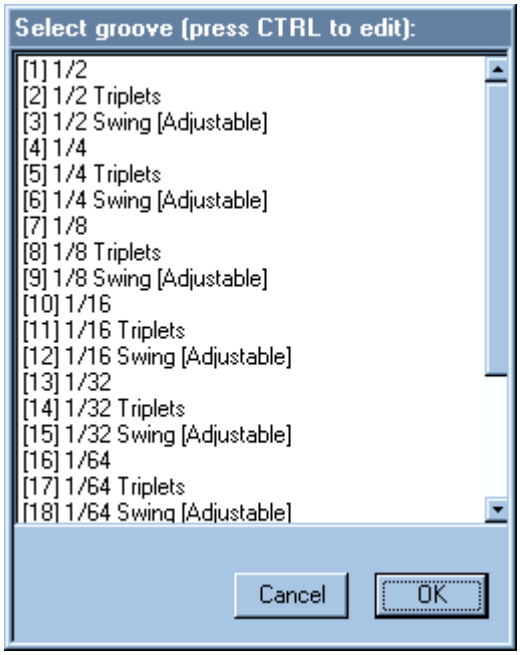
Duplicate

Copies the selected track's data to a duplicate track. The new track has the same name and a Duplicate suffix (it is appended to the end of the Track list, and the original track is automatically muted).

Editing Affects – Option

Either the current track or All Selected tracks may be affected. Simply click on your selection.

Quantize



This presents a list of **Groove/Quantize** tables. Either double-click on a **Groove** or select one and press O.K.

These may be edited by pressing "Control" while clicking on the groove name, this will open the **Groove** window in which the Quantize parameters are defined. Refer to **Grooves** in the **Tables Menu.**

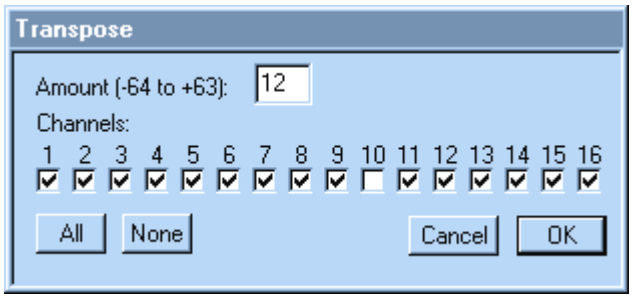
Save Edits Back to the Same Track – Option

Selecting Yes will overwrite the current tracks with the edited data. Selecting No will mute the current tracks and save the edited data to new tracks.

Track Name

Track names are automatically generated when the tracks are created. They may be edited by double-clicking on the tracks' name and typing in the name field.

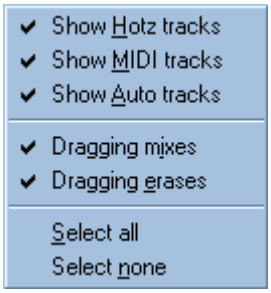
Transpose (Track)



Transpose (Tracks) is used to transpose all data on selected tracks. In semitones or half-steps (12 semitones equal an octave). A checkmark under a channel number indicates that all data on that channel will be transposed. **All** places a checkmark under all of the channels. **None** removes all of the checkmarks that are currently selected.

Chapter 5: Main Window - Detail

Track Options

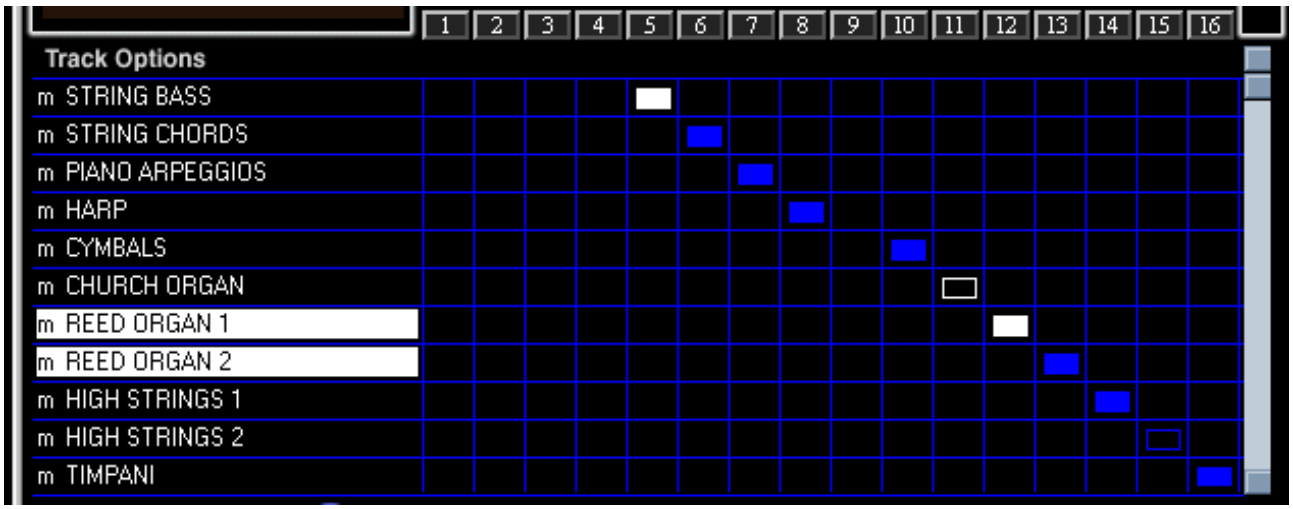


The **Tracks** window can display three types of tracks: standard MIDI data, automation data, and **HotzCode** data (Hotz code is recorded automatically when Hotz Conductor events occur during recording). This separation of track types allows you to manipulate HotzCode data separately from all other MIDI data. Select the desired **Track View** by left-clicking on "Track Options" in upper left corner of the "Track area".

Select All and **Select None** are used for selecting/de-selecting all tracks quickly (**All** are selected or **None** are selected).

Left-click on a track name to select a track for editing with any of the Tracks commands (delete, quantize, duplicate, offset, etc). Selected tracks are highlighted in white and are the tracks that will be affected by various editing actions.

Track names can be edited by double-clicking on the desired track name. Names can be up to 30 characters long. Whenever a new track is recorded, a name containing the date and time of the recording is inserted automatically. Copying and editing actions are also automatically noted in the track name. An asterisk next to a track name means that track has been modified in some way (either copied over by another track, mixed with another track, or otherwise altered by a Track Menu command) since the song was last saved.

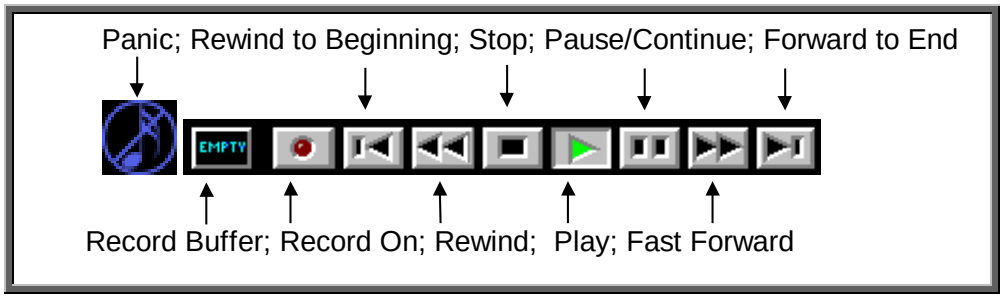


As can be seen above, track data is shown as small cells that correspond to mixer channels. For example, the String Bass track is active on mixer channel 5, while the Church Organ track is muted on mixer channel 11. Active data cells are solid blue and flash white when they output MIDI note data. Muted cells are shown as unfilled rectangles. Right-click on a cell to toggle its' mute state. Right-clicking on a track name will toggle the mute state of all cells on that track.

When the **Dragging Mixes** box is checked, left-clicking on a cell or track and dragging that cell/track to another location will move the cell/track to the new location and mix the data from both locations. If the **Mixes** box is not checked, left-clicking on a cell/track and dragging that cell/track to another location will move the cell/track to that location and replace whatever was originally there. Pressing **Control** while left-clicking and dragging a cell/track will place a copy of the cell/track in the new location. Channel columns can also be selected for move/copy actions by left-clicking just below the mixer channel number and dragging to another column. When the **Dragging Erases** box is checked, dragging a cell or track name off the grid deletes the data of that cell/track. *This is the quickest way to delete unwanted data.*

Chapter 5: Main Window - Detail

TRANSPORT CONTROLS



Fast Forward/Forward to End

Fast Forward moves the song forward in time and selects pause when released. Forward to End jumps immediately to the end of the song.

Panic(Stop All Notes)

Panic(Stop All Notes) sends its message on all MIDI channels (use it if you experience stuck notes). This control works whether Play is active or not.

Pause/Continue

Pause/Continue stops the transport during recording or playback. Clicking on it again continues the transport from the current song location.

Play

Play will start playing from the **Locate** position.

Record Enable

Record can be activated during playback or before **Play** starts the transport. This is a global record control and must be active for any recording to take place. (see **Record Options** under the **Options** menu to determine what is recorded and **Channel Select (Record Solo)** under the **Mixer** section to determine where it is recorded).

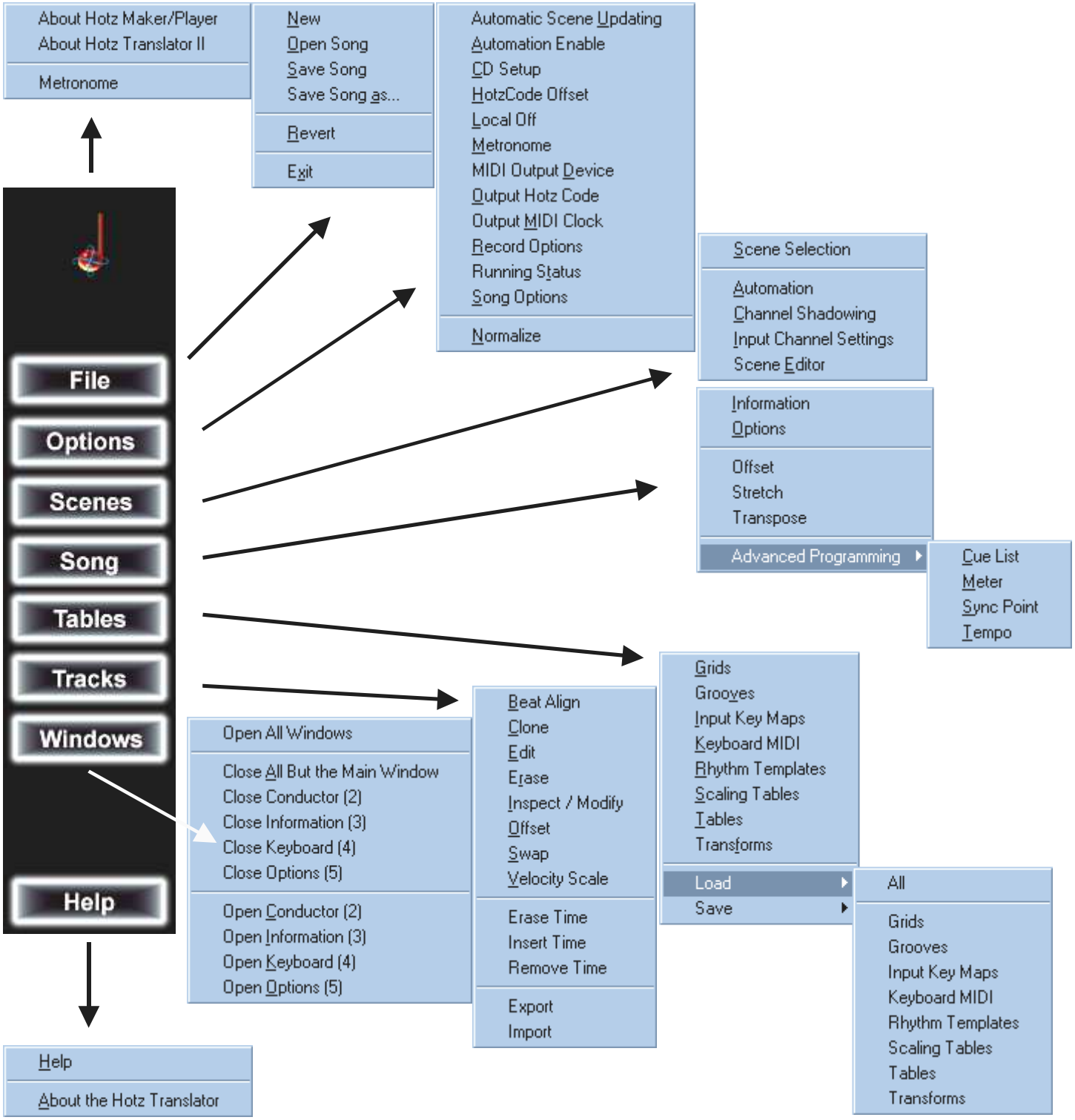
Record Buffer

Record Buffer allows the user to discard a take before recording is stopped and the data written to a track. Click on it before you stop the transport, to prevent the recorded data from being written to a new track.

Rewind/Rewind to Beginning

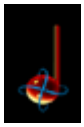
Rewind moves the song backward in time and selects pause when released. **Rewind to Beginning** jumps immediately to the beginning of the song.

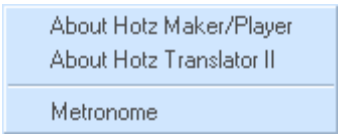
Chapter 6: Menu Buttons



Chapter 6: Menu Buttons

About the Translator / Metronome



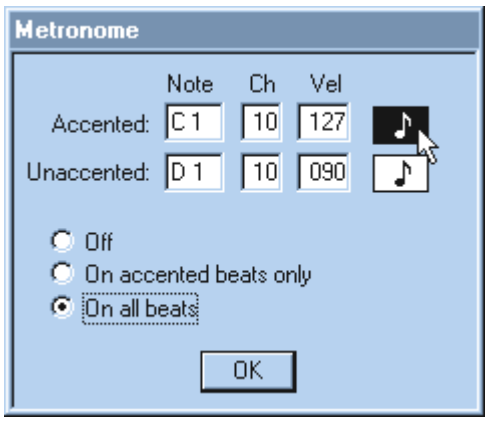


The Hotz icon serves as a visual metronome during play. Left-clicking on it opens a pop-up menu with Metronome as an option. Selecting this option opens the Metronome settings window, which provides note, channel and velocity options for the metronome events

Selecting **About Hotz Translator II** opens the following screen.

Hotz Translator II Information Screen

Metronome

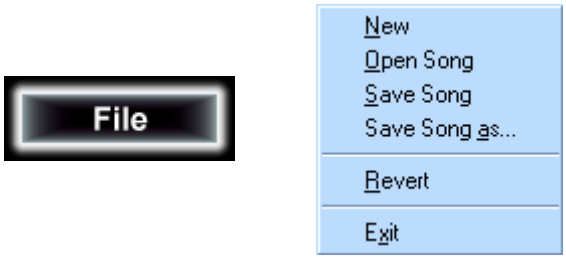


Click on the desired value box and either drag up/down, or type in a new value. For example, if D1 is selected, typing 4 will turn it into a D4 (note names can also be typed in). This convention is followed throughout the Translator architecture. **Off** turns the metronome off. If you want the assigned notes to sound on playback and recording, you can choose between **On all beats** or **On accented beats only.** Left-clicking on the note icon plays the note for testing.

Chapter 6: Menu Buttons

File Menu Commands

The **File** menu provides access to file loading, saving, and other song related information.



New

Clears the current song and loads the default new song file.

Open

Opens a song saved to disk.

Save

Saves the current song to disk.

Save As...

Saves the current song with a different name.

Revert

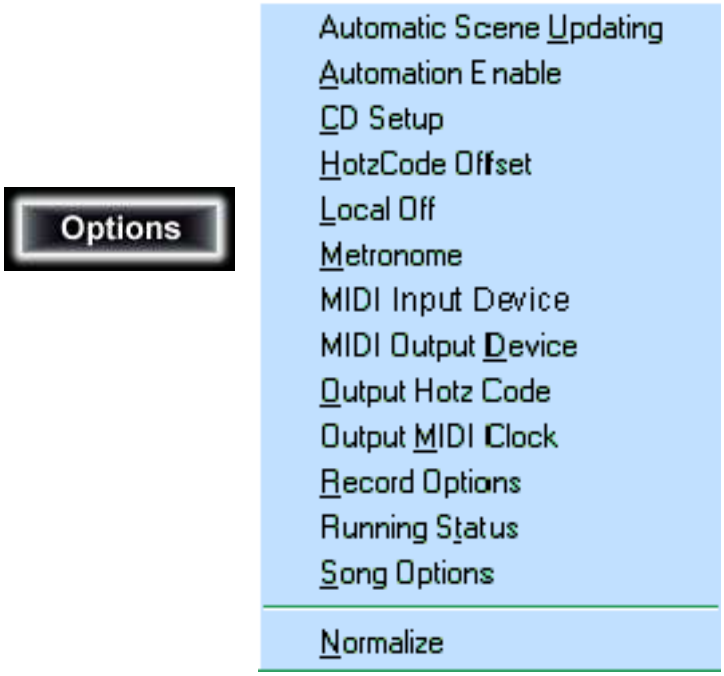
Returns the current song to its previously saved version.

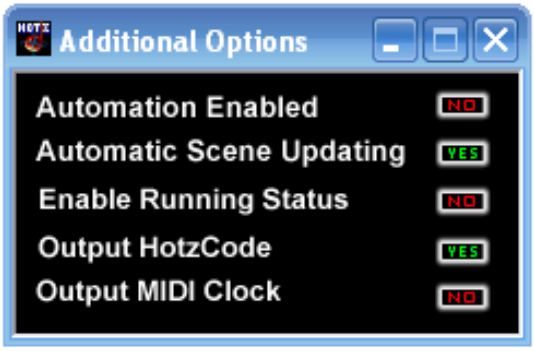
Exit

Quits the Translator program.

Chapter 6: Menu Buttons

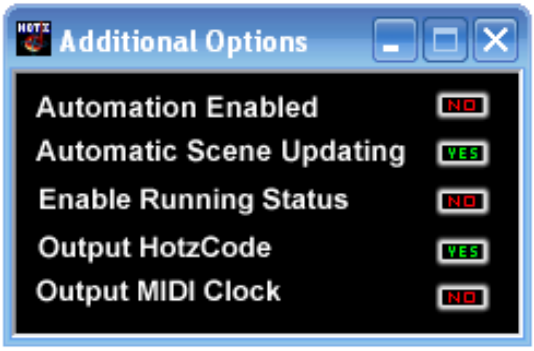
Options Menu Commands





Allow Scene Editing (Updating)

When **Allow Scene Editing (Updating)** is selected, all scene parameters (Volume, Pan, Patches, Track Mutes, etc.) are automatically updated.



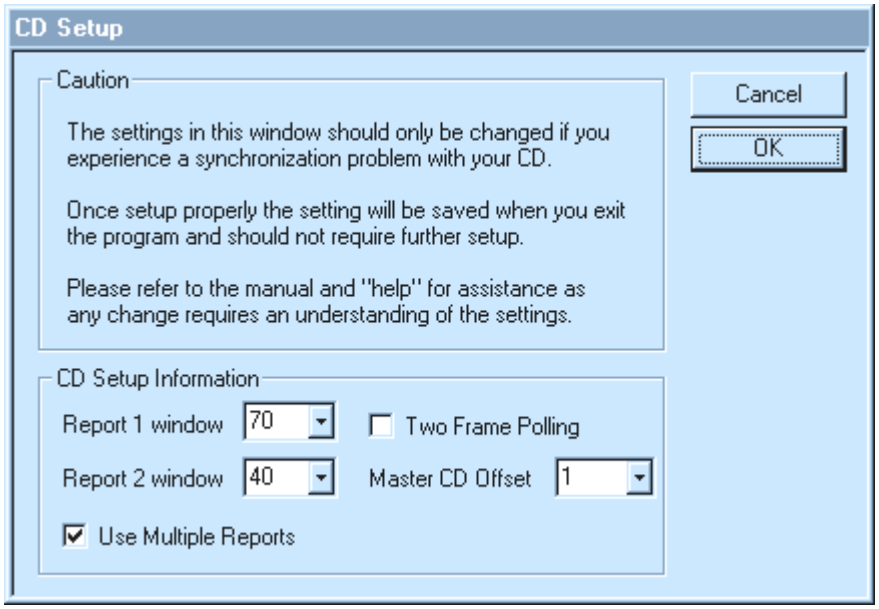
Use Automation (Automation Enable)

Use Automation (Automation Enable) may be enabled or disabled by either **Use Automation** in the **Setup Dialog** or the **Automation On/Off** switch in the **Mixer section.**

Chapter 6: Menu Buttons

CD Setup

These parameters do not usually need to be changed but are editable for unique situations.

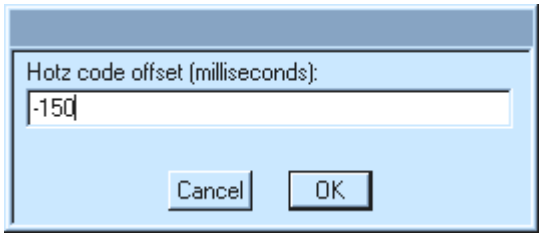


The **Master CD Offset** can be adjusted to ensure proper synchronization between the CD Rom drive and the Translator software. If you experience any sync difficulties, this parameter can help you fine-tune tracking problems. This is the master delay setting, and should only need to be set once for your particular hardware. These settings are saved in the Translator preferences when you quit the program.

This is an additional CD delay parameter that is added to or subtracted from the CD Offset in the Main Window.

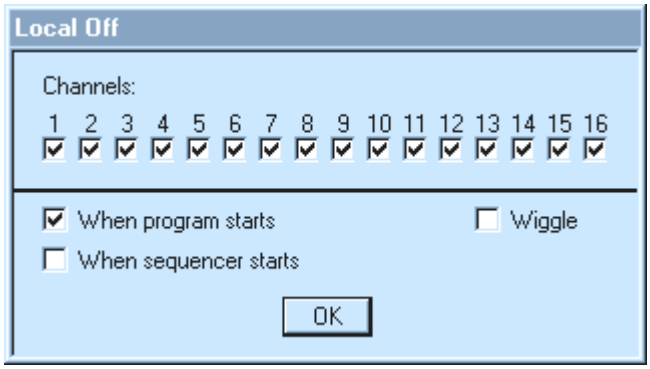
The other parameters in the **CD Setup Information** section are available for deep level support from Hotz and the description of their functions is beyond the scope of this manual.

Hotz Code Offset



Hotz Code Offset is the automatic offset value applied to all data on Hotz tracks. This pre-delay is necessary to ensure that the appropriate tables are already available when a chord/scale change takes place during a musical progression. The default value of -150 ms should be adequate for most applications.

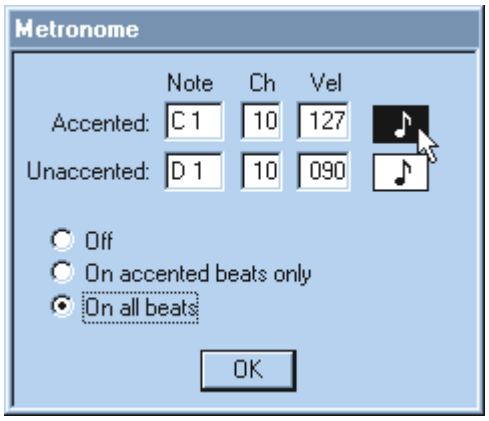
Local Off



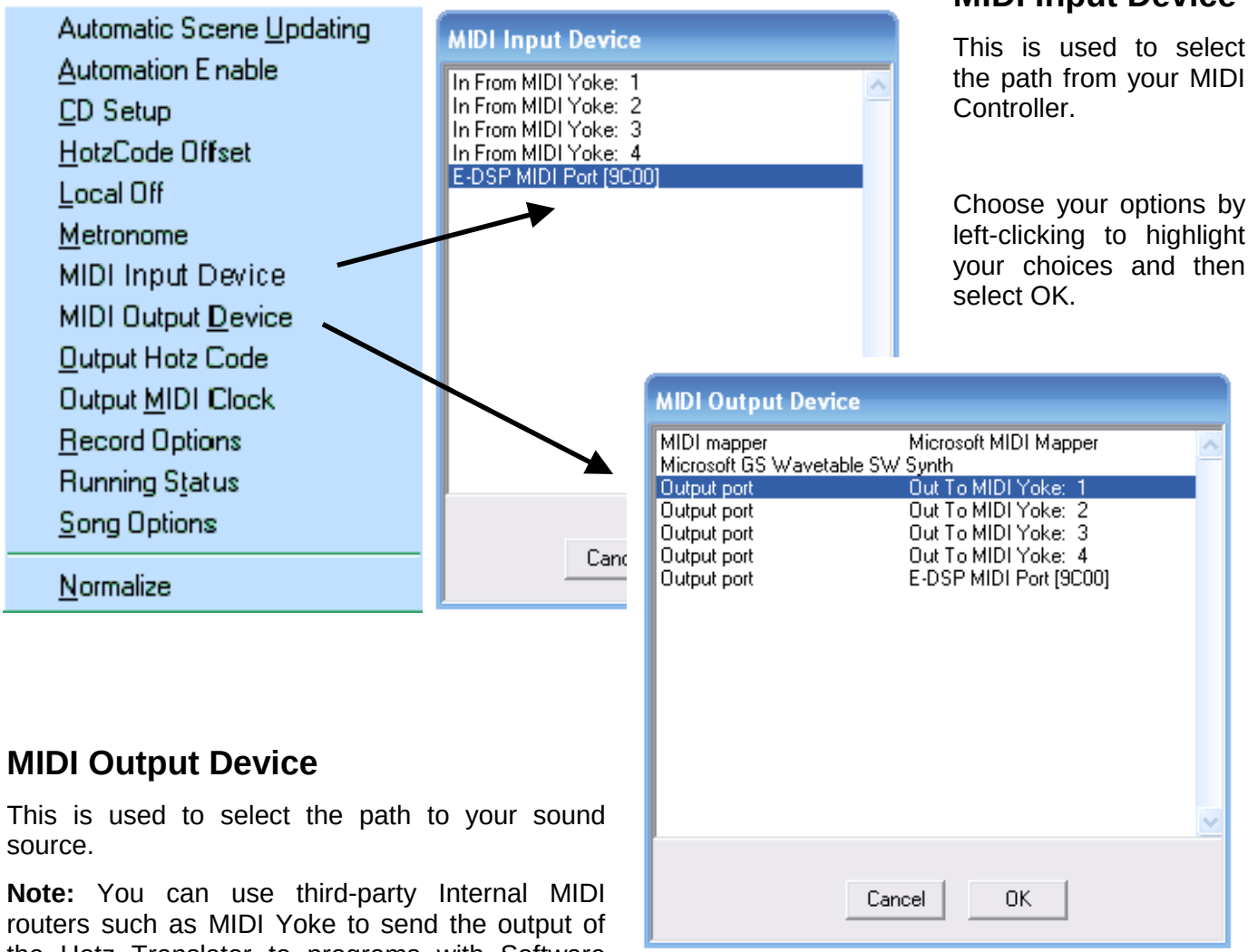
This command is used if you want to send remote Local Off controls to any MIDI instrument in your system. **Wiggle** sends a patch change after Local Off (some synthesizers, such as Oberheim Matrix, require this to turn Local Off).

Chapter 6: Menu Buttons

Metronome



Click on the desired value box and either drag up/down, or type in a new value. For example, if D1 is selected, typing 4 will turn it into a D4 (note names can also be typed in). This convention is followed throughout the Translator architecture. **Off** turns the metronome off. If you want the assigned notes to sound on playback and recording, you can choose between **On all beats** or **On accented beats only.** Left-clicking on the note icon plays the note for testing.



MIDI Input Device

This is used to select the path from your MIDI Controller.

Choose your options by left-clicking to highlight your choices and then select OK.

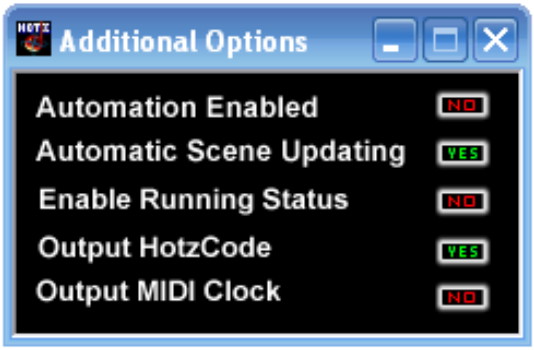
MIDI Output Device

This is used to select the path to your sound source.

Note: You can use third-party Internal MIDI routers such as MIDI Yoke to send the output of the Hotz Translator to programs with Software Sound Modules.

Caution! Be sure not to create MIDI Feedback loops when using Internal MIDI routers.

Chapter 6: Menu Buttons

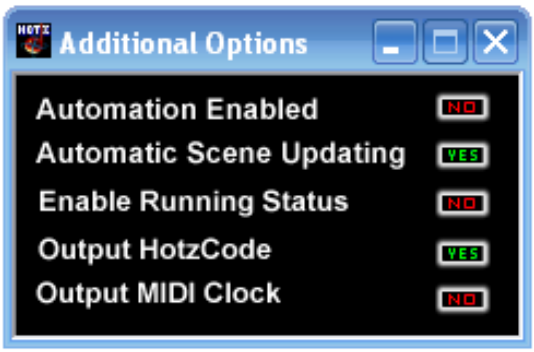


Output HotzCode

Click on this command to bring up the Additional Options window.

To enable/disable left-click on the **Output HotzCode** button.

Output HotzCode sends Hotz Chord and Scale data to other Hotz systems in real-time.



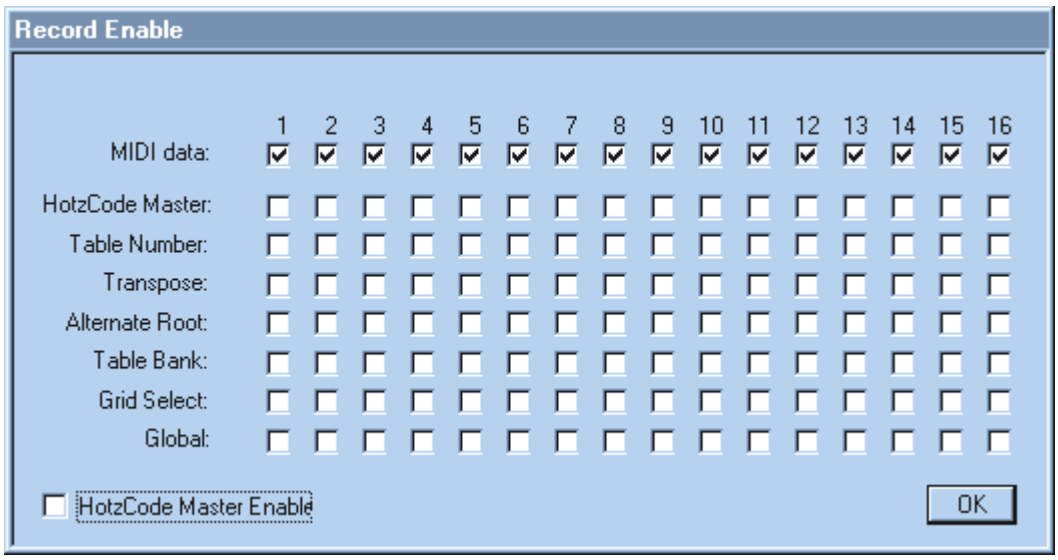
Output MIDI Clock

Click on this command to bring up the Additional Options window.

To enable/disable left-click on the **Output MIDI Clock** button.

Output MIDI Clock is used to drive other sequencers as slave units. When enabled, the program outputs a MIDI Start if starting at the **Sync Point**, otherwise a Song Position Pointer plus a MIDI Clock. Whether or not in CD Audio Sync, if **MIDI Clock Output** is enabled, locating to before the Sync Point is not possible. Also, the effective locate time is automatically backed up to the previous possible **Song Position Pointer** location (only every 16th note) if necessary.

Record Options

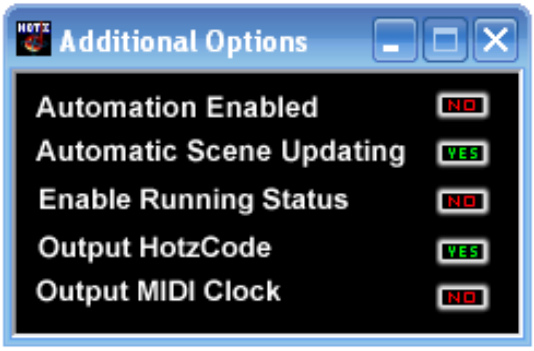


Record Options determine what is recorded when the Record button on the Transport is enabled.

Currently only the **Overall MIDI** data for each channel and the **HotzCode Master Enable** are selectable.

Checked boxes under **MIDI data** indicates which channels will be recorded.

HotzCode Master Enable should be enabled while recording HotzCode (Chord/Scale changes) and disabled when you do not wish to record HotzCode. This setting is saved in the song file.



Running Status

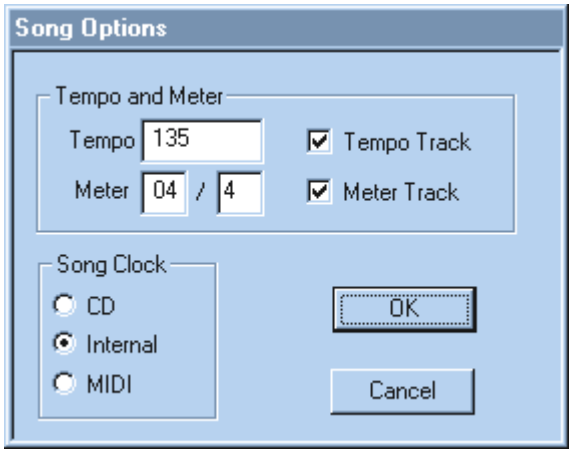
Click on this command to bring up the Additional Options window.

To enable/disable left-click on the **Enable Running Status** button.

When enabled, this command allows **Running Status** to be sent by the Translator software. This feature can make the MIDI stream up to 30% more efficient, but some MIDI devices have problems with Running Status.

Song Options

These Options are also available on the main screens Transport/Location section.



Tempo sets the speed of the song in "beats per minute". If Tempo Track is checked its values will override the current Tempo value. For details on the Tempo Track see Tempo in the Song Menu.

Meter sets the values used in calculating the length of a measure. The left value is the number of beats in the measure while the right number is the type of note the beats refer to. A Meter setting of ¾ would mean 3 quarter notes per bar. If Meter Track is checked its values will override the current Meter values. For details on the Meter Track see Meter in the Song Menu.

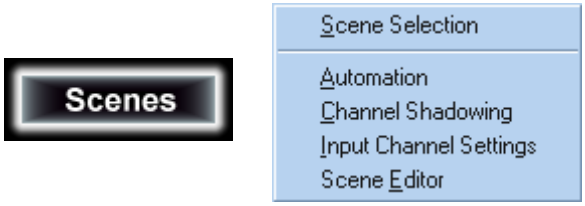
Song Clock selects the Clock/Sync source.

Normalize

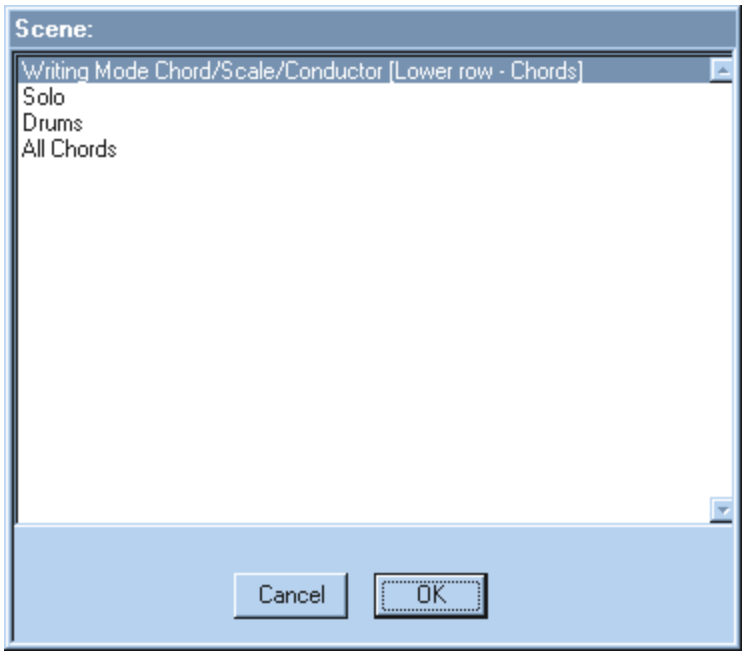
Sends out program change and volume data to any instruments in your MIDI system that have changed patches or volume for any reason, and you want them to be reset to the current song's Mixer volume and instrument settings.

Chapter 6: Menu Buttons

Scene Menu Commands

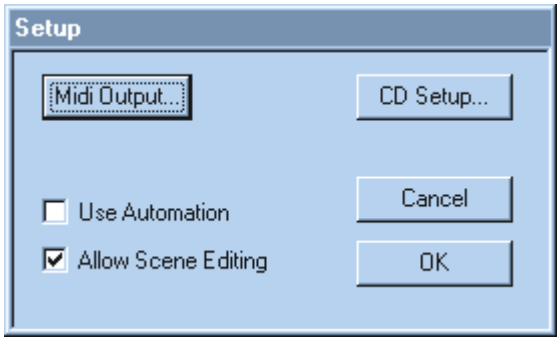


Scene Selection



To select a **Scene,** double-click on a **Scene's name** or left-click and press OK.

Automation On/Off

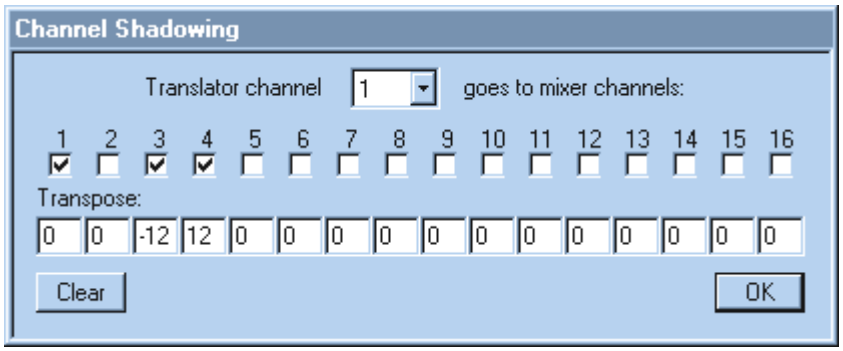


Left-click on **Use Automation** to enable or disable. Automation data is Volume, Pan, Patch and Effect data that is stored in tracks. When enabled, any automation data will override the current scene data of the same type. When **Automation** is enabled, you may create automation data by moving any of the Mixer controls mentioned as Automation data types. You may also use the Automation Snapshot to take a picture of the current state of the mixer and store the data onto a track. The Automation On/Off state is stored with each scene as well as any cell mute states of any automation tracks.

Chapter 6: Menu Buttons

Channel Shadowing

The software offers another processor at the output of the translation engine**. MIDI Channel Shadowing** allows all MIDI data that is output on each of the 16 Independent MIDI channels to be simultaneously Transposed and output on any combination of channels.



The **Translator channel** selects an output channel to setup the shadows for.

Clear sets the current **Translator channel** to a default non shadowed settings.

A **check** under a **channel number** means that data originally routed for output on the current **Translator channel** will be simultaneously output on any combination of the checked channels.

Transpose: Each Translator channel's output channels may be independently transposed as well.

Although these are normally set to octave values such as (-24,-12,0,12,24), they may be set to any value. Amazingly unique and unusual sounds may be created using multiple output channels with a unique combination of Transpose values.

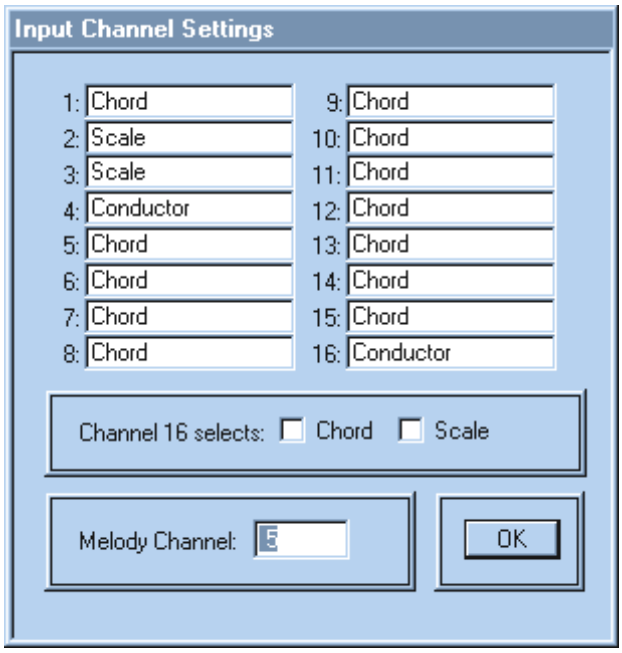
This process occurs directly before sequencer input and delivers the same data to the MIDI output as the sequencer receives. Internal MIDI channels passing from the translation processors can be easily mapped by the user so that all desired channels receive the MIDI data. Again, the software provides a very clear and intuitive graphical editor for this process.

Chapter 6: Menu Buttons

Input Channel Settings

The **Input Channel Settings** are used to determine which incoming MIDI channels will be assigned to which Translation processors. Each channel can be assigned to one of the five processors: Thru (Input=Output), Chord Table, Scale Table, Melody Table, or Conductor events which generate real-time Chord, Scale, Melody, Grid, and Cue List changes.

These settings are saved with each song.



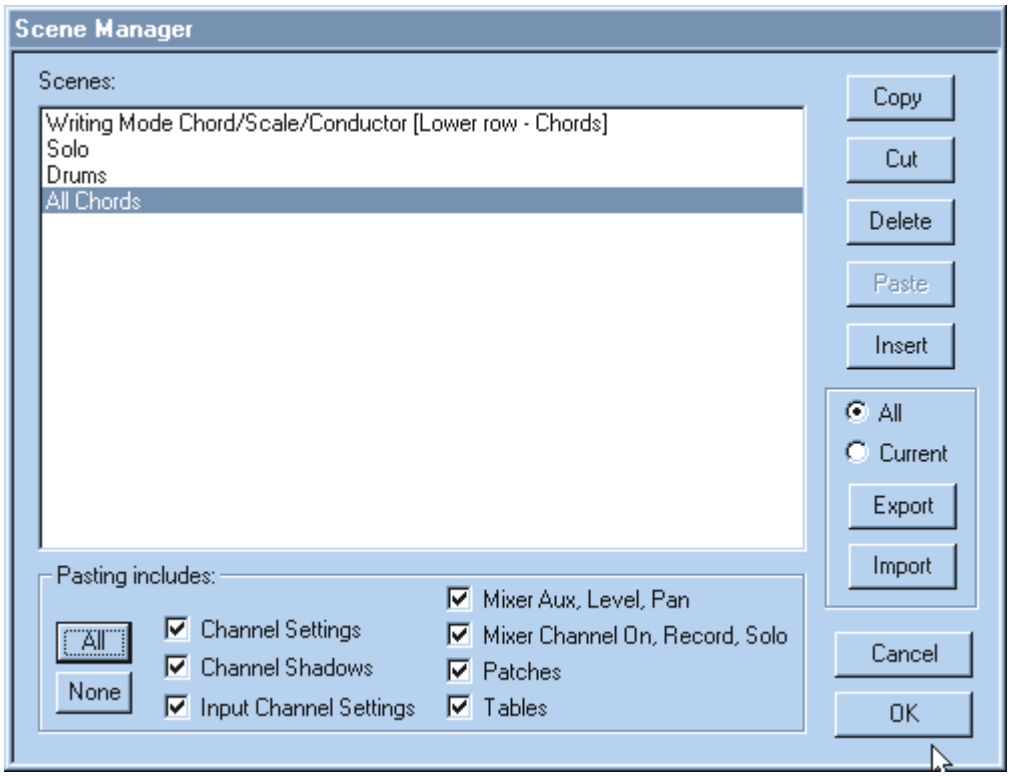
Click on the box next to the incoming channel number to scroll through the five processor types until the one you want is displayed. Choose **Chord** for playing chord notes and their scalar extensions, **Scale** for playing scales, and **Conductor** for selecting and changing tables in real-time. Thru is for non-translated channels, and **Melody** is an expert function to be covered in future documentation.

Melody Channel and **Channel 16 Selects** are in place for future applications, and should normally not be changed.

Warning: *Do not assign channel 1 to Conductor, as this can lead to a MIDI feedback loop and may freeze your system (Conductor Events output their "chord's notes" on channel 1 which should normally be assigned to Chord).*

Chapter 6: Menu Buttons

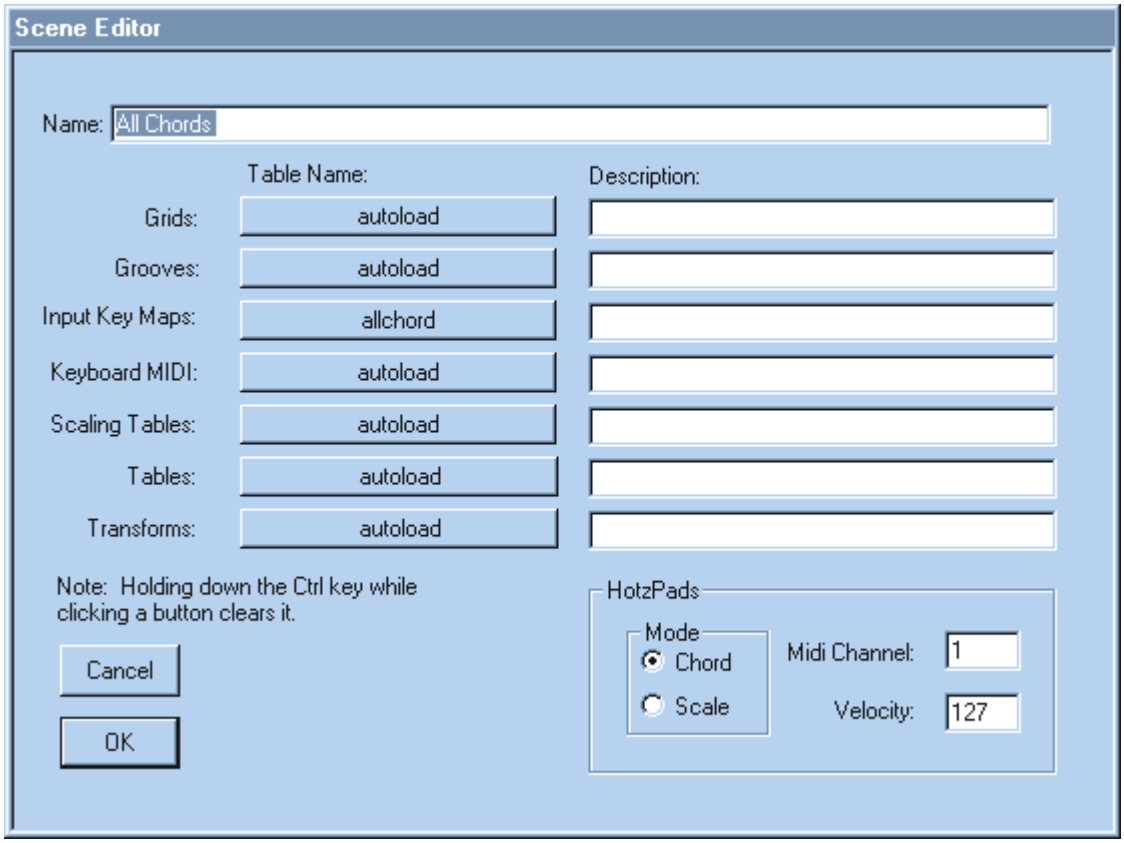
Scene Editor



To edit a **Scene** double-click on a **Scene name** or left-click and press OK.

Scenes may be cut, copied, and pasted using any or all of the data values whose options appear in the "**Pasting includes:"** area. All of these data values as well as **Track Mutes** are stored in each **Scene.** All of the Scenes currently listed in the Scene Manager are stored in the Song file.

Scenes may be imported and exported.



To select the specific Tables that you wish to use for the Scene you are editing, left-click on the desired Table Name.

An **Open** file selector will appear allowing you to choose a table.

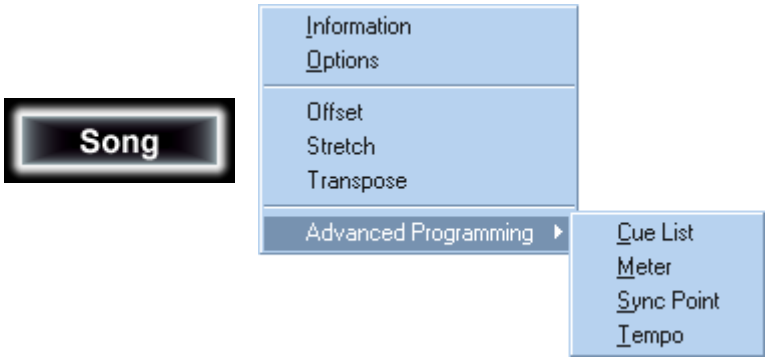
The **HotzPads** section allows you to set up options for **Virtual HotzPads** on Hotz Products that support this feature.

Chapter 6: Menu Buttons

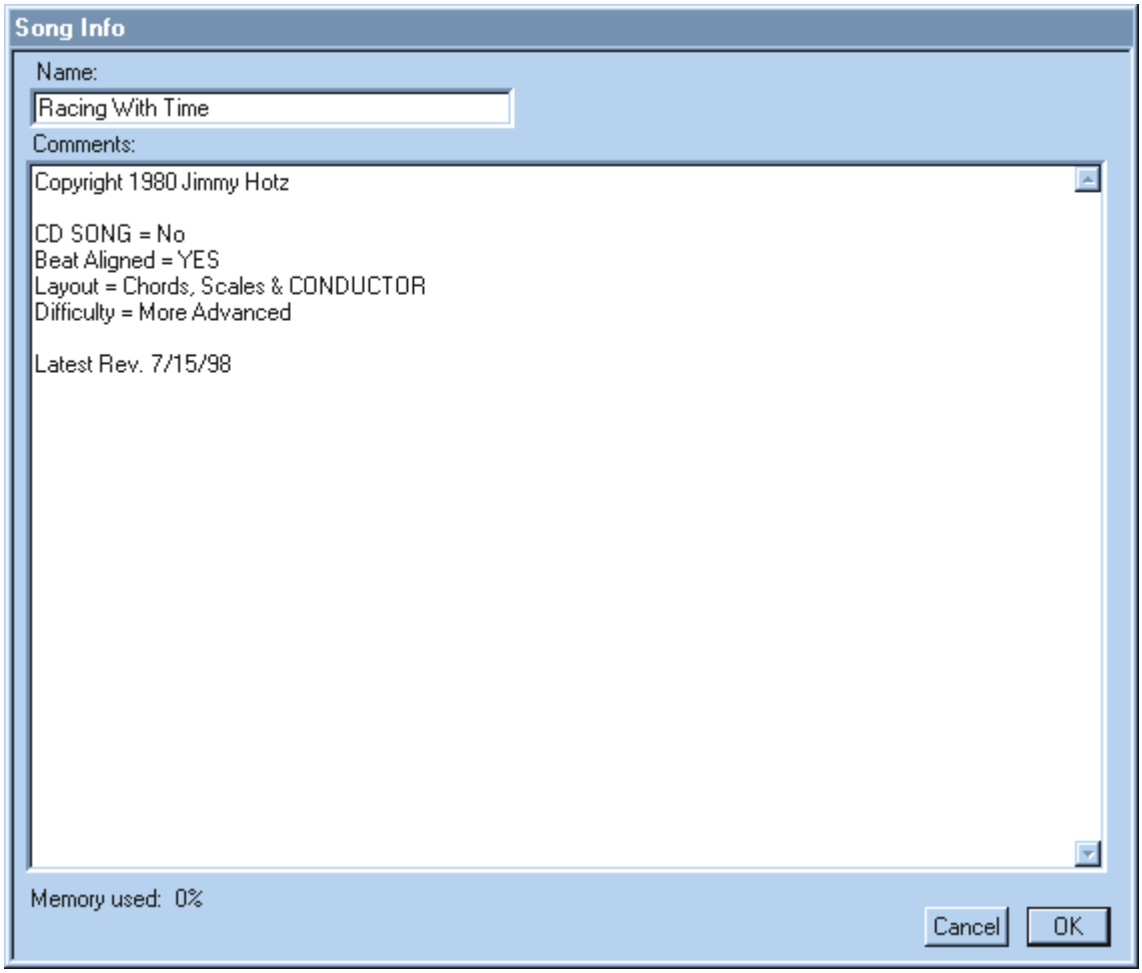
Song Menu Commands

The **Song Menu** provides access to a series of song-related controls such as **Tempo** and **Meter** tracks, synchronization settings, song offset in beats, and global duration values.

Song information includes all **Scene** data (**Mixer** settings, **Input Channel Settings**, and **Track Mutes**), **Transport settings,** and the **Conductor Cue List .**



Information



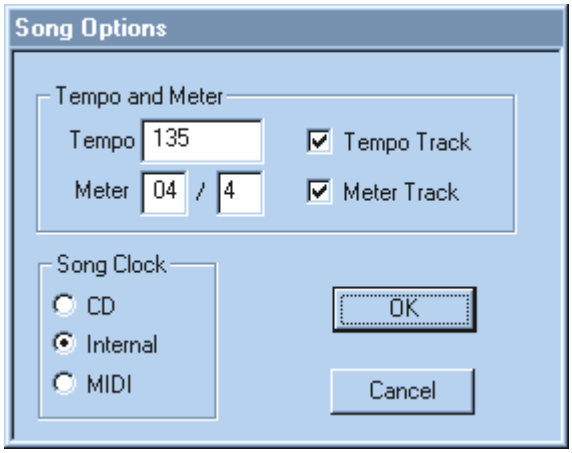
This area allows you to add comments to the song file.

Memory resources used by the song are shown in the lower left corner.

Chapter 6: Menu Buttons

Song Options

These Options are also available on the main screen's Transport/Location section.

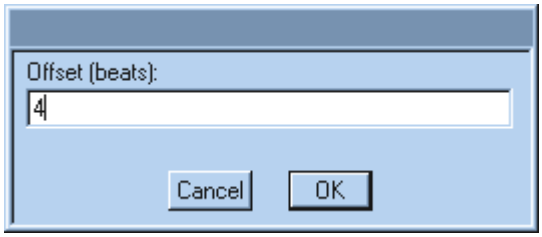


Tempo sets the speed of the song in "beats per minute". If Tempo Track is checked, its values will override the current Tempo value. For details on the Tempo Track see Tempo in the Song Menu.

Meter sets the values used in calculating the length of a measure. The left value is the number of beats in the measure while the right number is the type of note the beats refer to. A Meter setting of ¾ would mean 3 quarter notes per bar. If Meter Track is checked, its values will override the current Meter values. For details on the Meter Track see Meter in the Song Menu.

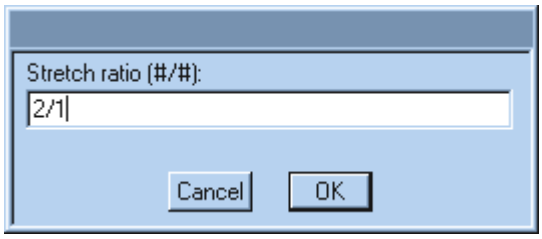
Song Clock selects the Clock/Sync source.

Offset (Song)



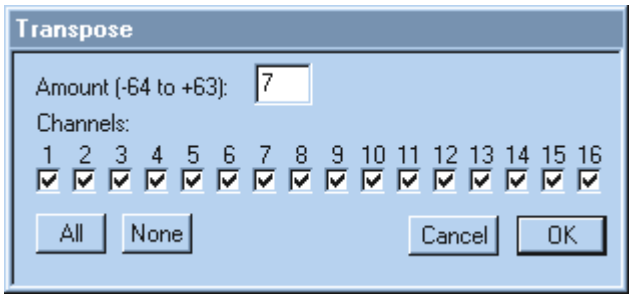
The **Offset Song** command is used to set the number of beats by which a song's tracks will be offset from the first bar, first beat. It can be used for a count-off by recording an empty track then offsetting the song before recording the musical tracks.

Stretch (Song)



Stretch Song can be used to proportionately move in time all of the events of a song. Any ratio with the higher number on the left will add to the length of the song while any ratio with the higher number on the right will reduce the length of the song. **Example:** A 2/1 ratio would double the length of the song, while a 1/2 ratio would half the length of the song.

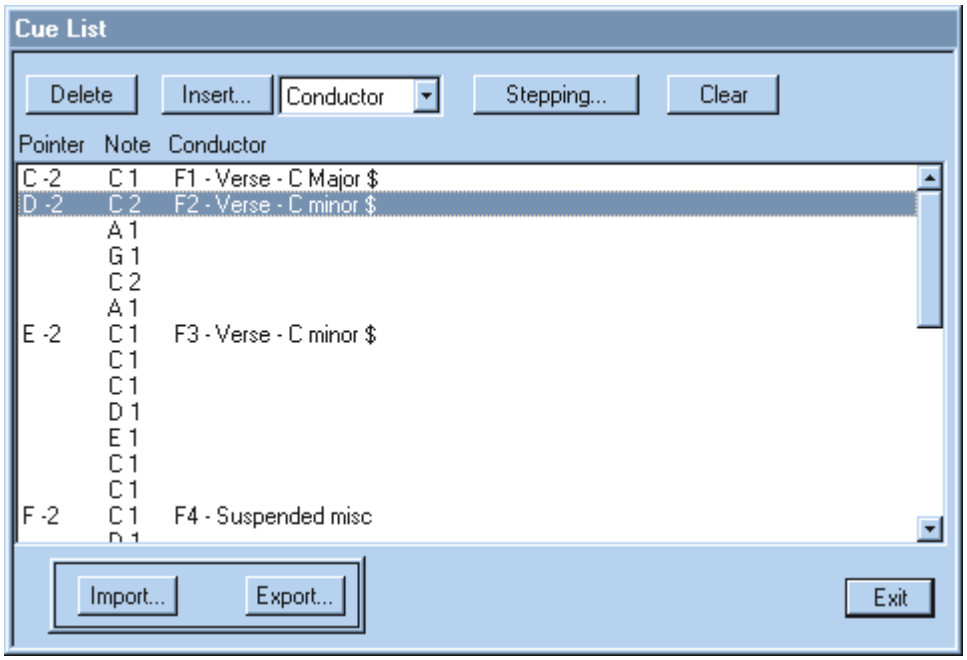
Transpose



Transpose is used if you want to transpose the whole song instead of particular tracks. A checkmark by a particular channel means any data on that channel will be transposed. Normally when transposing you should deselect (uncheck) any channels with drum parts.

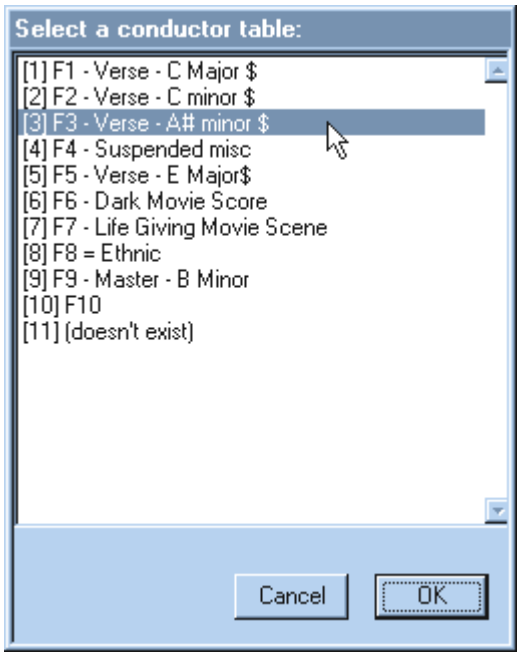
Chapter 6: Menu Buttons

Cue List



The **Cue List** window allows users to define the real-time and recording progressions of all Translation tables. Think of it as a master list of sequential table changes. It is organized into separate sections called **Conductor** tables. Conductor table **Cue Lists** can be created as a single list (or set) for a whole song or created separately for Intro, Verse, Chorus, etc. The Stepping function is used for stepping through the Cue List entries in a sequential manner.

The best way to understand a Conductor Table is to grasp the meaning of two important terms, **Pointer** and **Conductor Note**. Both terms refer to MIDI notes, but with different functions. **Pointer notes are used to select different Conductor Tables.** Assume a Cue List similar to the one shown above. Notice that Conductor Table F2 Verse C minor has (to its left) the Pointer D-2. If you were to play a D-2 note on your MIDI keyboard (on channels 4 or 16), this Conductor Table would become the current Conductor (the F2 key on the computer keyboard would produce the same result because it sends a D-2 note on channel 16, which is set to Conductor processing). If you play an E-2 note (F3 key on computer Keyboard), the F3 Chorus D Major Conductor Table will become the current Conductor (since its Pointer is E-2). The reason Pointers should be of very low values is that this area of the MIDI spectrum is hardly ever used to play sounds.

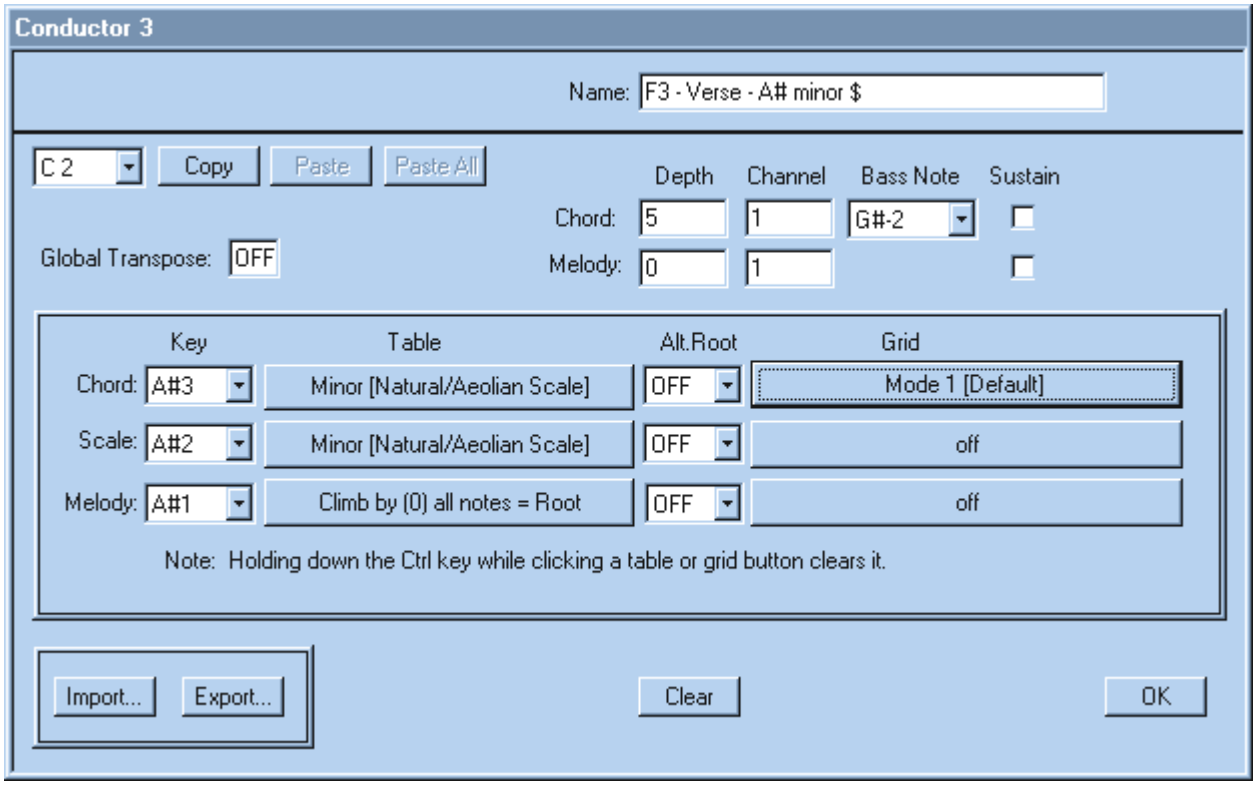


Click at the desired location in the Conductor column to insert a Conductor Table. The **Conductor Table List** will open, and you can choose from existing tables or click on ***doesn't exist*** to create a new one.

The Conductor Table is the main organizational building block for sequencing Table changes. It is analyzed in detail below.

Chapter 6: Menu Buttons

Conductor Note Assignments



Double-click on a **Conductor Table name** in the **Cue List** to open that particular **Conductor Table**. It will look like the figure above. Notice that C2 is the Conductor Note in the upper left corner. Whenever a C2 note is received by the Translator on a channel assigned to *Conductor* processing, the various parameters and Tables on this window will become active in the Translation engine. Think of this as the C2 snapshot of values for this Conductor Table. A Conductor Table can have as many snapshots as there are MIDI notes, but a good standard is to start assigning white notes from C1 on up as Conductor Notes. It is also a good idea to have several different Conductor tables set up for different parts of a song. This keeps the whole process of coding and real-time improvisation as simple as possible (keeping track of twenty changes is much simpler if they are logically organized into separate sections).

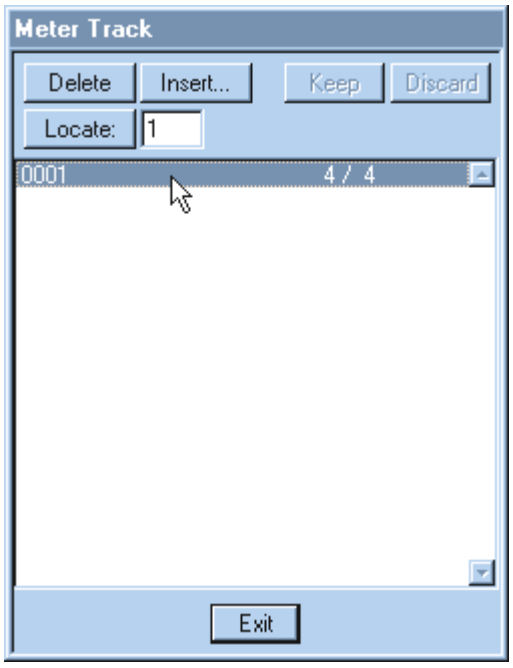
Copy is used to copy all values of the current snapshot. **Paste** will paste the copied snapshot values into a newly selected Conductor Note snapshot. **Paste All** pastes the copied snapshot values into all 128 Conductor Note snapshots of the Conductor table (it is used primarily for initializing a Conductor table).

The **Global Transpose** value is used to transpose all active table values by the same number of semitones. The Chord and Melody **Depth**, **Channel**, and **Sustain** control the note output values when the snapshot becomes active (this feature is useful if you want to hear a reference chord when new tables become active, important when evaluating the timing accuracy of recorded Hotz code changes). A Depth of 5 is a good standard value (this means the chord will be played with 5 chord notes and a bass note). The **Key**, **Table**, **Alternate Root**, and **Grid** values can be set separately for the three different structure table processors. **Import** and **Export** are used to load and save individual Conductor Tables. **Clear** is used to clear all Conductor Table snapshots (values).

Chapter 6: Menu Buttons

Meter (Track)

The **Meter** track is used to define Meter changes for a musical sequence (MIDI or CD).



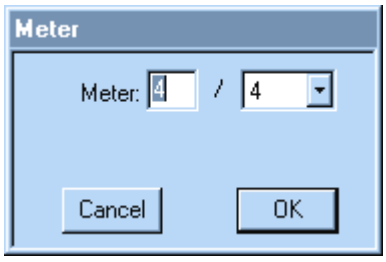
Locate is used to jump to a specific point. Enter the (Bar, Beat and Tick) values and then left-click on Locate to go there.

Delete is used to delete a selected **Meter** event. **Keep** adds the changes made during the current editing session to the track data. **Discard** deletes all changes made during the current editing session.

Use **Insert** to add a copy of the selected **Meter** event.

Editing a Meter Event

To **Edit** an event, double-click on it.

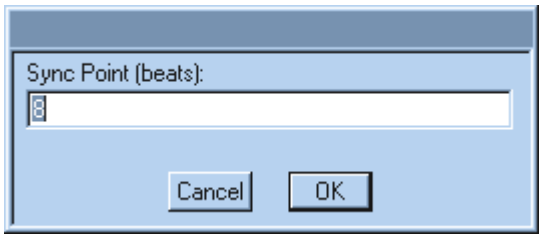


Set the desired **Meter** values for the current Bar/Beat/Tick Location.

Chapter 6: Menu Buttons

Sync Point

The **Sync Point** determines at which place in a song CD audio playback starts. It is ignored in Internal and MIDI clock modes.



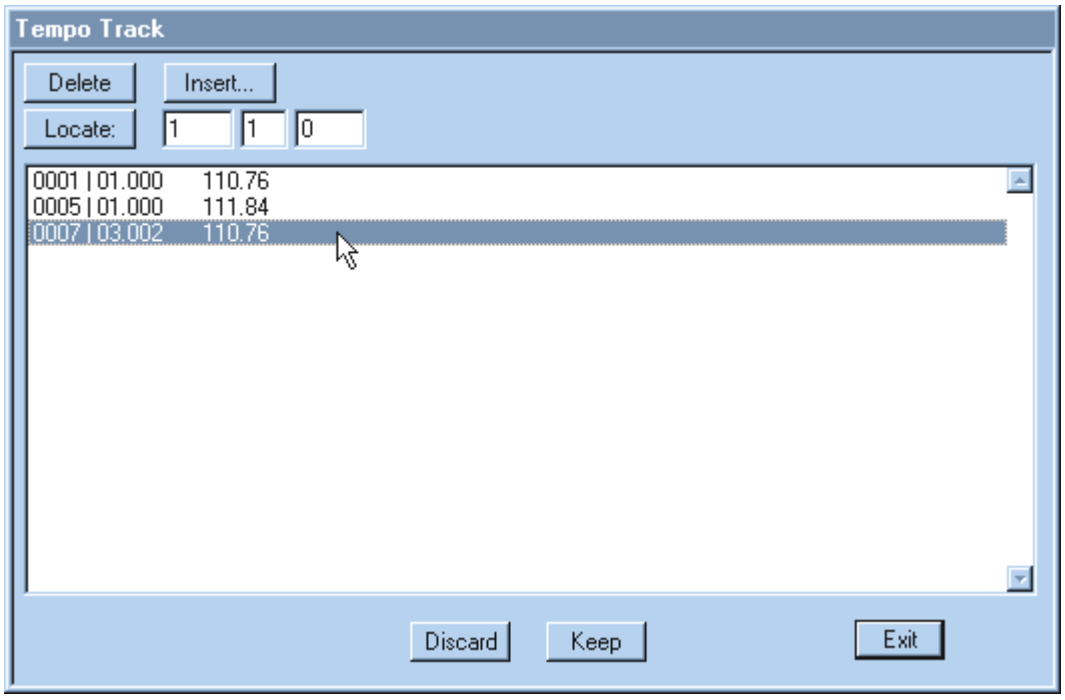
The **Sync Point** and the **Offset Song** function combine to give the authoring user the ability to have an effective countoff, and even have that countoff be in a different time signature than the first bar.

Let's say you have an existing song that is in 4/4 overall but the first bar on the CD only has 1 beat. Set up a meter table where bar 1 is at 1/4 and bar 2 is at 4/4, and then do a Beat Align (Beat Align will know that the first bar only has one beat). Once that is completed, determine how long you want the count–off to be. Let's say it's 7 beats, to have one full bar, plus three more rests before the pickup beat. Use Offset Song to offset all tracks by +7 beats. Then edit the Meter Table to change the initial meter to 7/4, and insert the 1/4 meter change at bar 2. Finally set the Sync Point to the start of bar 2.

Now when you play in CD Audio Sync, the sequencer will play from the same point where the start of the song was before, but when you play in Internal Sync you have 7 extra beats. Be aware that anything in those beats before the Sync Point won't get played (although it will get chased) if you play in CD Audio Sync, so you should be careful to move any early MIDI events to at or past the Sync Point if you want them to play with the CD.

Chapter 6: Menu Buttons

Tempo (Track) [see also Beat Align/Track Menu]



The Tempo Track provides an advanced way to sync up accurately to musical passages that vary in tempo during the performance. It is used in two different ways: to create a tempo variation track for a Hotz-coded MIDI sequence or to generate (in conjunction with the Beat Align [Track Menu] command) a customized tempo map for a CD + Hotz coded song. Song Tempo values should be set using this command, not from the Transport window (which is designed to display Tempo values and can be used to turn the Tempo track on [X] or off). The Tempo track must contain at least two entries before it can be turned on. A green “TT” indicates the Tempo track is active.

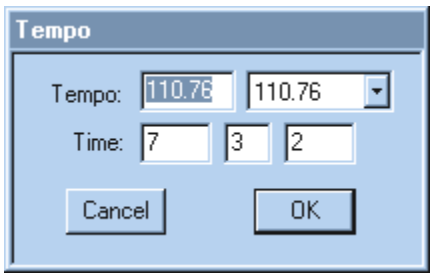
Locate is used to jump to a specific point. Enter the Bar, Beat, and Tick values and then left-click on Locate to go there.

Delete is used to delete a selected Tempo event. **Keep** adds the changes made during the current editing session to the track data. **Discard** deletes all changes made during the current editing session. Also see the **Beat Align** section for more information on creating sophisticated Tempo maps.

Use **Insert** to add a copy of the selected Tempo event.

Editing a Tempo Event

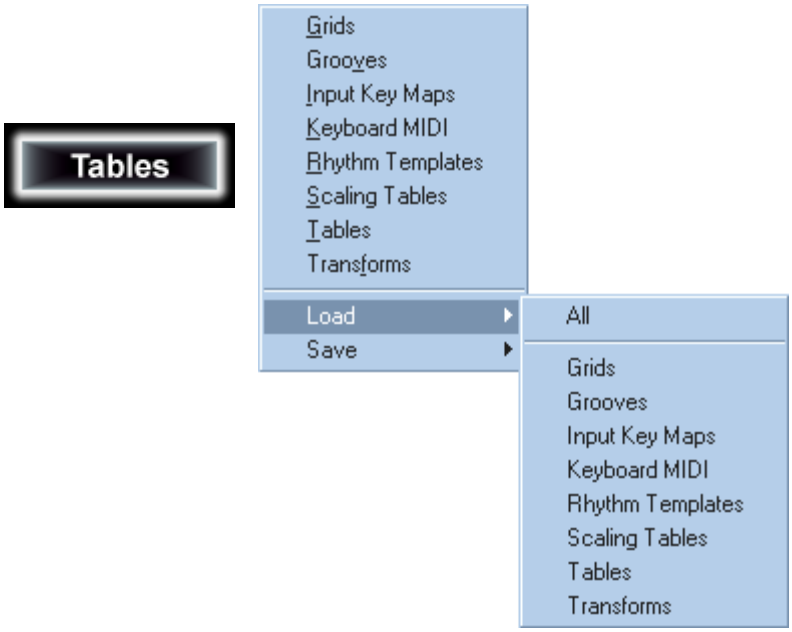
To **Edit** an event double-click on it.



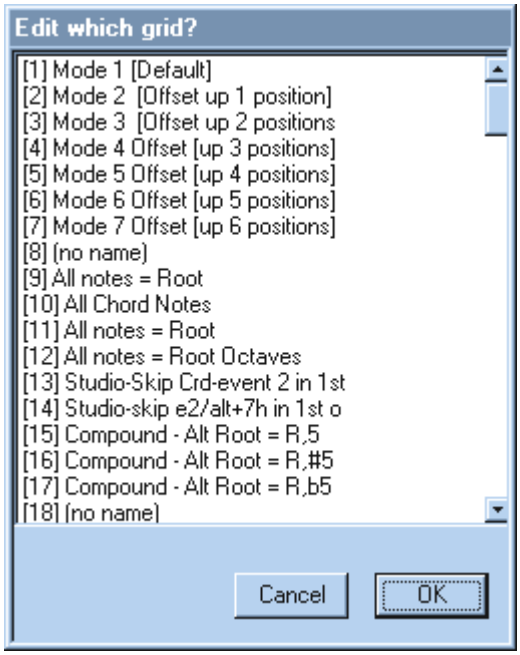
Set the Bar/Beat/Click Location and the desired Tempo value for the event (*notice that Tempo values are shown exactly to 2 decimal places, not as approximations*).

Chapter 6: Menu Buttons

Tables Menu Commands



Grids



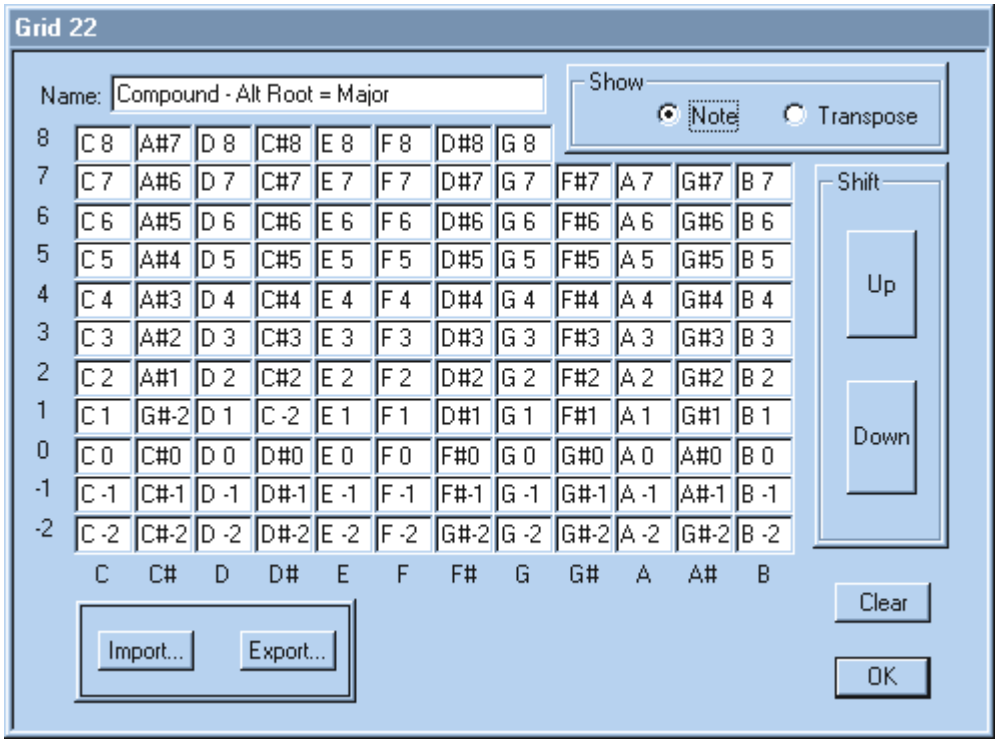
Grids are closely associated with **Chord/Scale Tables** because they determine how the data in the **Chord/Scale Tables** can be used. Acting as an additional manipulator in the translation process, they allow super indexing into the **Chord/Scale Tables**.

Grids allow data that is stored in specific locations within the table, which are normally used by a specific input, to be used by any input event that has access to the data tables. This greatly amplifies the already formidable power of the **Chord/Scale Library** by allowing endless rearrangement, mode shifting, and hyper-transposition of all existing tables in a consistent manner. Grids can be dynamically selected from a list of 128 preprogrammed and user-configurable grids. Additional banks may be created if needed. Grids are selected/triggered during music sequences, or in live performance, by HotzCode conductor events the same way that chord, scale, and melody tables are selected.

Grids can also be used to change the mode of a Scale (i.e. G, A, B, C# ... would become A, B, C#, D... when using the *Mode 2 Offset 1 position* grid). Since grids can be changed in real-time, they offer endless possibilities for extending structure table output and functionality.

Chapter 6: Menu Buttons

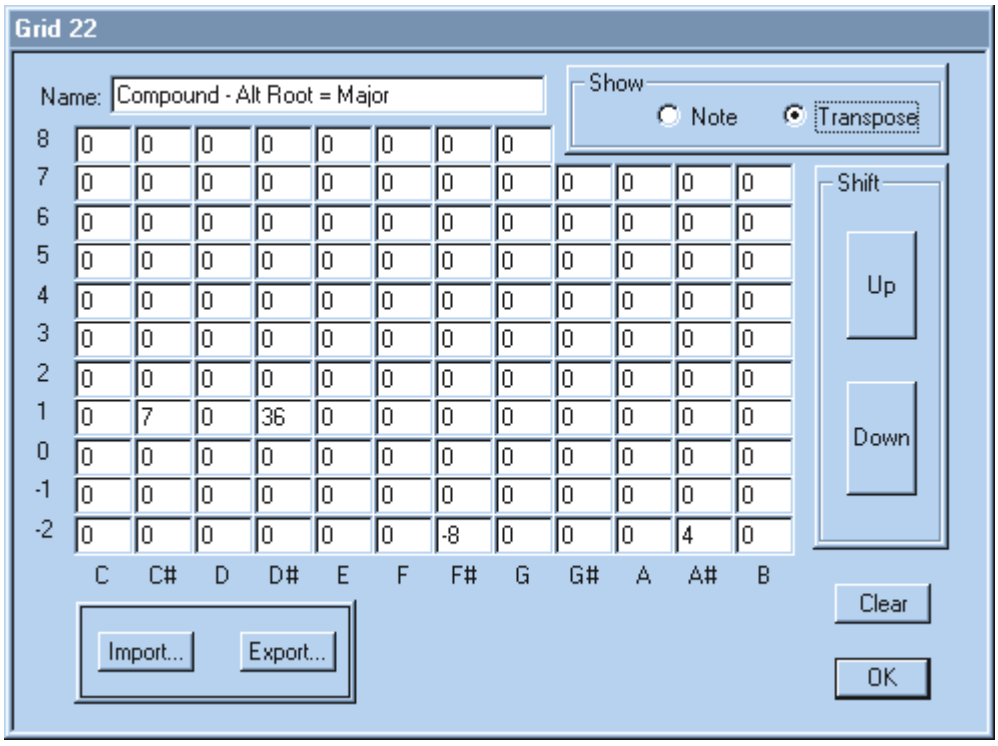
Grid Editor



Show (**Note)** selects the **incoming note** to **outgoing note** grid for display and editing.

Name is used to display and edit the name of the **Grid.**

Up/Down are used to move all displayed values up or down a semitone. Individual positions can be changed by click-dragging in them or typing of new values in the desired location.



Show (**Transpose)** selects the **outgoing notes'** transpose value for display and editing.

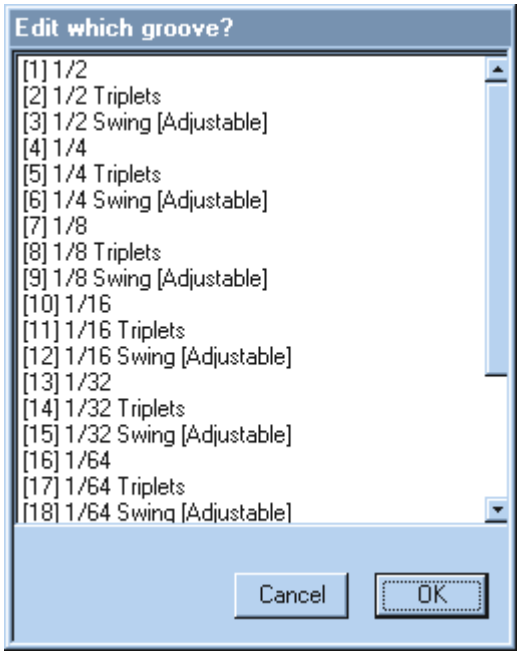
Up/Down are used to move all displayed values up or down a semitone. Individual Transpose may be changed by click-dragging.

Clear sets all Grid values to a default setting that essentially has no index manipulation or additional transposition.

Individual **Grids** may be imported and exported.

Chapter 6: Menu Buttons

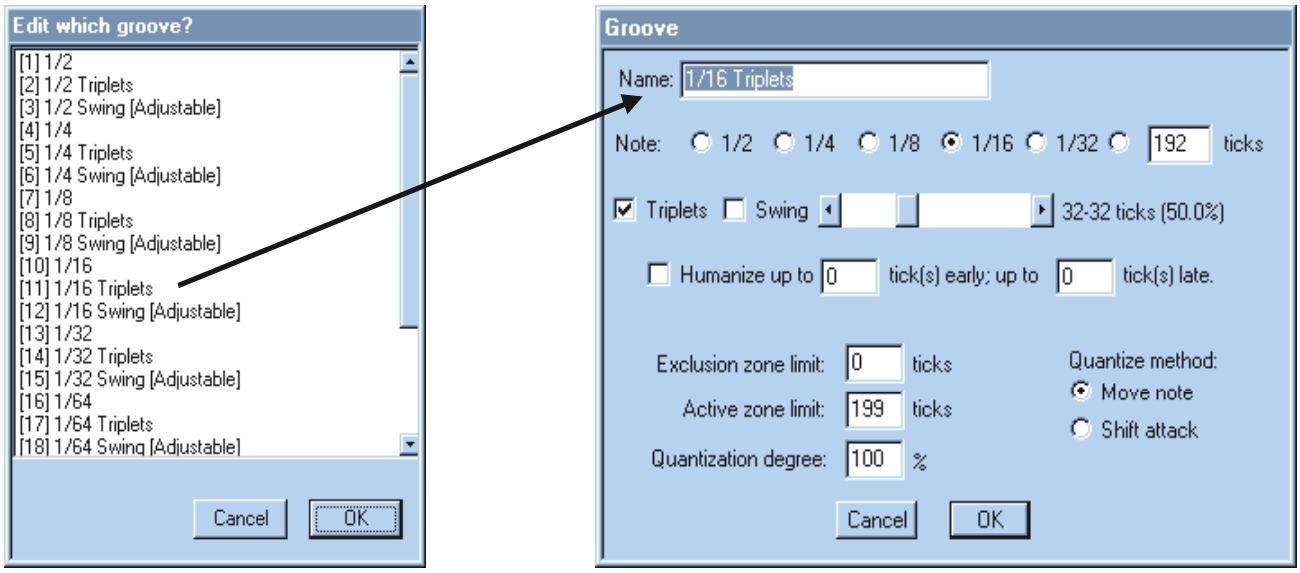
Grooves



Presents a list of **Groove/Quantize** tables. Double-click on the groove name or left-click and select OK to open the Groove editor. New **Groove/Quantize** tables can be added by scrolling to the bottom of the list and selecting **"Doesn't Exist".** Added tables must be saved with the **Save Grooves** command.

Groove Editor

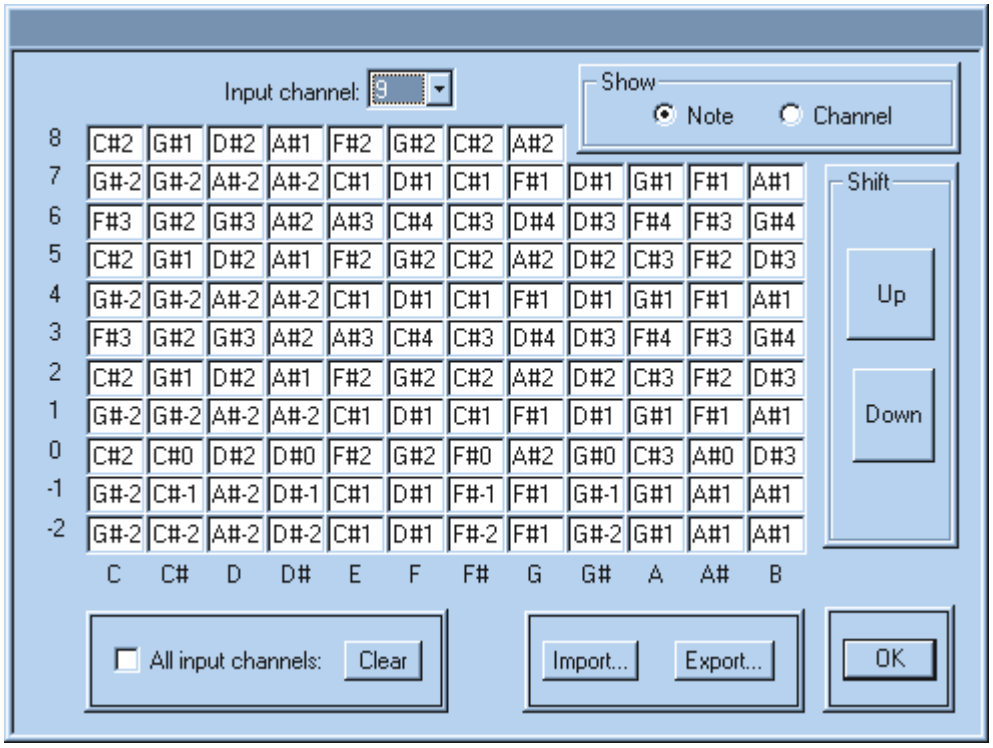
There are two types of quantization provided: **Move Note** moves the note itself. **Shift Attack** moves only the attack point. **Note** sets the basic value of Quantize operation. Check **Triplets** if you want to quantize to triplets, or **Swing** for a swing feel (set swing percentage with the scroll bar). **Humanize** values allow for random variations within the chosen tick limits. **Exclusion** zone limit leaves notes within selected tick values unquantized (i.e. if set to 6, then any note less than six ticks away from the target location will not be quantized). The **Active Zone** limit represents the zone over which quantization will be applied (199 represents "no limit"). **Quantization degree** is the percentage of correction to be applied (50% means move only half-way to target value).



Chapter 6: Menu Buttons

Input Key Maps

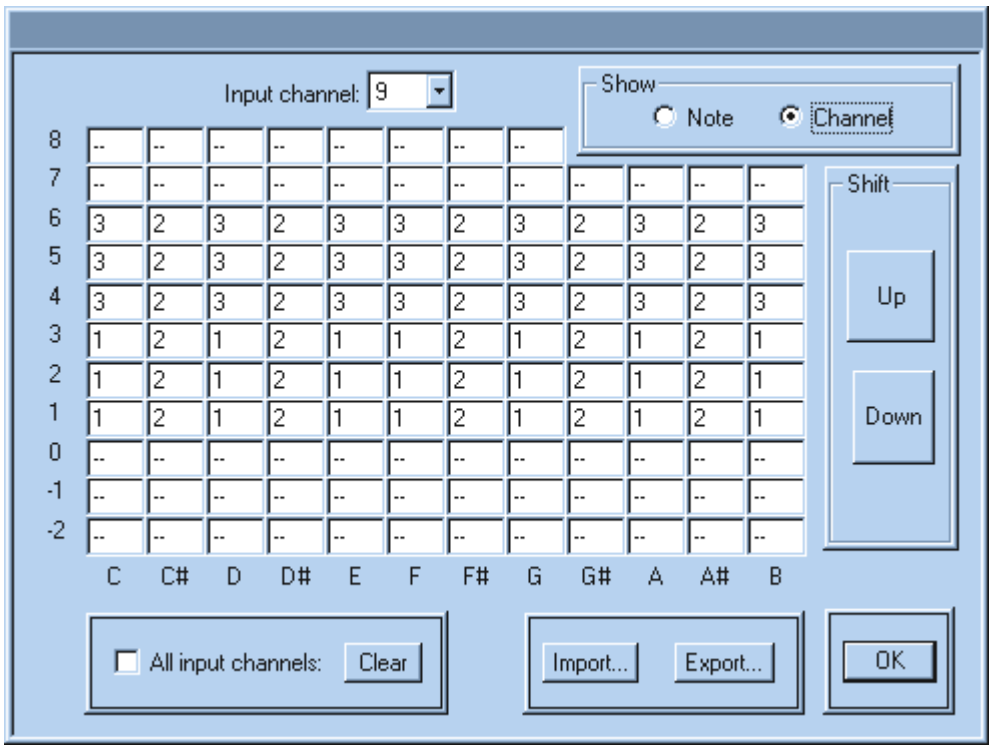
Input Key Maps are used to redefine global MIDI input NOTE and CHANNEL values before they are processed by the structure Tables. There are separate mapping tables for each channel, or all channels can be mapped by one table.



Show (**Note)** selects the **incoming note** to **outgoing note** grid for display and editing.

Input Channel (at top) selects the input channel map to be edited.

Up/Down are used to move all displayed values up or down a semitone. Individual positions can be changed by click-dragging in them or typing of new values in the desired location.



Show (**Channel)** selects the **incoming note** to **outgoing note channel** grid for display and editing.

Up/Down are used to move all displayed values up or down. Individual values may be changed by click-dragging.

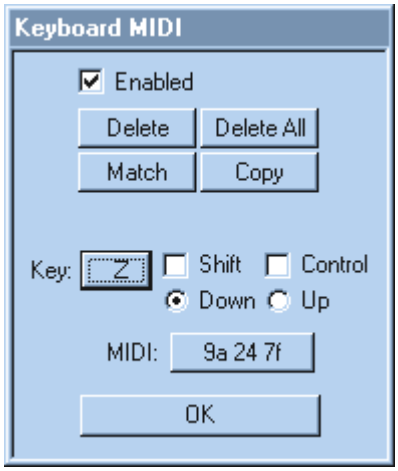
Clear sets all **Input Key Maps** values to a default setting that essentially has no index or channel manipulation.

Individual **Input Key Maps** may be imported and exported.

Chapter 6: Menu Buttons

Keyboard (Computer) MIDI

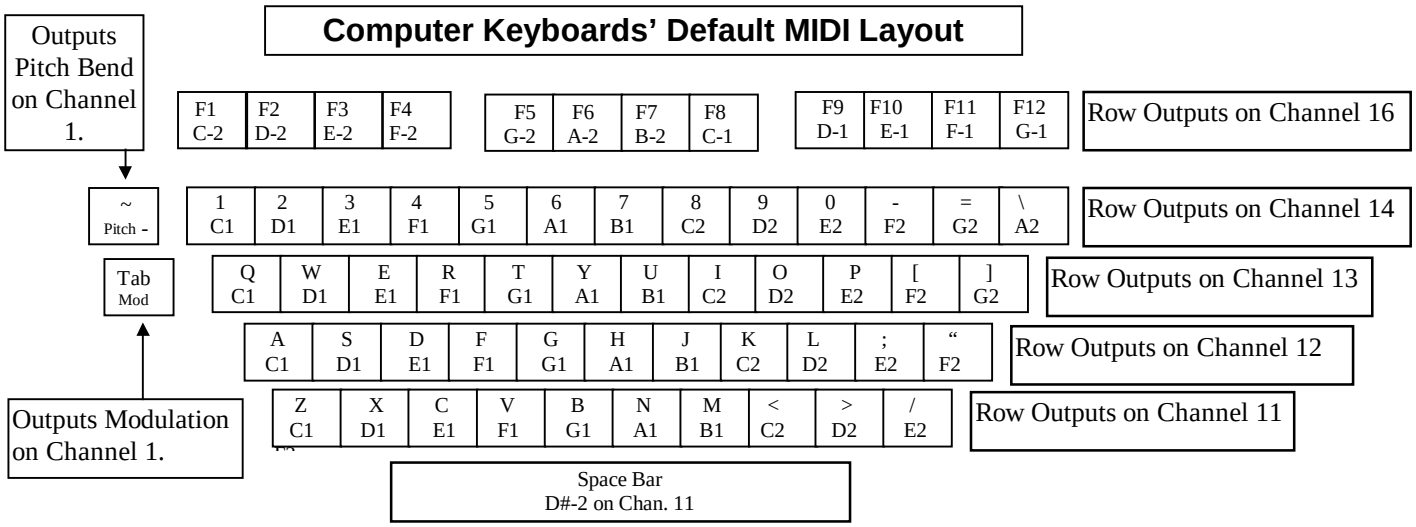
Make sure the **Keyboard MIDI** option **Enabled** is checked (Tables Menu).



The Keyboard MIDI option can be used to produce incoming MIDI data. In the current software version, different keyboard rows output on channels 11-16. However, Input Key Maps are used to re-route channels 11-14 to channels 1-4. Keep this in mind if you are sending on channels 11-14 from your MIDI keyboard. The channel data will be re-routed by the default Input Key Maps).

Default Computer Keyboard MIDI Layout

The following graphic illustrates the MIDI output values of the keyboard layout.

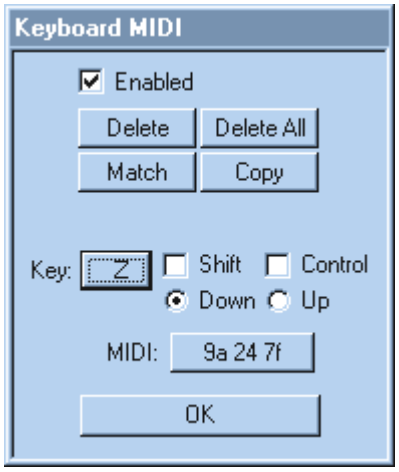


Transpose buttons: **↑= +12 ; ↓= -12 ←= -1 ; →= +1**

The computer keyboard default layout can be useful for Hotz coding and auditioning current Tables. For Hotz Coding, make sure the Channel Assignments for Translation Tables are set to: **Conductor** on Channels 16 and 4. Remember that selecting Solo record on any Mixer channel will direct all input to that channel, so this is a good way to audition scales (key rows A,Q,1) and chords (key row Z).

Chapter 6: Menu Buttons

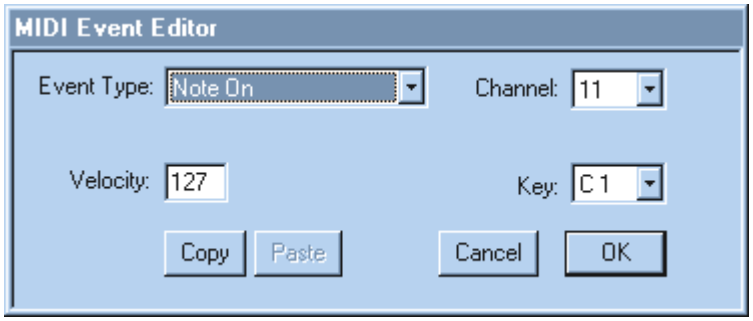
Editing Keyboard MIDI



The **Keyboard MIDI** window can be used to redefine the MIDI functions of the computer keyboard keys. Each Key Down and Key Up has to be assigned separately. Click on Key A button to edit any key. The key editor will become active, and the next key touched can be edited.

Delete clears the current key values, and **Delete All** clears all key assignments. If you edit the default assignments, be sure to save your new work (Save Keyboard MIDI or Save All...in the *Advanced Menu*). The **Match** button automatically places a note off event for the up part of all keys assigned note on events. **Copy** is used to copy a whole bank of key values (shift, control, etc.) to another bank.

Click on the button just to the right of **MIDI:** to open the MIDI Event Editor.



Here you can set the desired **Event Type**, **Channel**, and values for the **Data Bytes** (**Key** and **Velocity** in this example) that correspond to the **Event Type**.

The tags and display of the **Data Bytes** will depend on the **Event Type** chosen.

Example above:

Event Type = Note On

Channel = 11

Data Bytes

Key(Note) = C1

Velocity = 127

Chapter 6: Menu Buttons

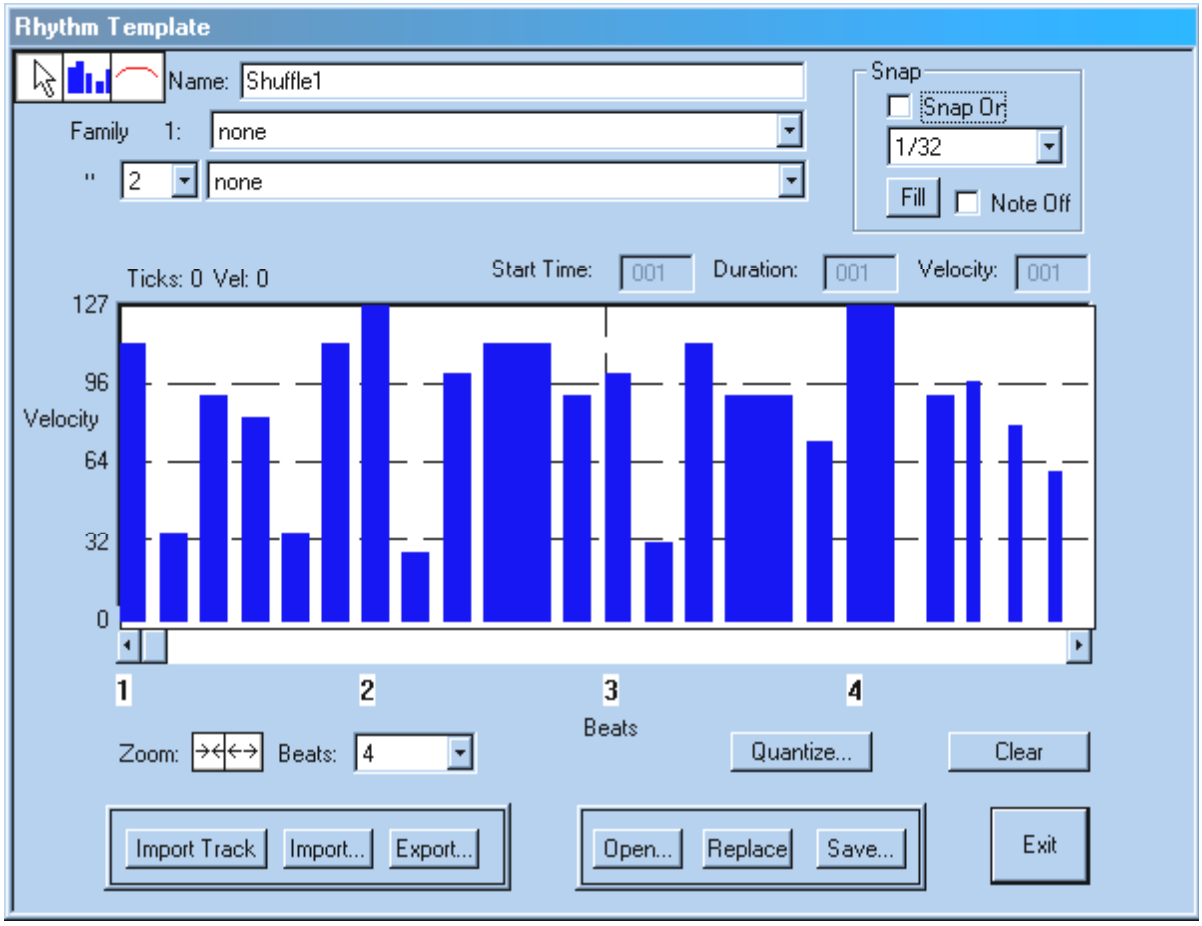
Rhythm Templates

The Hotz Rhythm Engine

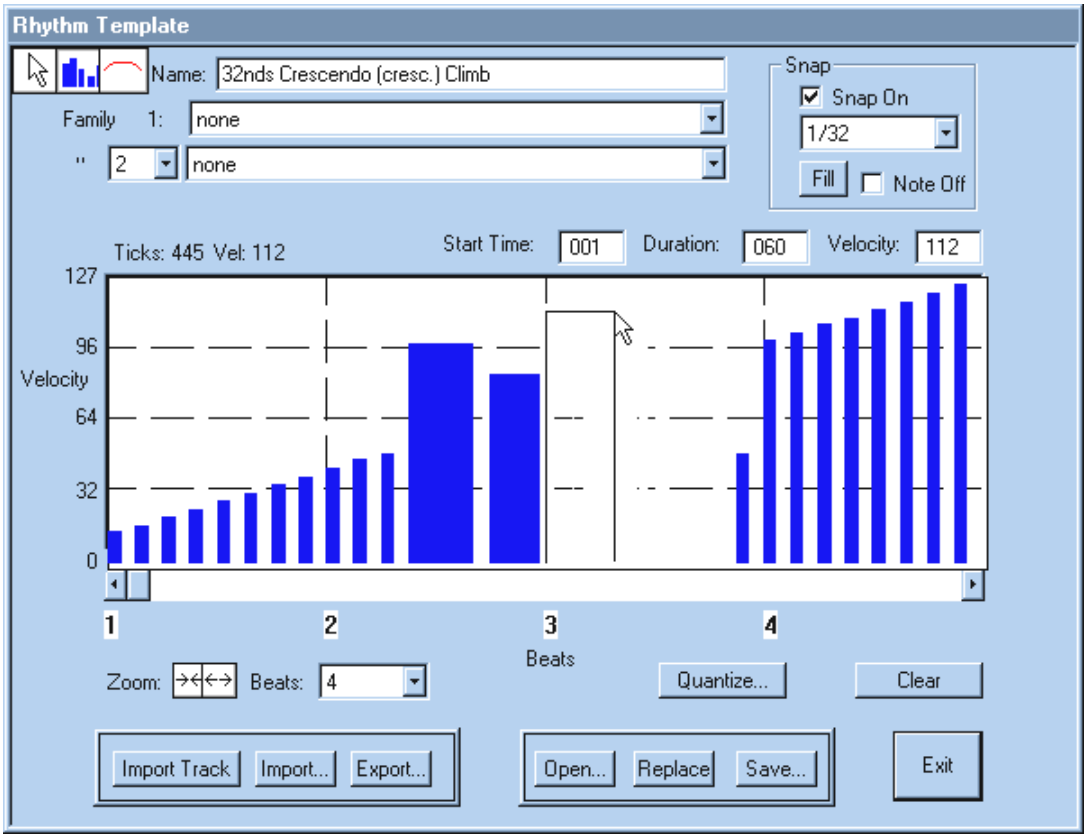
The **Hotz Rhythm Engine** does for rhythm (timing manipulation) what the Hotz Translation Engine does for Chord and Scale manipulation in real-time. Computer assisted manipulation of the rhythmic elements of music in real-time. This enables on the fly changeable timing correction while the song progresses and provides a way to perform difficult rhythmic feats without error.

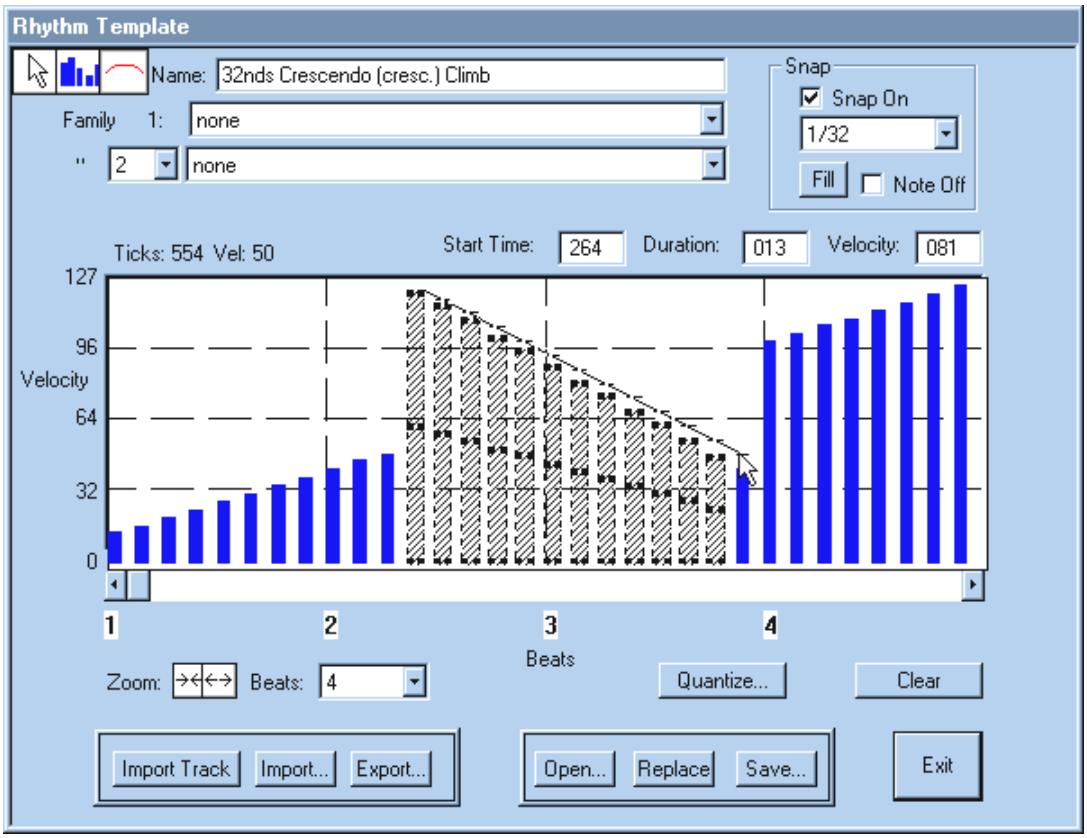
Hotz is currently developing a **Rhythm Library** for the **Rhythm Engine.** The template below is used to create that Library.

While it is presently beyond the scope of this manual to cover the details of the **Rhythm Template** editor and exactly how these tables are used, we have included a few screen dumps for your visual reference. Some of its functionality will become obvious when experimenting with it.



Chapter 6: Menu Buttons



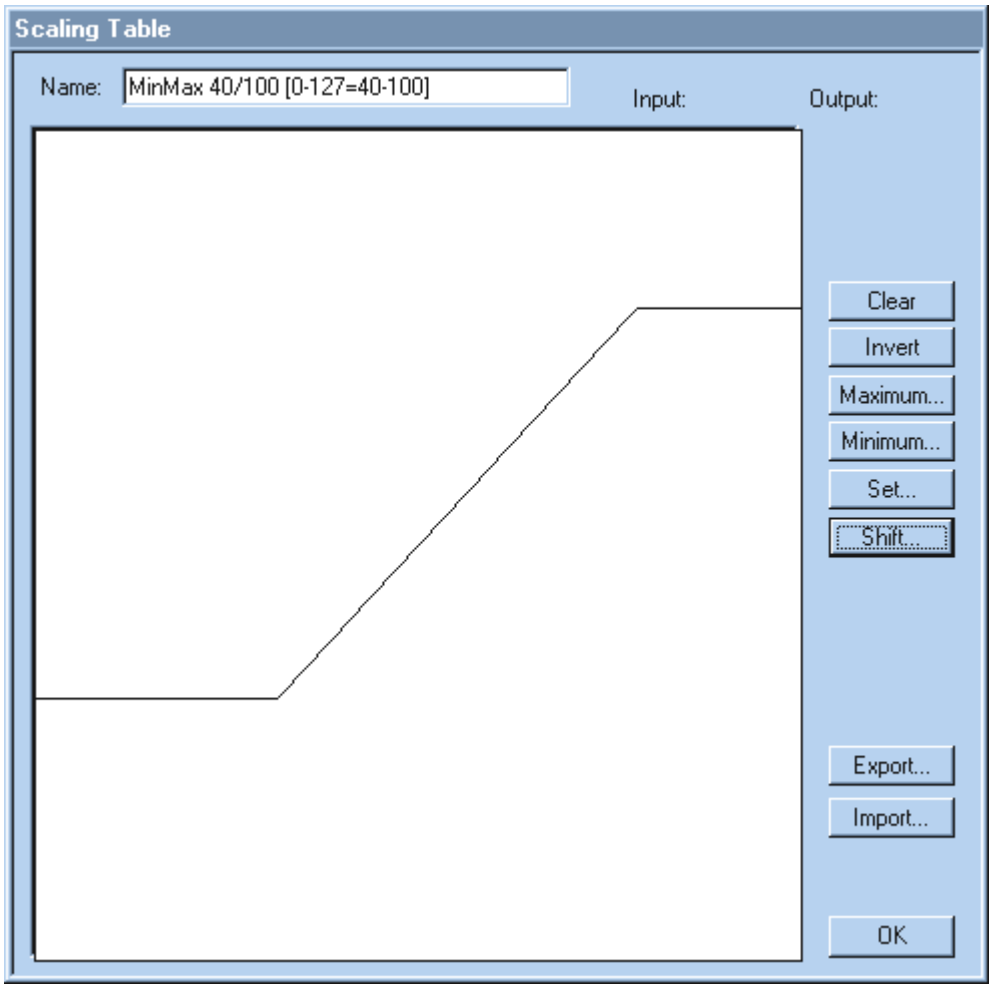


Chapter 6: Menu Buttons

Scaling Tables

The **Scaling Table Editor** is a graphical editor that allows mouse draw editing of the Scaling Tables. During mouse draw, the "**Input:**" shows the value for the incoming data byte while the "**Output:**" shows the value the incoming data byte will be transformed to. The **Name** is an editable text field.

Additional commands are as follows:



Clear

Clear resets the table to input = output.

Invert

Invert takes the current Velocity curve and reverses all its values (i.e. low input = low output becomes low input = high output).

Maximum, Minimum

Maximum, Minimum are used to easily define a ceiling and a minimum value for the Velocity curve.

Set

Set fixes the Velocity curve to a constant value (all input = 64 output, for example).

Shift

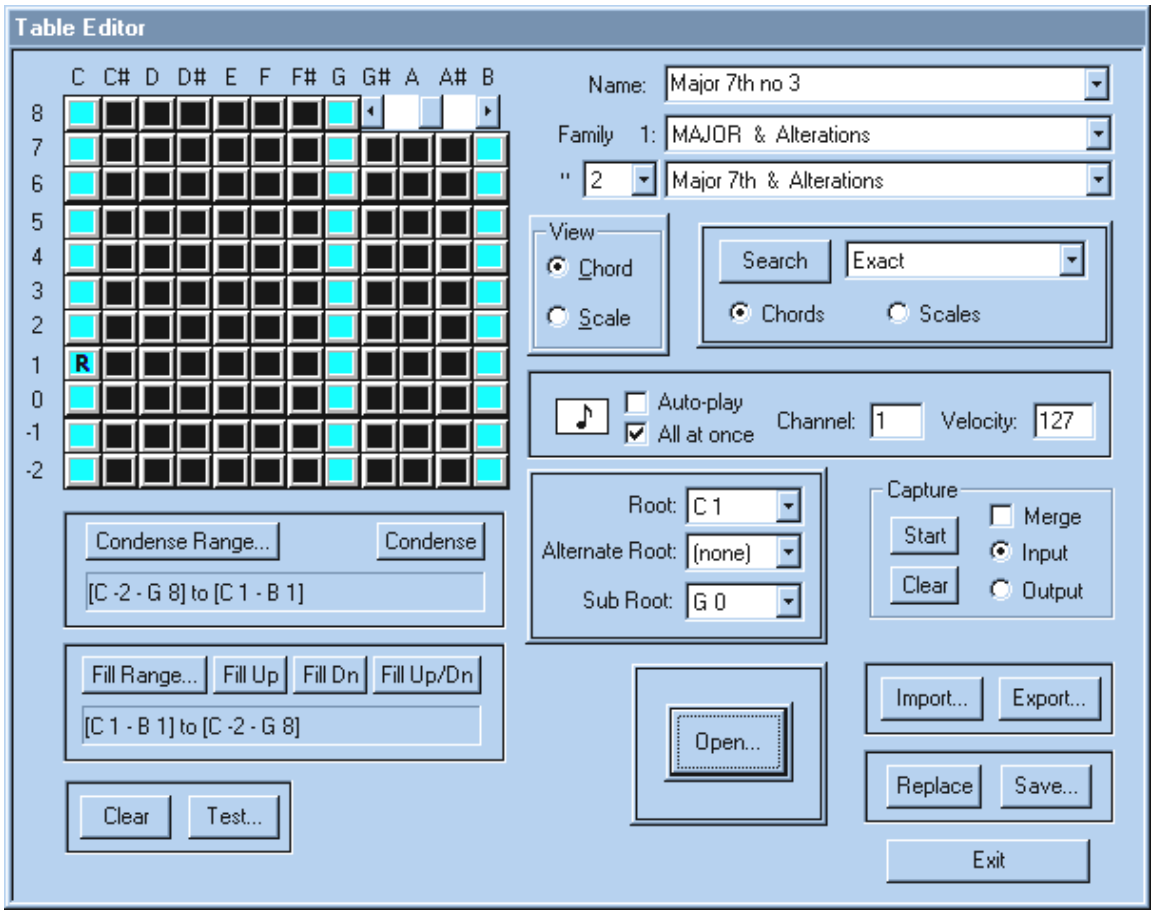
Shift uses an entered value (plus or minus 0-127) to shift the current Velocity curve up or down by a set amount.

Export, Import

Export, Import are used to save and load Scaling Tables as single files (extension .SC1). Scaling Tables are also saved in banks.

Chapter 6: Menu Buttons

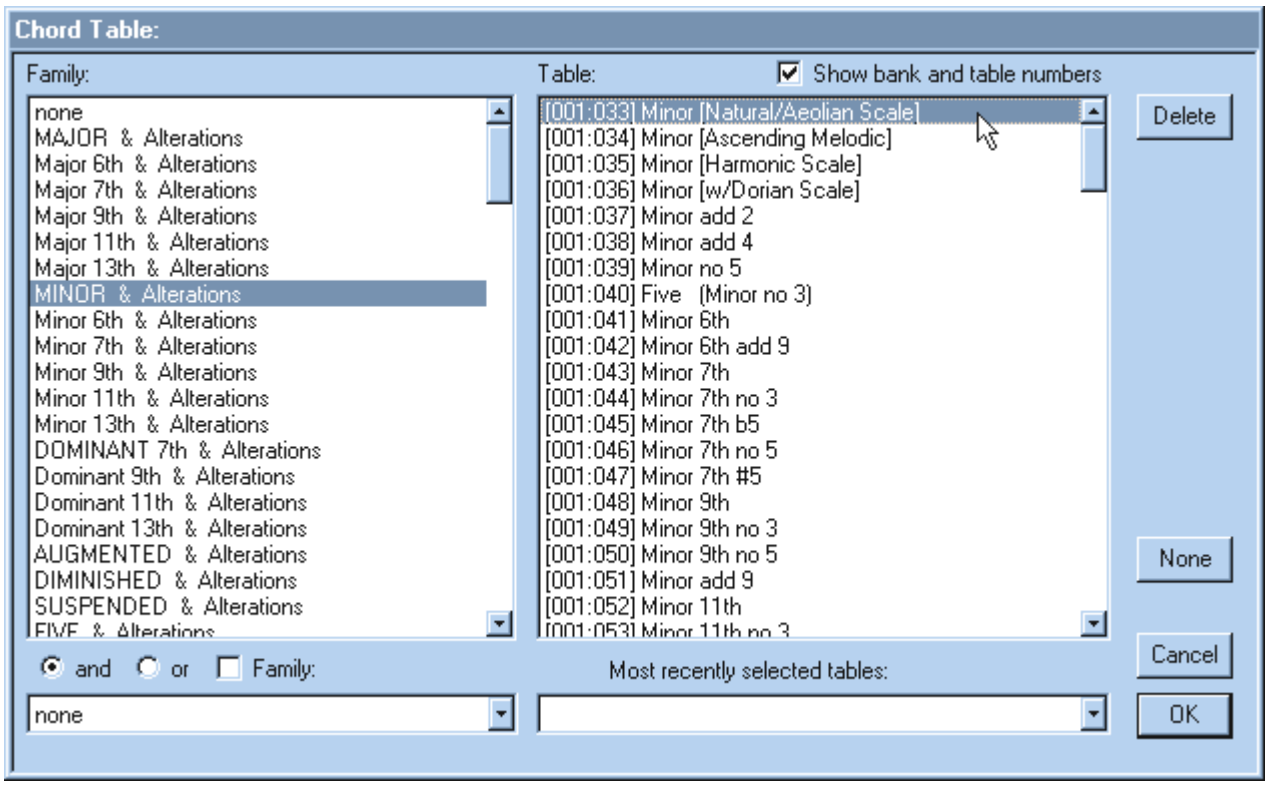
Tables (Chord and Scale Structure Tables)



The **Table Editor** can be used to search the database for specific structures as well as add new structures to the database.

To view a **Table,** click on **Open** and choose the desired structure from the list window that pops up.

Tables list window. Click on a family name to see only the structures belonging to that family.

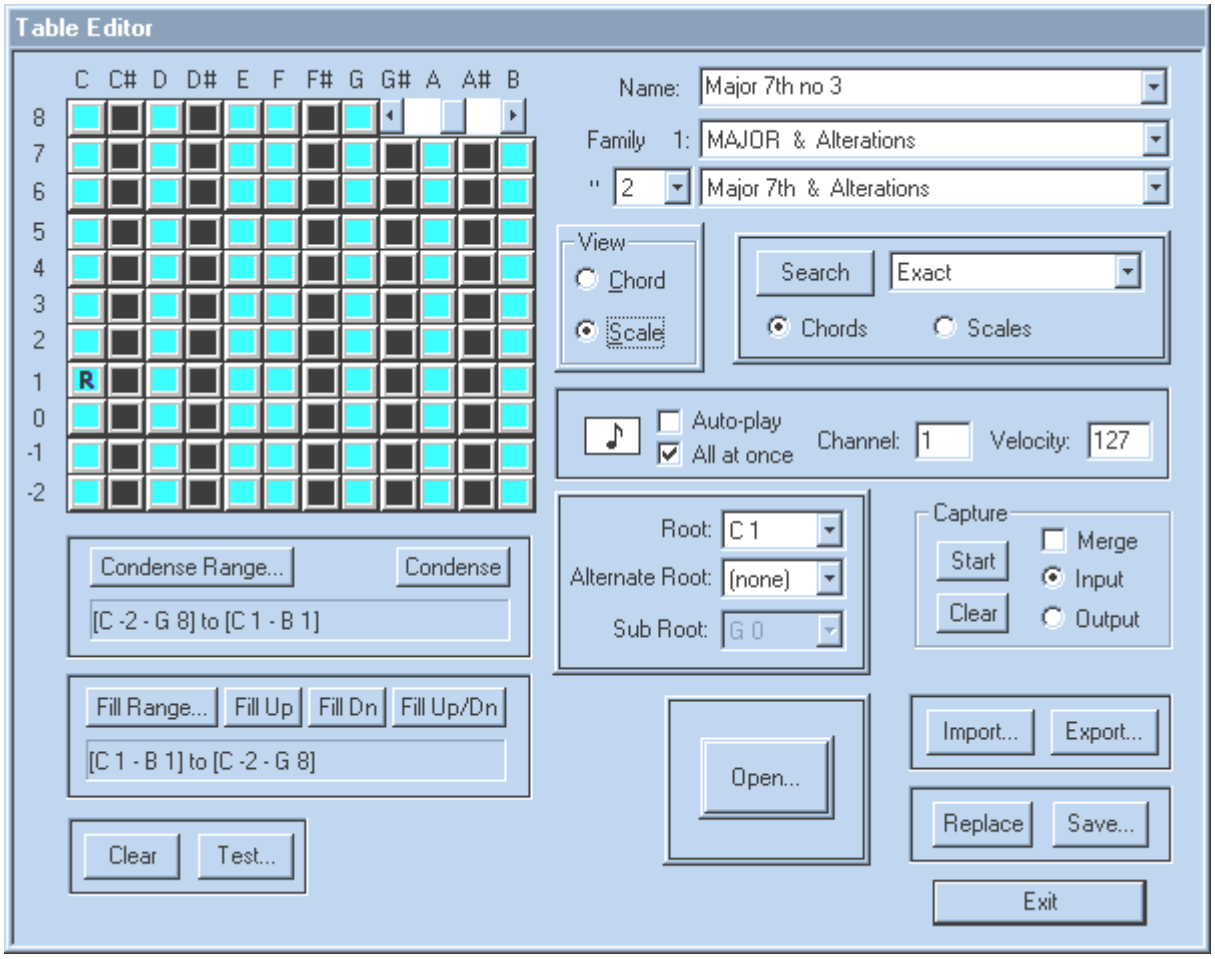


Chapter 6: Menu Buttons

.

Each structure **Table** contains both Chord and Scale data. For a variety of reasons the **Chord data** and **Scale data** may be independently viewed and manipulated.

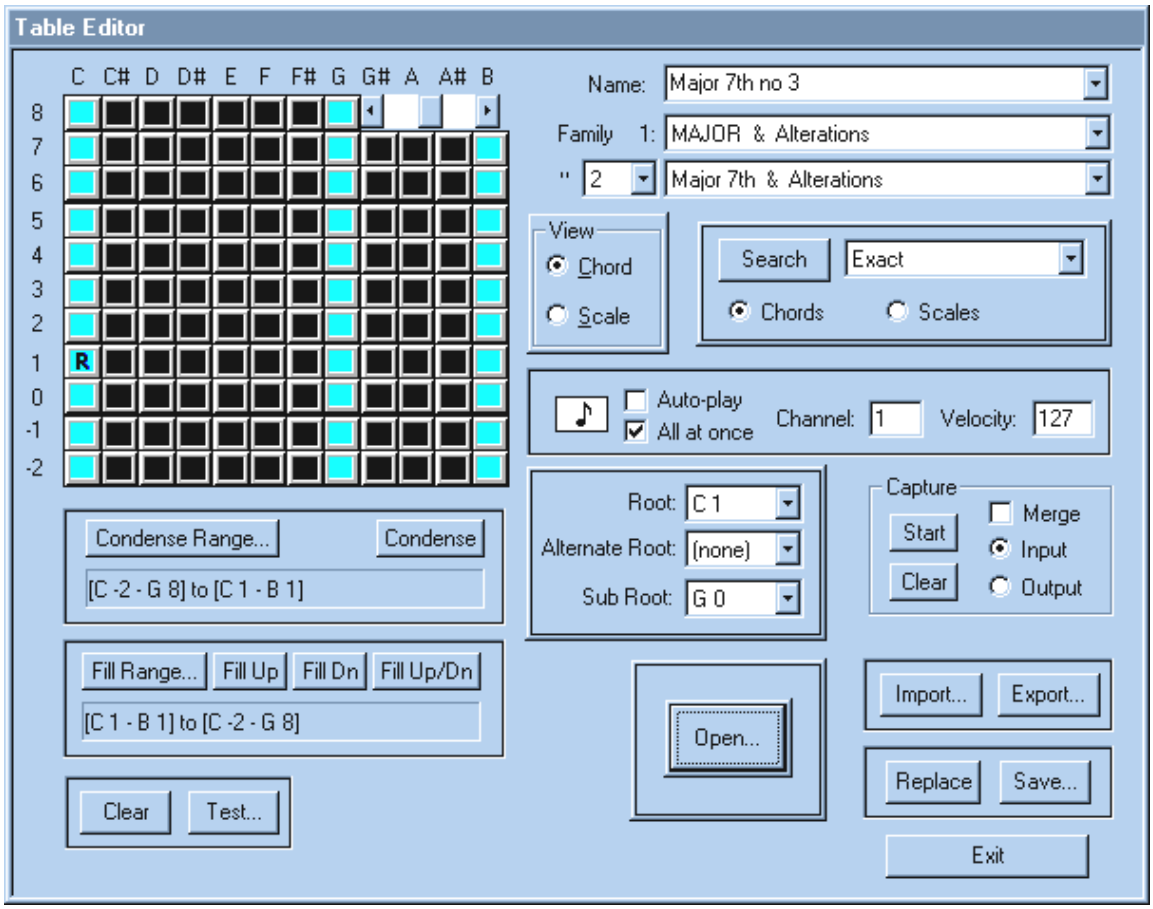
View selects the type of data to be displayed. Select **Chord** to see the chord part of the Table or select **Scale** to see the scale part of the Table. As can be seen in the following examples, both chord and scale sub-tables are displayed throughout the 128-note MIDI range.



All default **Chord/Scale Library** tables are programmed with a universal ROOT point at C1. Key changes are processed separately using KEY note assignments. This programming convention ensures that all data will be translated consistently throughout the various translation mechanisms.

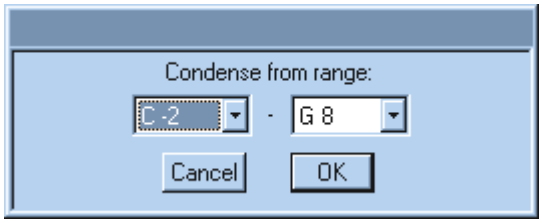
Chords are programmed along with their associated scales. For example, a **Dm9** (**minor 9** table shifted to the root key of **D**) would have the associated scale (D, E, F, G, A, B, C). Scales are programmed along with a chord derived from its' note members. For example, the table for the **Bhairavi That** Indian scale in the key of C (C ,C# , D# ,F ,G , G# ,A#) also contains the derived chord (C, D#, G). This programming standard allows the most efficient use of the Translator tables.

Chapter 6: Menu Buttons

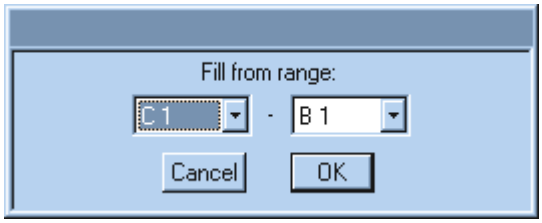


Name: is used to display/edit the structure name. Family 1 (below Name) is for display and editing of the primary family associated with the structure. Below it is a number scroll list that allows you to view/edit the other families associated with the structure (up to eight total, including the primary family).

Condense moves all notes from a selected range into a selected destination range.



Fill copies the pattern of notes in the selected range to a wider range of octaves.

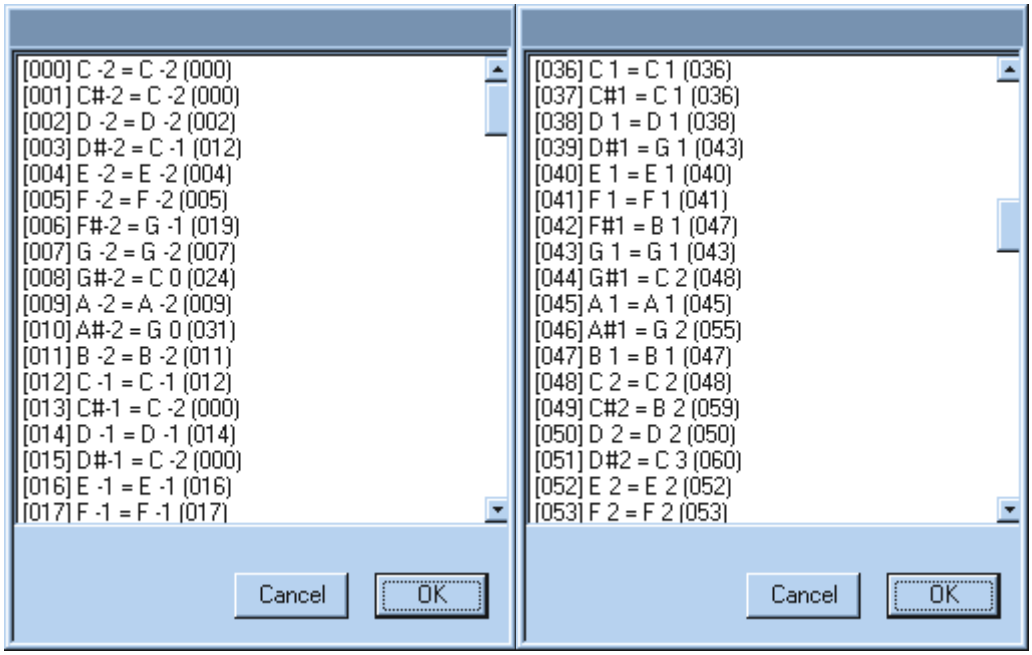


Chapter 6: Menu Buttons

Root is the fundamental reference point of the table that **Key** uses for Transposition.

Alternate Root and **Sub-Root** are desirable related bass notes.

Test shows the pure translation values programmed into the structure table.



Auto Play when selected lets you hear the individual notes as you enter them into the table.

The **Note Icon** when pressed will play the structure either sequentially from lowest to highest or as a chord if **All at once** is selected.

Channel selects the auto-play channel output, and **Velocity** selects its output velocity level.

Capture Input/Output provides a simple way to enter data from your Keyboard controller (input) or Translated data (output) into the Table. **Merge** mixes the captured data with the existing table data.

The **Search** functions apply to only one of the sub-tables, either **Chord** or **Scale**.

Match criteria can be:

Exact (absolutely the same structure).

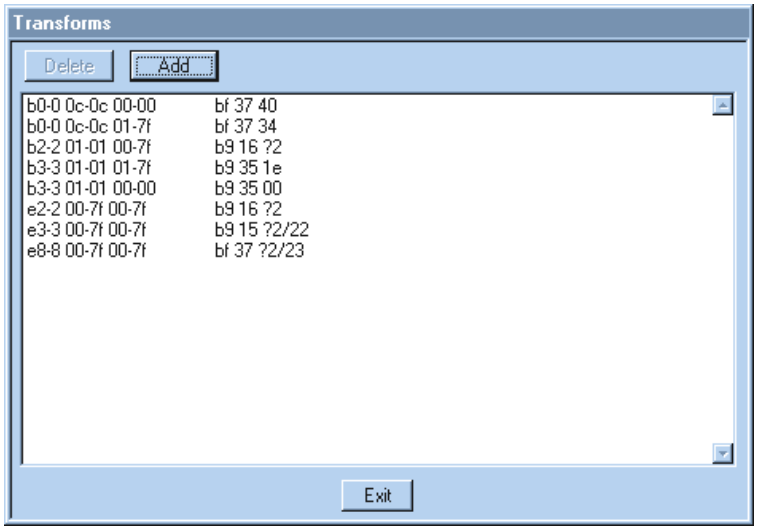
Limited Exact (absolutely the same up to the highest note in the search criteria).

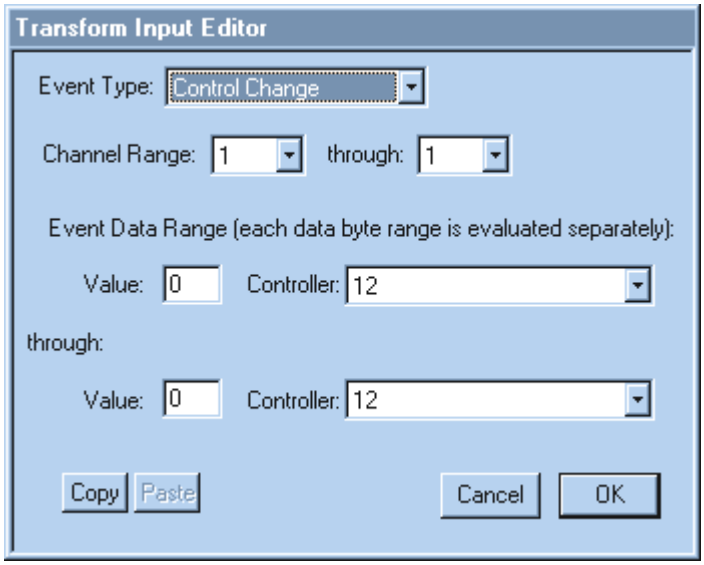
SubSet (structures contain all of the notes of the search criteria but may contain other notes).

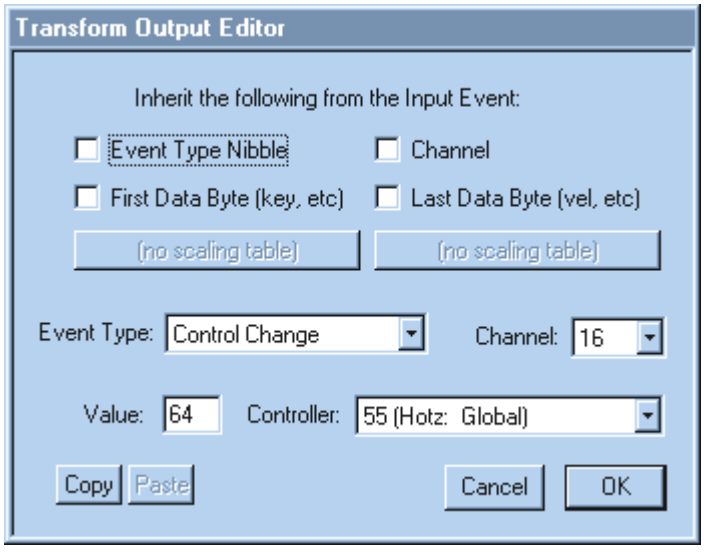
To search for a specific structure click on **Clear** and enter the component notes of what you want to search for. The notes may be entered in any key or octave desired, but you may have better results if you use **C1** as your root and enter the desired structures as closely, and in as few octaves, as possible. The entire table can be shifted, by using the arrows in the upper right hand corner of the note grid.

Chapter 6: Menu Buttons

Transforms (MIDI String Processor)







The MIDI-in HEX data stream is delivered to the **Transform Processor**, which has two roles, depending on whether its output is sent only to MIDI-out or whether it is passed to the **Translation engine**. Although the advanced user can easily modify the function of this processor for limitless customization of MIDI data, the program is preconfigured so that the operation of this processor is invisible to most users. The processor can also direct selected data directly to MIDI out and hence bypass the **Translation engine** altogether. **Transforms** is used to convert a MIDI input instruction to any type of MIDI output instruction and only acts on input data when it is explicitly required. The processor reads the first nibble and determines if action is required. If no processing is required, the data is ignored, hence reducing the already negligible processing time for the entire Translation process.

When the processor receives an “in range” HEX command string (MIDI event), it converts the command to a predefined alternative using a string conversion table. In this way, various commands (such as volume, pitch bend, modulation, transposition and so on) can be created or transformed. Graphical techniques are used to illustrate and control dynamic parameters (velocity, etc.). The software is pre-configured to control strings in a way that the majority of users will be satisfied with. However, for advanced users, the software includes a *MIDI String Table Editor,* which lets the user graphically program string conversions. The screen has a help bar which displays the function of the currently selected input or output string. The user can edit these using a variety of entry methods.

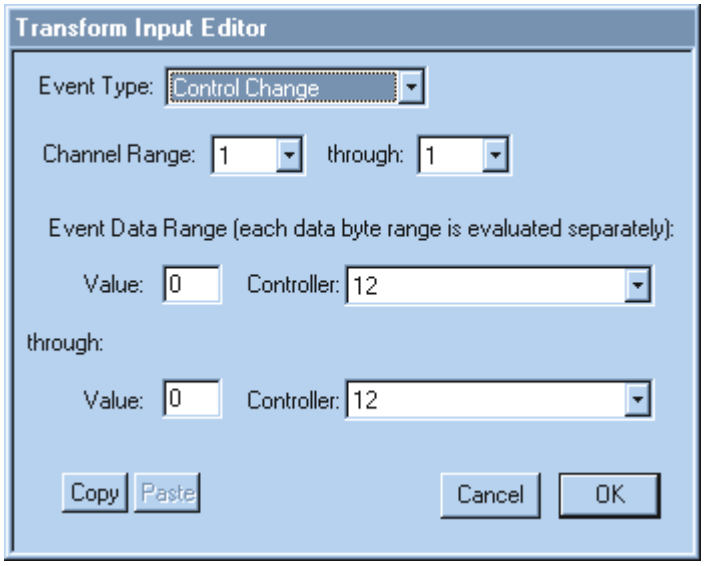
The String Processor can also operate on MIDI note data strings. MIDI note numbers, 0 to 127, are predefined to generate a corresponding note in a MIDI sound generator in the range C-2 to G8. To modify MIDI note events so that they generate the actual tones of some ethnic scales in the database (African, Asian, Indian, and other ethnic types), micro-tonal tuning can be implemented with pitch-control scaling tables.

Chapter 6: Menu Buttons

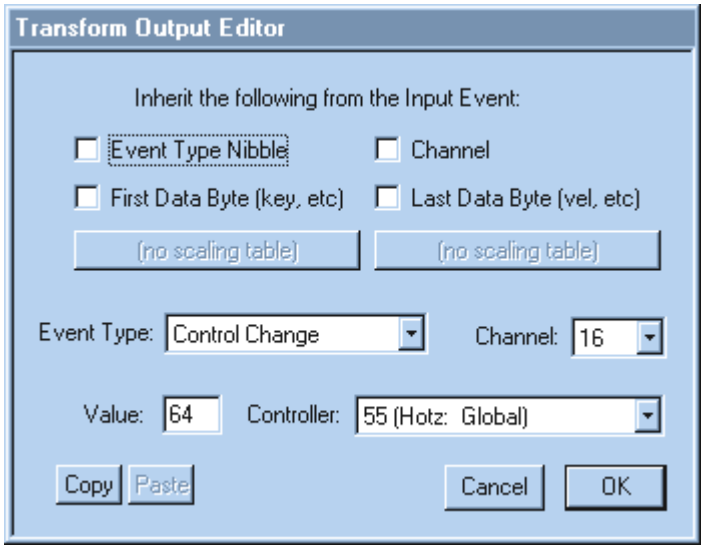
If required, the String Processor uses micro-tonal tuning by applying a predetermined MIDI pitch bend to every note delivered to the processor. The range of pitch bend is defined in a preprogrammed *Scaling Table* associated with an ethnic scale that the translator software is currently set to play. Advanced users of the software can graphically generate their own scaling tables and so create entirely new musical scales that can be recalled and played effortlessly through software MIDI control.

The Transforms feature is an extremely powerful MIDI processor which provides advanced programmers with endless customization options for MIDI input/output control.

The general procedure is to **Add** an input string and edit it using the Transform Input Editor.



Next **Add** an output string next to it (by clicking on the Add button again) and then click on that new string to open the Transform Output Editor.

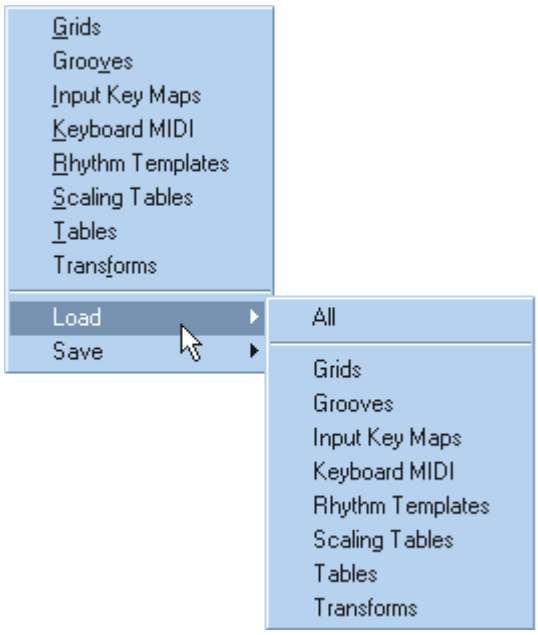


Set the desired output parameters.

If more strings are necessary, continue to add and edit more input and output strings. Using the Transforms feature requires in-depth knowledge of MIDI programming. See also Technical Appendix, page 2.

Chapter 6: Menu Buttons

Load Tables



Load All

Loads all file types listed in the menu.

Load Grids

Loads a bank of 128 Grids.

Load Grooves

Loads a bank of quantization Grooves.

Load Input Key Maps

Loads an Input Key Maps file.

Load Keyboard MIDI

Loads the Keyboard MIDI file (for computer keyboard control).

Load Rhythm Templates

Loads a bank of Rhythm Templates.

Load Scaling Tables

Loads a bank of velocity Scaling Tables.

Load Tables

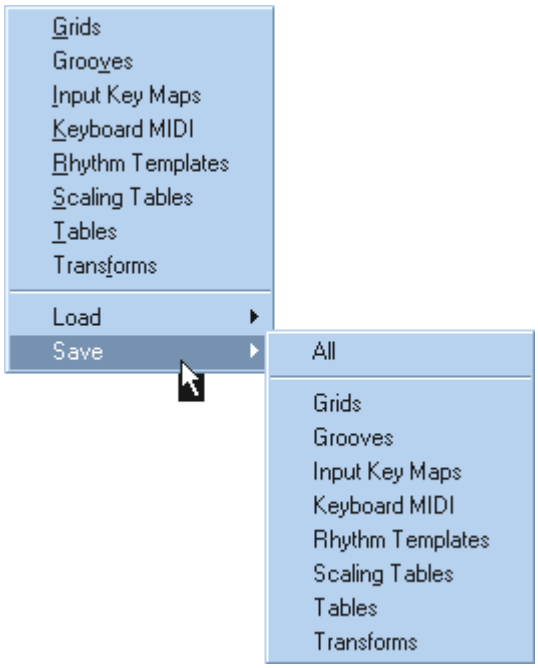
Loads a bank of Structure Tables (Chord and Scale Tables.

Load Transforms

Loads a set of **Transforms**.

Chapter 6: Menu Buttons

Save Tables



Save All

Saves all file types listed in the menu.

Save Grids

Saves a bank of 128 Grids.

Save Grooves

Saves a bank of quantization Grooves.

Save Input Key Maps

Saves an Input Key Maps file.

Save Keyboard MIDI

Saves the Keyboard MIDI file (for computer keyboard control).

Save Rhythm Templates

Saves a bank of Rhythm Templates.

Save Scaling Tables

Saves a bank of velocity Scaling Tables.

Save Tables

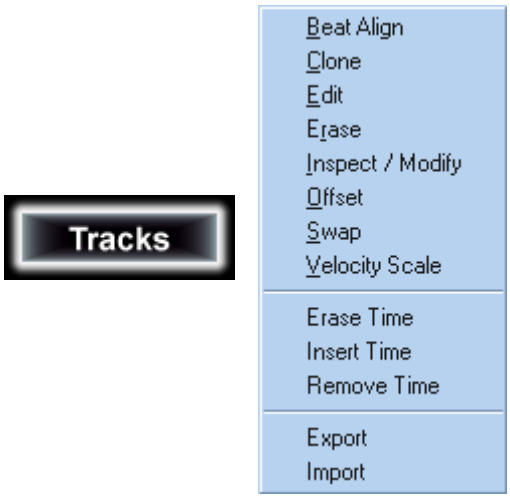
Saves a bank of Structure Tables (Chord and Scale Tables).

Save Transforms

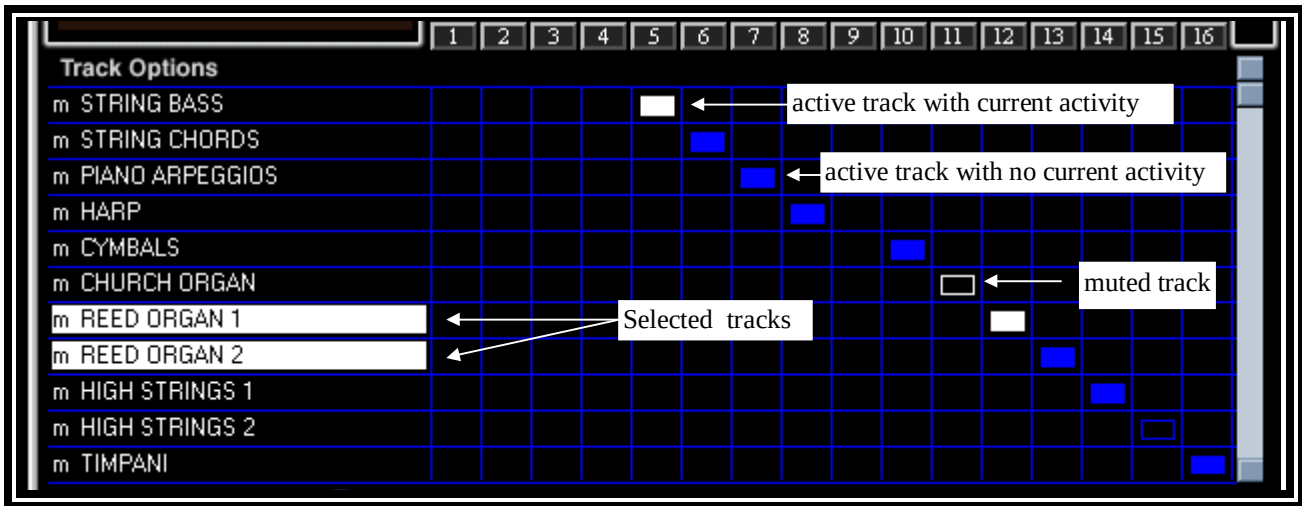
Saves a set of **Transforms**.

Chapter 6: Menu Buttons

Track Menu Commands



The **Track Menu** provides additional commands for manipulating track data. One or more tracks must be selected on the Track window for these commands to function. (See also **Track Editing - Basic Functions** in the **Track Grid** section**.)**



Beat Align

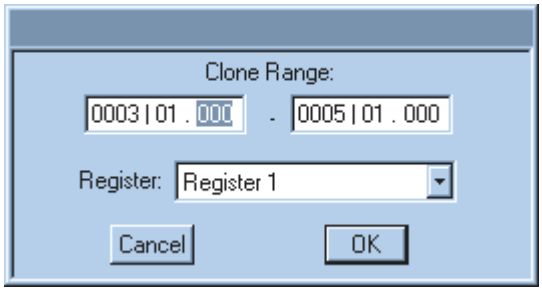
The **Beat Align** feature allows you to record a MIDI track in free time. With tempo turned off, create a sophisticated Tempo map by tapping a reference beat track (on each beat or the first beat of each bar). This feature will give unprecedented power to interact with MIDI files and CD tracks that have fluid Tempo movements. It will even allow you to quantize against a free time track!

Chapter 6: Menu Buttons

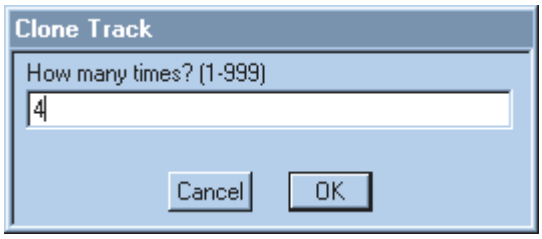
Clone

Clone allows you to make a track much longer. For example, when you clone a 2 bar track 4 times, it becomes 8 Bars. This is especially useful for duplicating patterns, sections, verse, chorus, etc.

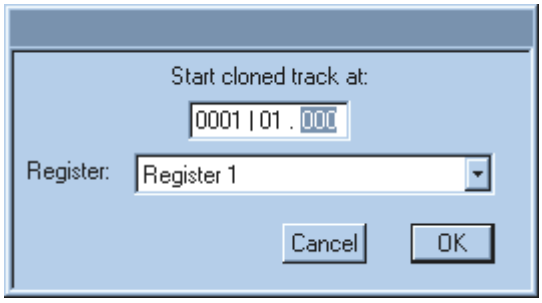
Clone step 1: Select the Range you wish to Clone (Copy).



Clone step 2: How many times should this repeat?

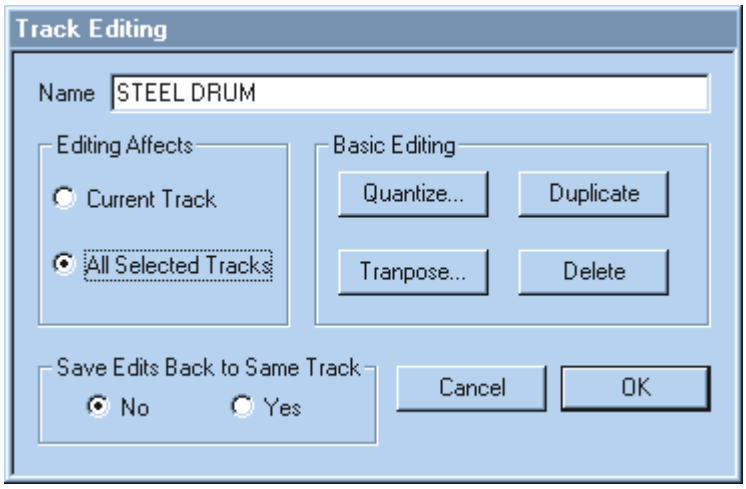


Clone step 3: Where in time should this placed?



Chapter 6: Menu Buttons

Edit



Edit brings up a dialog box that contains a few of the most often used **Track Editing** options. You may also get to this dialog box by double-clicking on a **Track Name** in the **Track Grid**. These options are also covered in the Main Window – Track Grid section.

Delete

Removes the track data from memory.

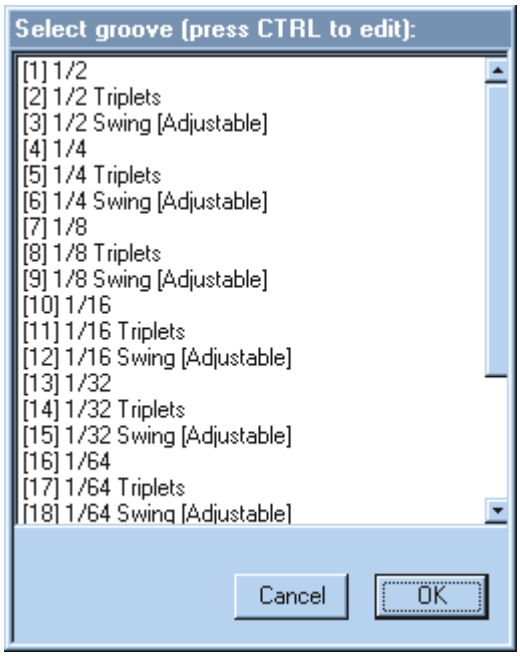
Duplicate

Copies the selected track's data to a duplicate track. The new track has the same name and a Duplicate suffix (it is appended to the end of the Track list, and the original track is automatically muted).

Editing Affects – Option

Either the current track or all selected tracks may be affected. Simply click on your selection.

Quantize



This presents a list of **Groove/Quantize** tables. Either double-click on a **Groove** or select one and press O.K.

These may be edited by pressing "Control" while clicking on the groove name. This will open the **Groove** window in which the Quantize parameters are defined. Refer to **Grooves** in the **Tables Menu.**

Chapter 6: Menu Buttons

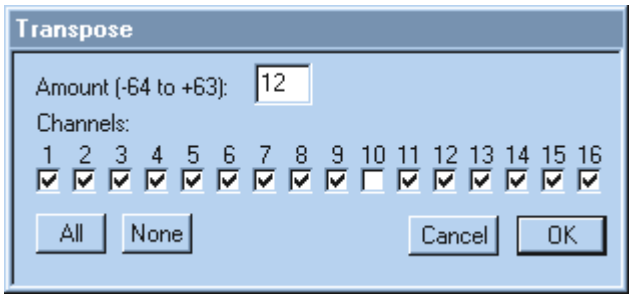
Save Edits Back to the Same Track – Option

Selecting Yes will overwrite the current tracks with the edited data. Selecting No will mute the current tracks and save the edited data to new tracks.

Track Name

Track names are automatically generated when the tracks are created. They may be edited by conventional means.

Transpose (Track)



Transpose (Tracks) is used to transpose all note and/or HotzCode data on the selected tracks. An X under a channel number indicates that all data on that channel will be transposed.

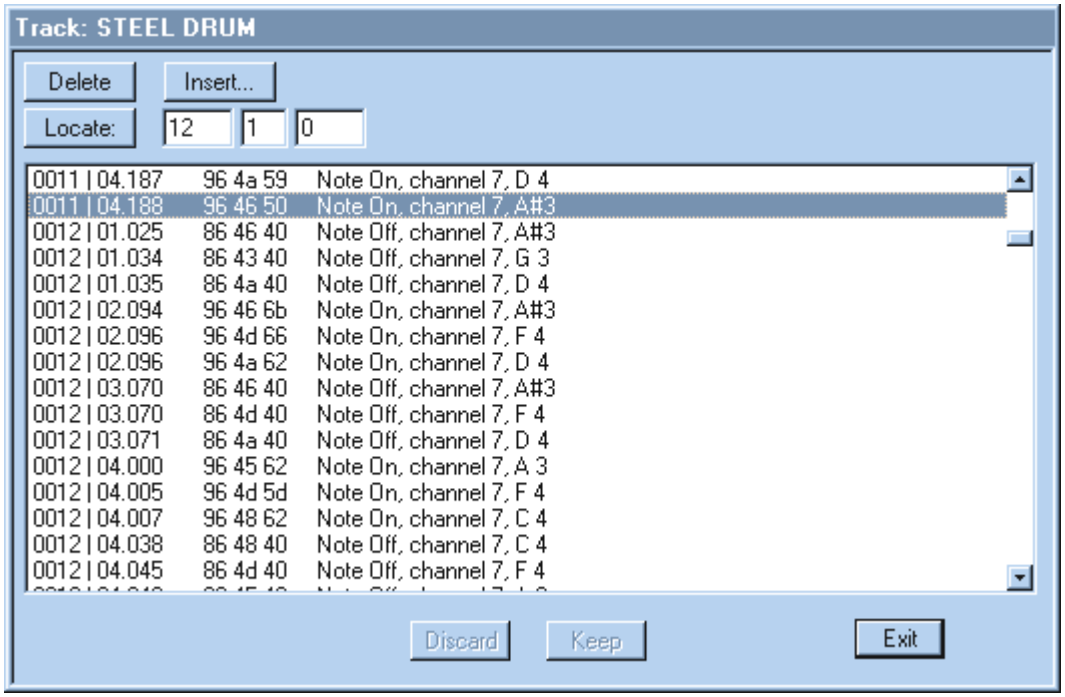
The amount (-64 to +63) refers to semitones.

(12 semitones equal an octave)

Erase

Erase clears all data from the selected track(s).

Inspect / Modify (Track)



Chapter 6: Menu Buttons

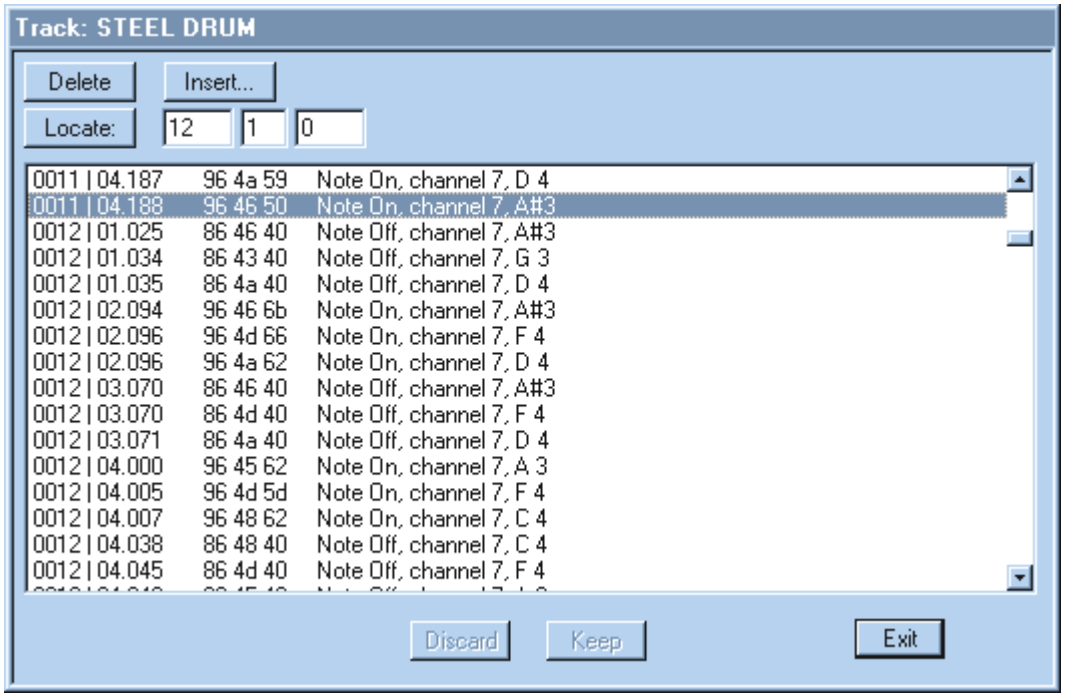
Inspect is a simple list editor that is useful for editing specific events (until a more advanced graphic editing utility is integrated in a later version of the software).

Delete removes a selected event, while **Insert** adds a copy of the selected event at the current location.

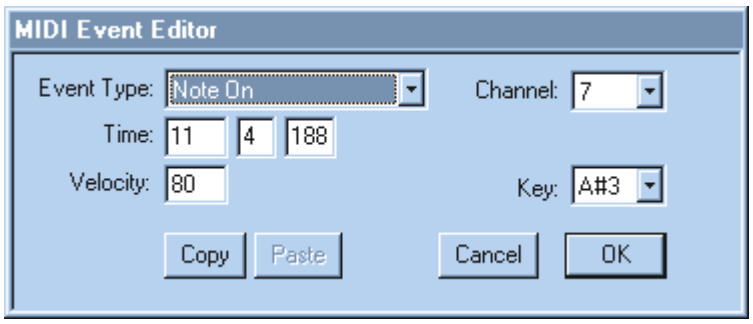
Locate finds the first occurrence of the specified time value.

Double-click on an event to enter the **MIDI Event Editor** (see next page). **Keep** retains editing changes, **Discard** removes them.

MIDI Event Editor



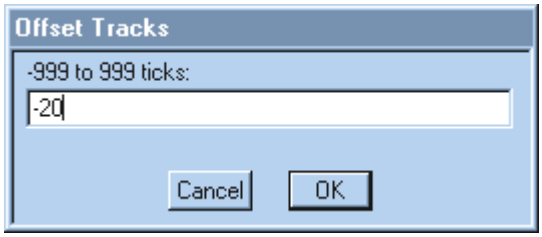
The **MIDI Event Editor** is used when specific events need to be edited. It is of most use to those who are very familiar with MIDI and its various event types. Click on Event Type to select the desired type and then set the rest of the values presented in the parameter window for that type of event.



For example, if it is a note event, set the channel, velocity, time location, and note/Key values. Each event type has a unique parameters window, too many to list independently.

Chapter 6: Menu Buttons

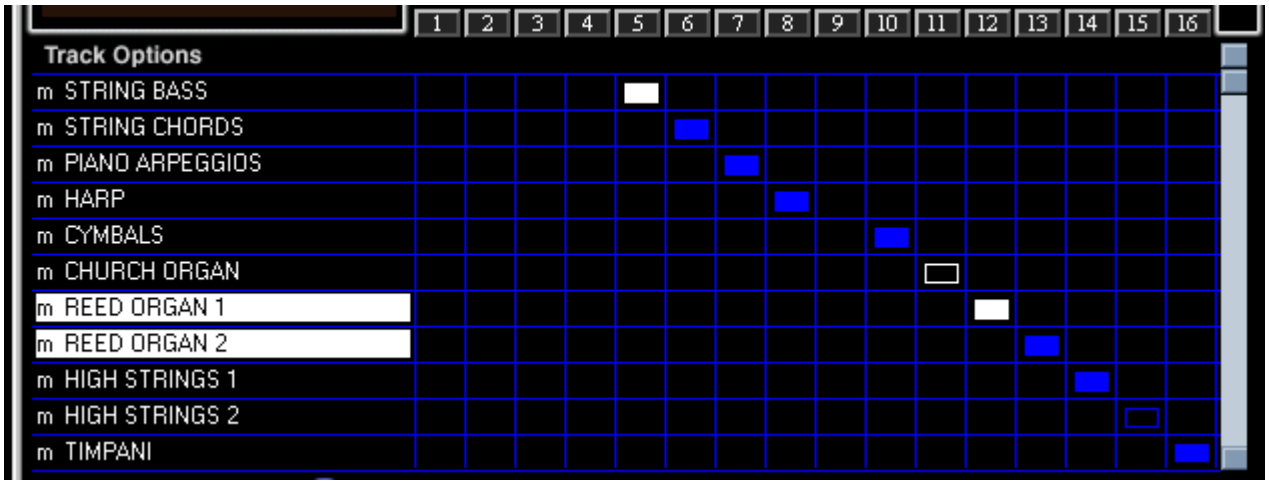
Offset



Offset Tracks is used to slide all data on one or more tracks, either forward or backward. This is especially useful if you have a controller that triggers slowly and you want to shift the data forward to compensate.

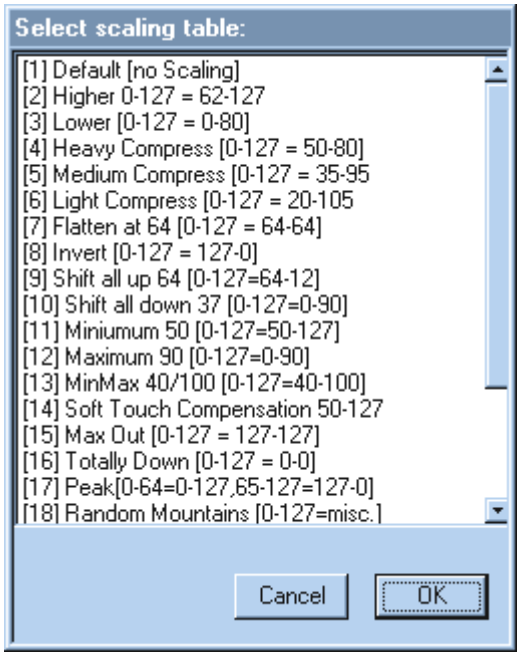
Swap

Select any two tracks by shift left-clicking and then choose the **Swap** command to swap their locations.



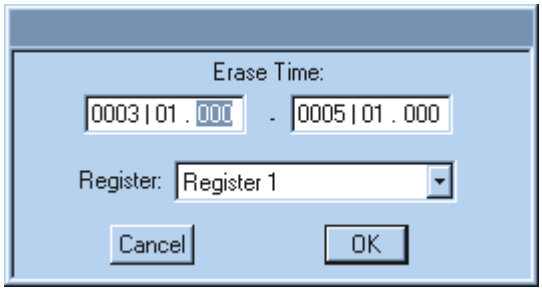
Velocity Scale

To transform a track or tracks with a **Velocity Scaling Table,** select the tracks and then choose **Velocity Scale.** Select a **Velocity** table by clicking on one of the available tables from the pop-up menu.



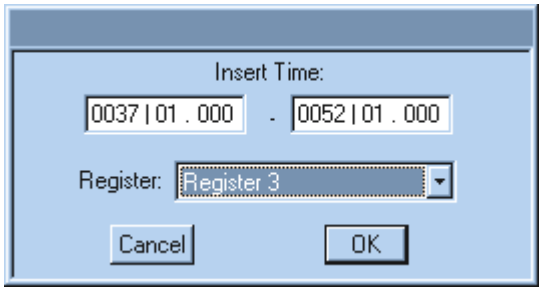
Chapter 6: Menu Buttons

Erase Time



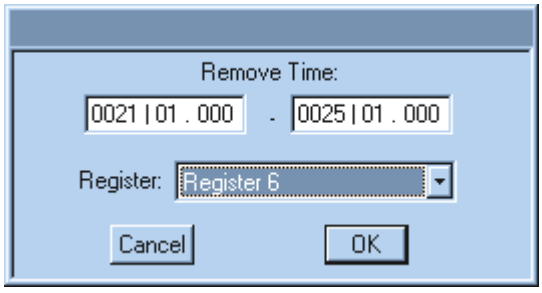
Erase Time functions exactly like the previous Time features, but it is used to erase a section from all selected tracks. Unlike the Remove feature, Erase leaves the cleared section in the track(s).

Insert Time



The **Insert Time** feature allows you to insert a blank section into selected tracks. If you use Registers to organize your song into sections, click on the desired Register and its values will become the insert values.

Remove Time



The **Remove Time** feature allows you to cut a section from selected tracks. If you use Registers to organize your song into sections, click on the desired Register and its values will become the Remove Time values.

Export (Track)

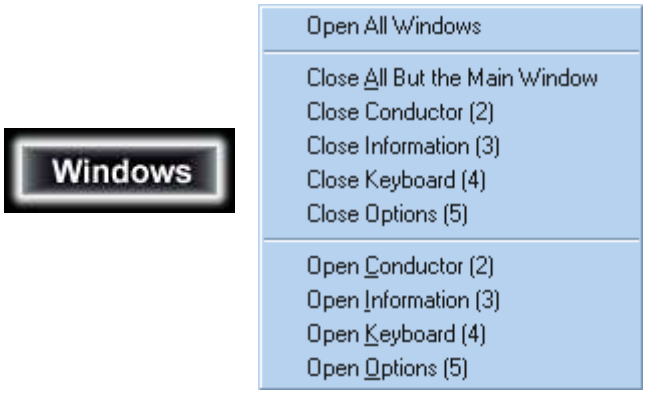
Export (Track) is used to export a single track as a .MID file.

Import (Track)

Allows users to **import** .MID tracks into an open song. If you wish to retain volume and patch settings you should enable **Automation** before importing.

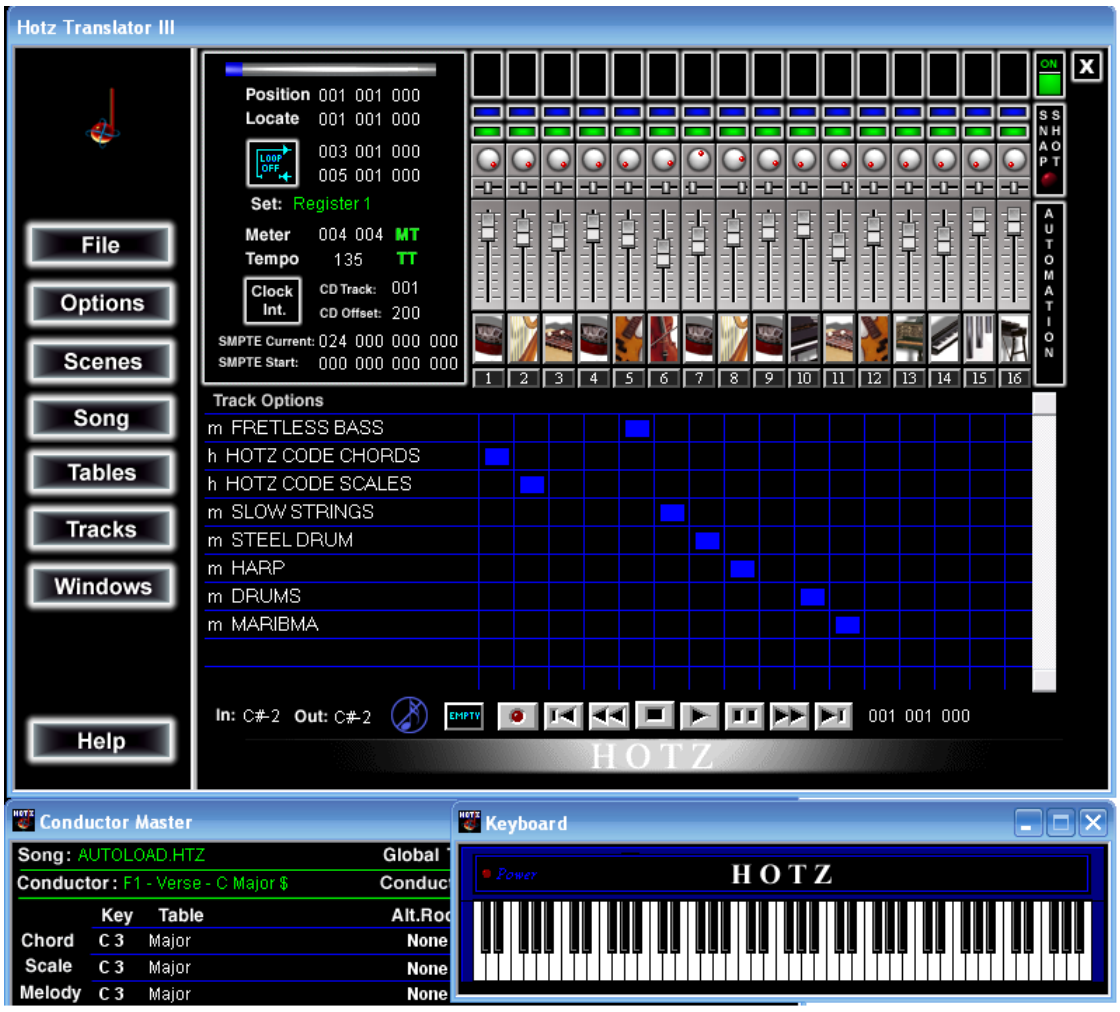
Chapter 6: Menu Buttons

Windows Menu



Open All Windows

Opens the initial windows layout.



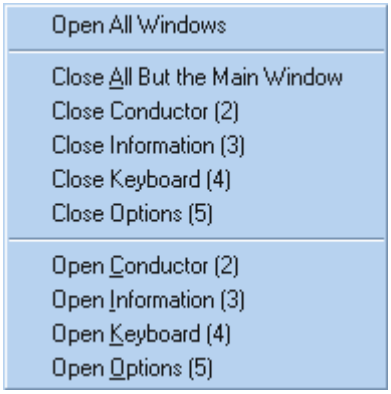
Chapter 6: Menu Buttons

Close All but the Main Window

The **Main Window** may not be closed without exiting the program. Selecting **Close All but the Main Window** will close all of the Windows except the **Main Window.**

These items are self-explanatory:

Close Conductor Window
Close Information Window
Close Keyboard Window
Close Options Window
Open Conductor Window
Open Information Window
Open Keyboard Window
Open Options Window

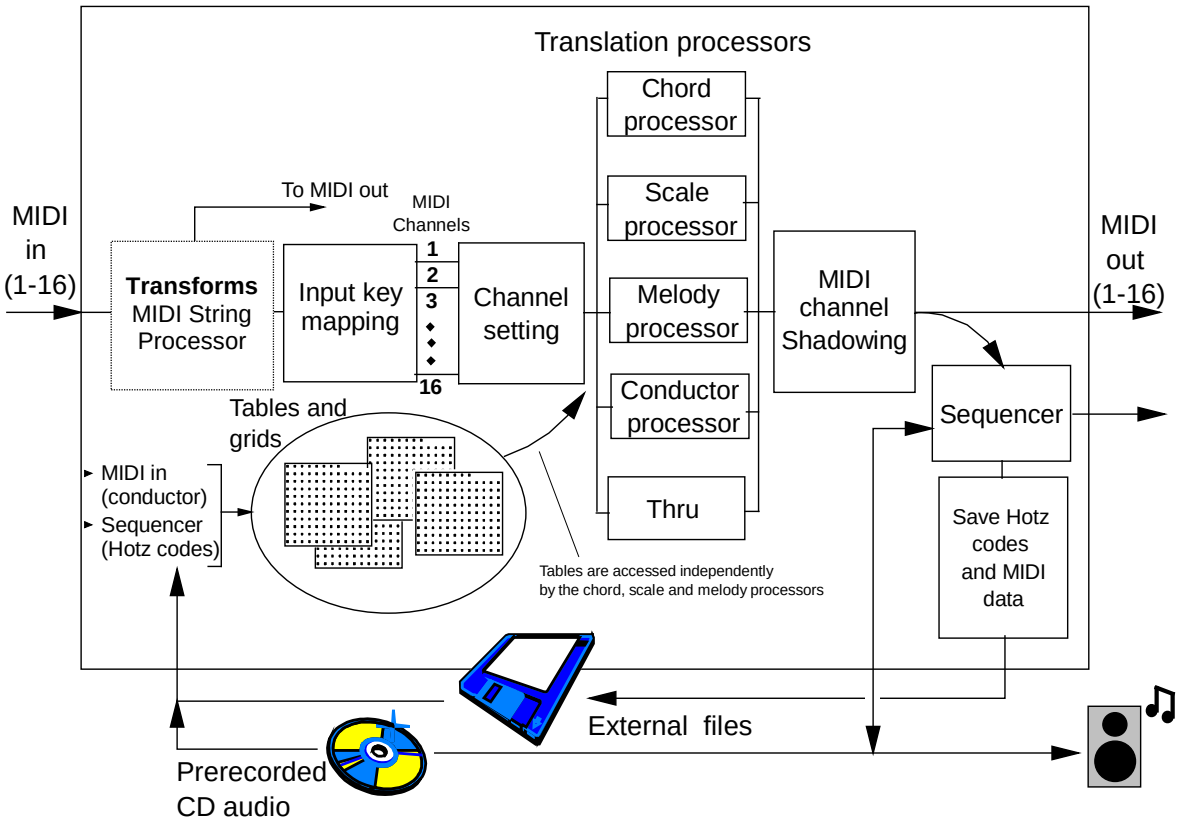


Keyboard Window Information

The **Keyboard** represents note output from the Translation engine as it would appear on a traditional keyboard. *The input and output notes are also shown simultaneously in the "Main Window" to the left of the Transport and in the "Conductor Window".*

Chapter 7: Hotz Translation Processing Overview

The advanced features of the Translator can be best understood by reviewing the passage of MIDI signals through the various software modules, or processors. The time taken for the signal to pass through the Translator is less than the time required to read an input MIDI event. The Translation process, therefore, has no discernible effect on timing. The following figure describes data flow through the software.

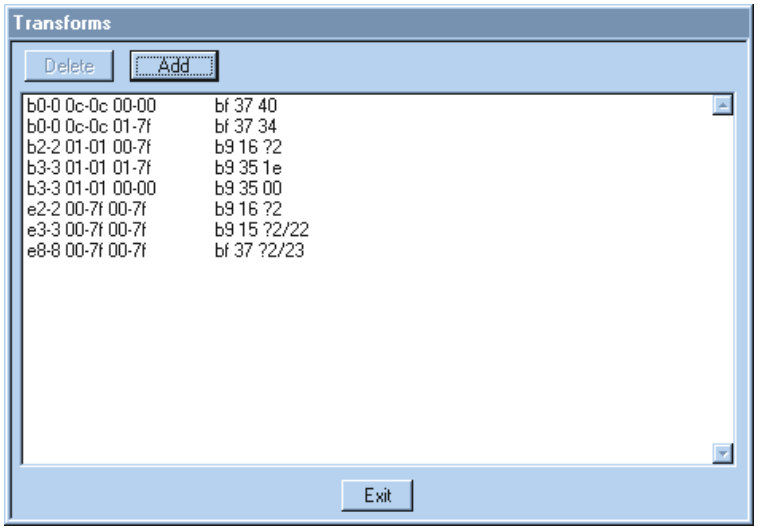


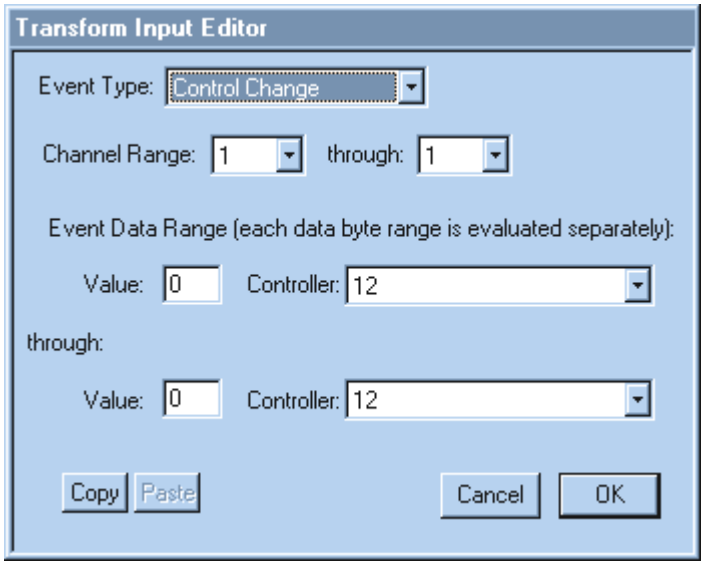
MIDI In

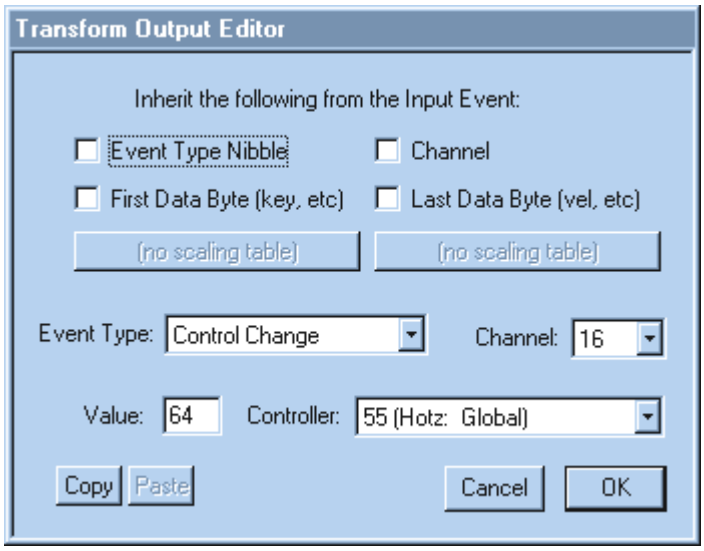
MIDI data is delivered to the **Translator** from any MIDI source, including MIDI-compatible instruments/controllers, Hotz MIDI controllers, or the computer keyboard. MIDI instruments that can be zoned can deliver MIDI data on any channel from one through sixteen. Input note and channel values can also be remapped within the software (**Input Key Maps** feature allows users to zone any MIDI input controller). The **Translator** also zones the computer keyboard keys. Using the **Keyboard MIDI** menu function, every computer keyboard key *up or* key *down* action can define a unique MIDI data sequence on any channel. This includes the keyboard's letter, number, and function keys (all with Shift and Control options). The data sequence could be more versatile than that delivered by a standard MIDI controller and can include note numbers and controls (velocity scaling, etc.).

Chapter 7: Hotz Translation Processing Overview

The Transforms MIDI String Processor







The MIDI-in HEX data stream is delivered to the **Transform Processor**, which has two roles, depending on whether its output is sent only to MIDI-out or whether it is passed to the **Translation engine**. Although the advanced user can easily modify the function of this processor for limitless customization of MIDI data, the program is preconfigured so that the operation of this processor is invisible to most users. The processor can also direct selected data directly to MIDI out and hence bypass the **Translation engine** altogether. **Transforms** is used to convert a MIDI input instruction to any type of MIDI output instruction and only acts on input data when it is explicitly required. The processor reads the first nibble and determines if action is required. If no processing is required, the data is ignored, hence reducing the already negligible processing time for the entire Translation process.

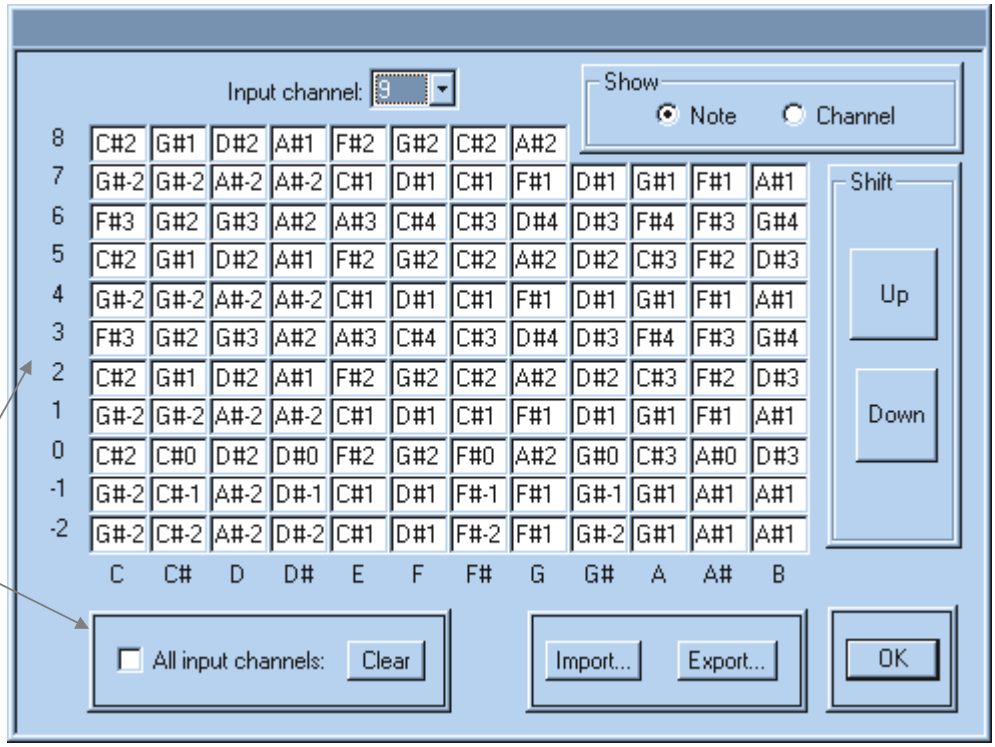
When the processor receives an “in range” HEX command string (MIDI event), it converts the command to a predefined alternative using a string conversion table. In this way, various commands (such as volume, pitch bend, modulation, transposition and so on) can be created or transformed. Graphical techniques are used to illustrate and control dynamic parameters (velocity, etc.). The software is pre-configured to control strings in a way that the majority of users will be satisfied with. However, for advanced users, the software includes a *MIDI String Table Editor*. This lets the user graphically program string conversions. The screen has a help bar, which displays the function of the currently selected input or output string that the user can edit using a variety of entry methods.

The String Processor can also operate on MIDI note data strings. MIDI note numbers, 0 to 127, are predefined to generate a corresponding note in a MIDI sound generator in the range C-2 to G8. To modify MIDI note events so that they generate the actual tones of some ethnic scales in the database (African, Asian, Indian, and other ethnic types), micro-tonal tuning can be implemented with pitch-control scaling tables.

Chapter 7: Hotz Translation Processing Overview

If required, the String Processor uses micro-tonal tuning by applying a predetermined MIDI pitch bend to every note delivered to the processor. The range of pitch bend is defined in a preprogrammed *Scaling Table* associated with an ethnic scale that the translator software is currently set to play. Advanced users of the software can graphically generate their own scaling tables and so create entirely new musical scales that can be recalled and played effortlessly through software MIDI control.

Input Key Mapping



Outside Note Values represent incoming MIDI notes; inside notes correspond to outgoing note values

As with the String Processor, the operation of this module is typically invisible to most users. The benefit of this module to the new user is that it keeps the translation process simple and logical from the very start. However, the advanced user can control input key mapping functions to create an extraordinary and uniquely powerful level of MIDI customization.

This section of the Translator software, by operating jointly on channel data and note data, has two primary functions: MIDI Channeling and keyboard re-zoning. When the **Input Key Map** operates on note values, it changes any input note on any channel to any other selected note value. When operating on channel data, an Input Key Map is used to route any channel, or any selected note on that channel, to any other channel. Any note can be assigned to any channel. At the next stage, the internal channels are variously routed to the chord, scale, melody, and conductor processors. These processors then use the structure *tables* to translate note values associated with the current scale and chord, and map these values to the various input key/trigger zones.

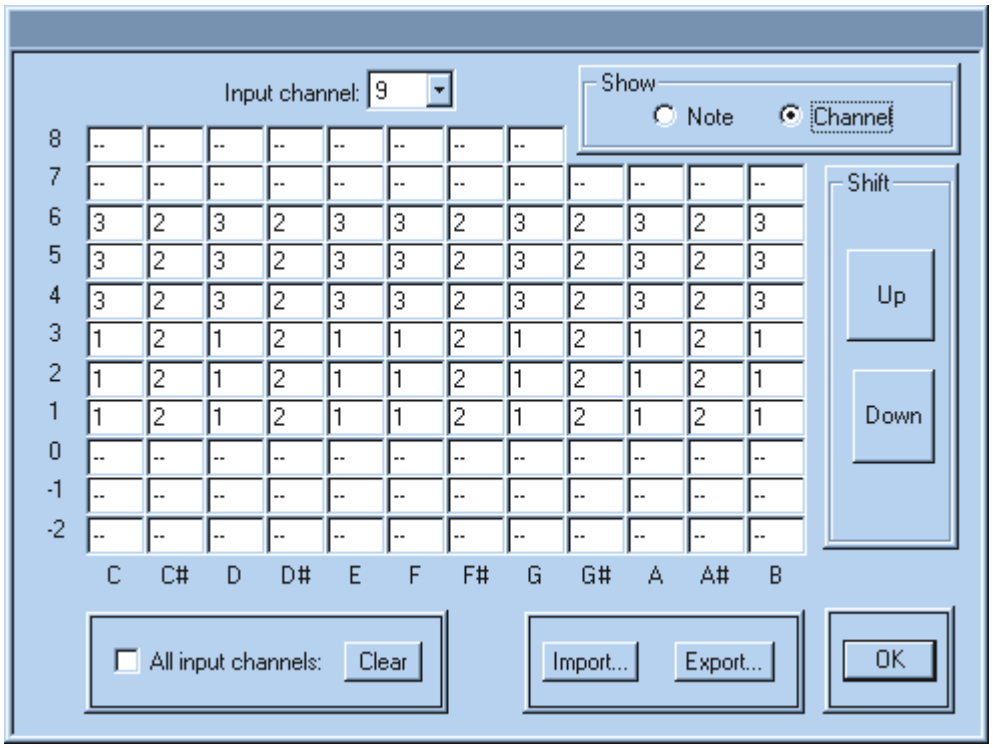
The effect of input key mapping is to customize the MIDI controller's output so that various groups of keys can be then assigned to specific functions: These can be scale notes, notes of the chord or its associated scale, percussion, MIDI controls, and so on. Input Key Mapping is a critical first step in the translation process because it configures the keyboard in a way that prepares translation in a clear and logical manner. For example, all the white notes (C^1, D^1, E^1, F^1, G^1, A^1, B^1, etc.) input on channel 1 can be routed to, say, channel 4 internal to the software. Additionally, all the black notes ($C\#^1$, $D\#^1$, $F\#^1$, $G\#^1$,

Chapter 7: Hotz Translation Processing Overview

A#[1], C#[2], etc.) can be routed to internal channel 5. These re-routings may cover all channels and the entire note range of C-2 to G8. In this example, channel 4 may then be processed through the scale processor, and channel 5 through the chord processor. By selecting the notes and then channeling them in this way, the subsequent processors can efficiently map the notes of the current music structure. In this example, the scale notes would be accessed by the white notes and the chord notes by the black notes. Later in the translation process, scale and chord tables assign the note values associated with the current music structure to the respective keys.

The mapping of white keys to scale notes, and black keys to chord notes is the Translator's default setting and is expected to be used by most players (since most scales are 7-note structures, and most chords are 3-5 note structures, this is a very efficient programming standard). For the more experienced user, the software provides a powerful, simple to edit mapper that operates on the 16 (channel) by 128 (note value) environment. Input Key Maps can be customized to pick out discrete note ranges, or even single keys, on an input channel and map these to another channel and/or note internal to the software. For example, all the white notes in the C^0 octave could be set to channel 8 and all the black notes in the C^3 octave could be set to channel 9, with the single key $G\#^3$ mapped to channel 15.

This freedom to re-map input layouts offers tremendous versatility for the advanced user. Custom keyboard layouts can be generated for each player or song, and may be changed between song performances. The user programs a keyboard layout that suits their playing style or the needs of the music to be performed. Scale and chord notes can be set up in various octaves for the left, right hand, or both. Bass root notes can be placed conveniently under your thumb. High and low notes can be reversed in order. Linear runs can be preconfigured. Octave or multi-octave intervals can be reached with two fingers.

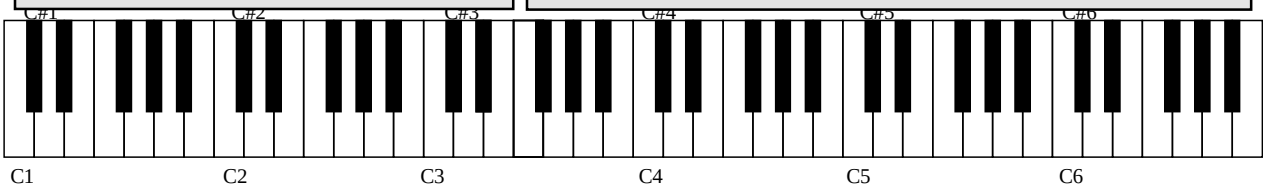


Chapter 7: Hotz Translation Processing Overview

An Input Key Map zoning example

Black Notes Zone C#1-D#3
outputs on
ch. 16 (Conductor), to be used for
Cue List Pointers:
C-2, D-2, E-2, F-2...G-1

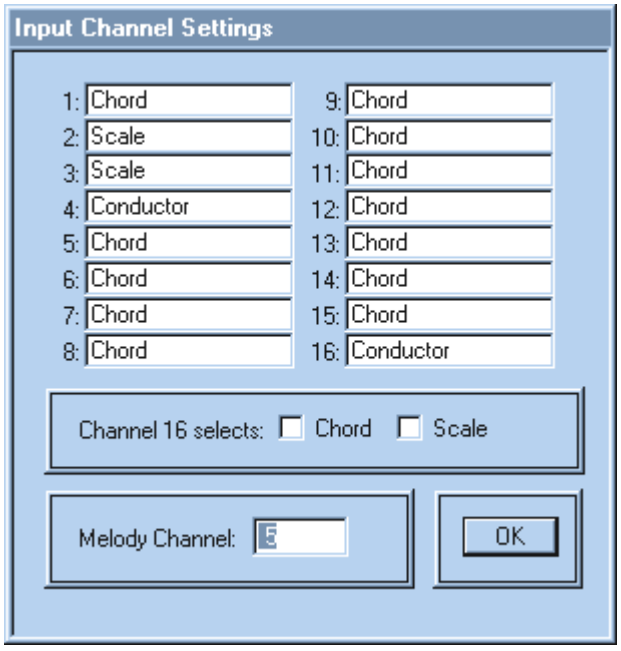
Black Notes Zone F#3-A#6
outputs on
ch. 16 (Conductor)
to be used for Conductor Notes:
C1, D1, E1, F1, G1, A1...F3



White Note Zone C1-B2
Outputs on Ch. 11 (Chord), to be used for chord notes of the Chord processor:
C#1, D#1, F#1....

White Note Zone C3-B6
Outputs on Ch. 12 (Scale),
to be used for scale notes
of the Scale processor:
C1, D1, E1, F1, G1.....

Input Channel Settings



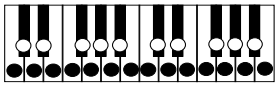
This module receives the MIDI data after it has passed through the **Transforms** (string processor) and **Input Key Maps**. Now the individual channels are directed to the five Translation Table processors: chord, scale, melody, conductor, or thru. The software features a simple menu that lets the user assign any channel to any processor. For example, channels 1, 2, 7 and 8 may be set to chord. In this case, the zones of the input controllers that are set by the Input Key Map to channels 1, 2, 7, and 8 will play the *chord* notes associated with the current scale. Similarly, channels 4 and 5 may be set to *scale*, and 9 and 10 set to *melody*. In whichever way the channel settings assign the channels, the input keys tied to that channel will perform the associated Table processing (as the following sections describe).
Note! *Channel 1 should not be assigned to Conductor as this could cause a MIDI feedback loop under certain conditions.*

Chapter 7: Hotz Translation Processing Overview

Scales and Chords

To understand the nature of the core translation process it is helpful to review some simple scale and chord structures. All Western musical structures are a subset of the chromatic 12-tone scale, shown here as a span of two octaves.

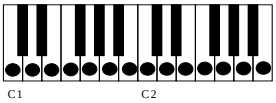
Chromatic 12-Tone Scale



From this scale, characteristic intervals are selected to create the tremendous range of basic and exotic scales used in all classical, jazz, blues, pop, and other western music.

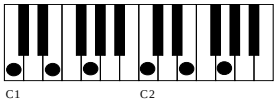
For example, the C major scale across two octaves is shown below, and related triad harmony chords derived from the scale are shown beneath it.

C Major scale:

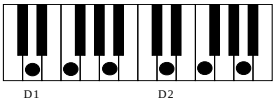


C Major Key Chords:

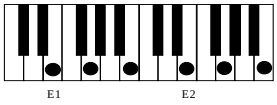
C Major Chord



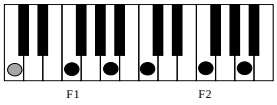
D Minor Chord



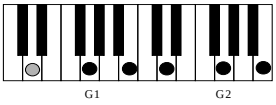
E Minor Chord



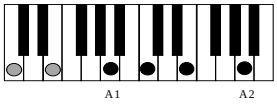
F Major Chord



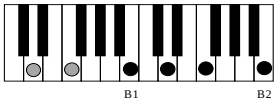
G Major Chord



A Minor Chord

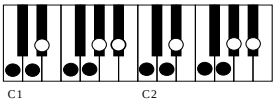


B Diminished Chord



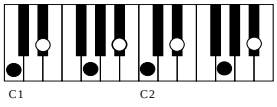
The C minor scale and some chords derived from the scale are illustrated below.

C Minor Scale:

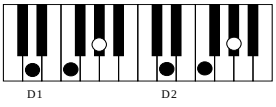


C Minor Key Chords:

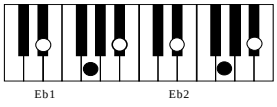
C Minor Chord



D Diminished

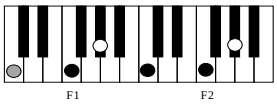


Eb Major



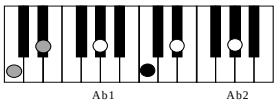
Chapter 7: Hotz Translation Processing Overview

F Minor

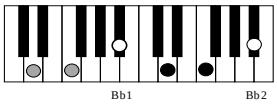


G Minor

Ab Major



Bb Major



The preceding examples show just a few scale and chord patterns. A good, basic musical performance capability would require the player to instantaneously recall these and *many, many more* structures, with all their associated fingerings. The player then has to flawlessly reproduce the correct fingering while focusing on rhythm, velocity, and stylistic aspects. This demanding requirement takes years of practice to develop and retain. Note too, the number of notes that are *not* required to play a scale, and particularly a chord.

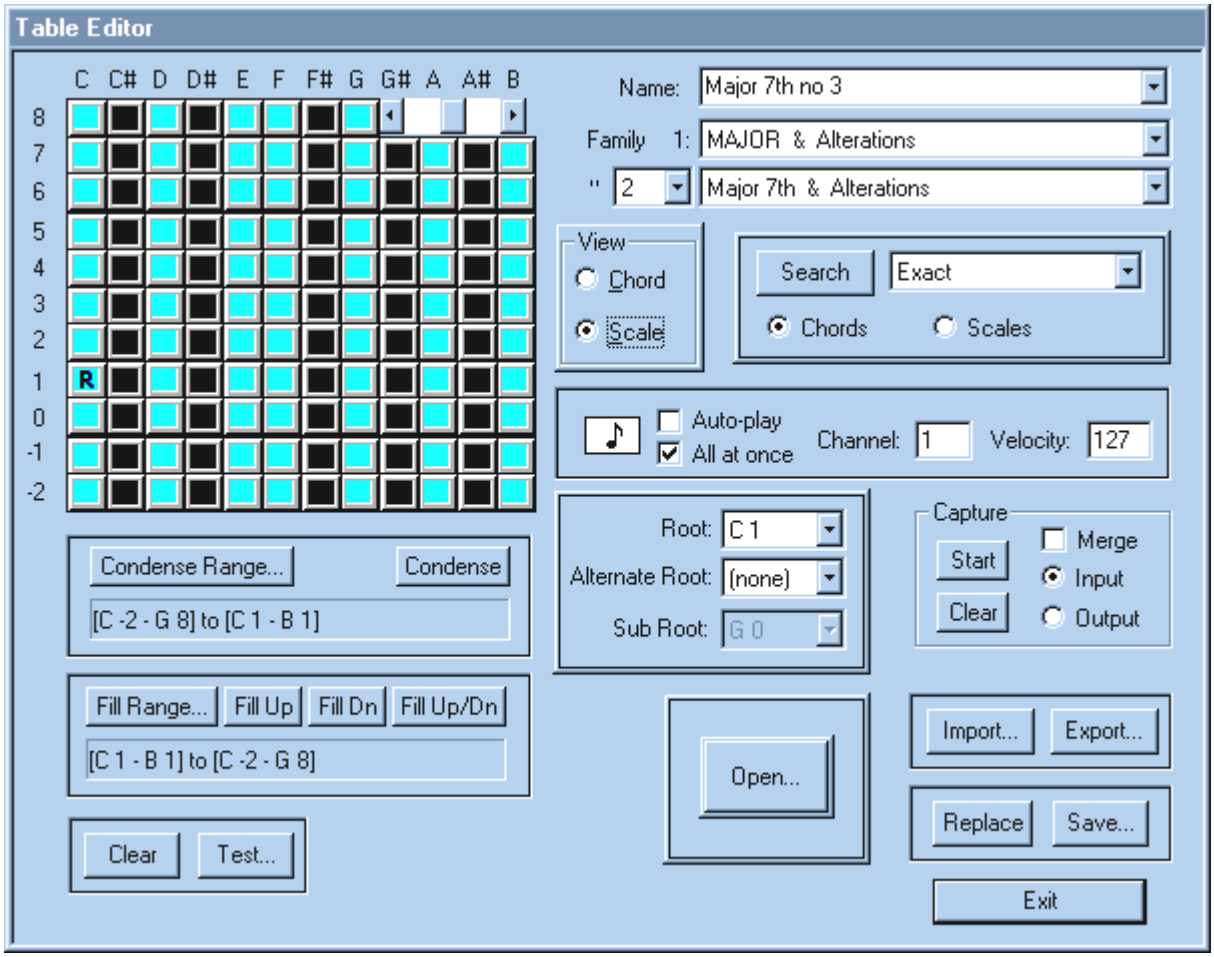
What is so significant about the Hotz Translation approach to playing musical structures is that only the component notes of the current scale and its chords are available on the keyboard at any one time. The keyboard becomes 100% efficient. Only the component notes of the current musical structure are presented to the player, and they are all condensed into the smallest possible region of the keyboard. They are available in all possible modes and voicings, in a way that even a master musician would be hard-pressed to play on a regular keyboard. Intervals and voicings that are physically impossible on a conventional polyphonic instrument, whether a keyboard or a guitar, are easily achieved on a translated keyboard. And the user can customize the keyboard for even more convenient playing of wide voicings if style or the requirements of the music demand it.

Chapter 7: Hotz Translation Processing Overview

Chord and Scale Tables

Once MIDI data has been routed by the channel setting to the various processors, the dynamic translation process comes into play. The underlying database contains the largest collection of musical structures assembled in a single reference. This information is organized in *tables* which are accessed independently by the Chord, Scale, Melody, and Conductor processors. The software has thousands of pre-built tables that can deliver literally all voicings of programmed scales and chords, rendering editing of these tables unnecessary. However, for the advanced user, the ability to edit or create new tables is provided. Entirely new scales and chords can thus be developed and integrated into the knowledge base.

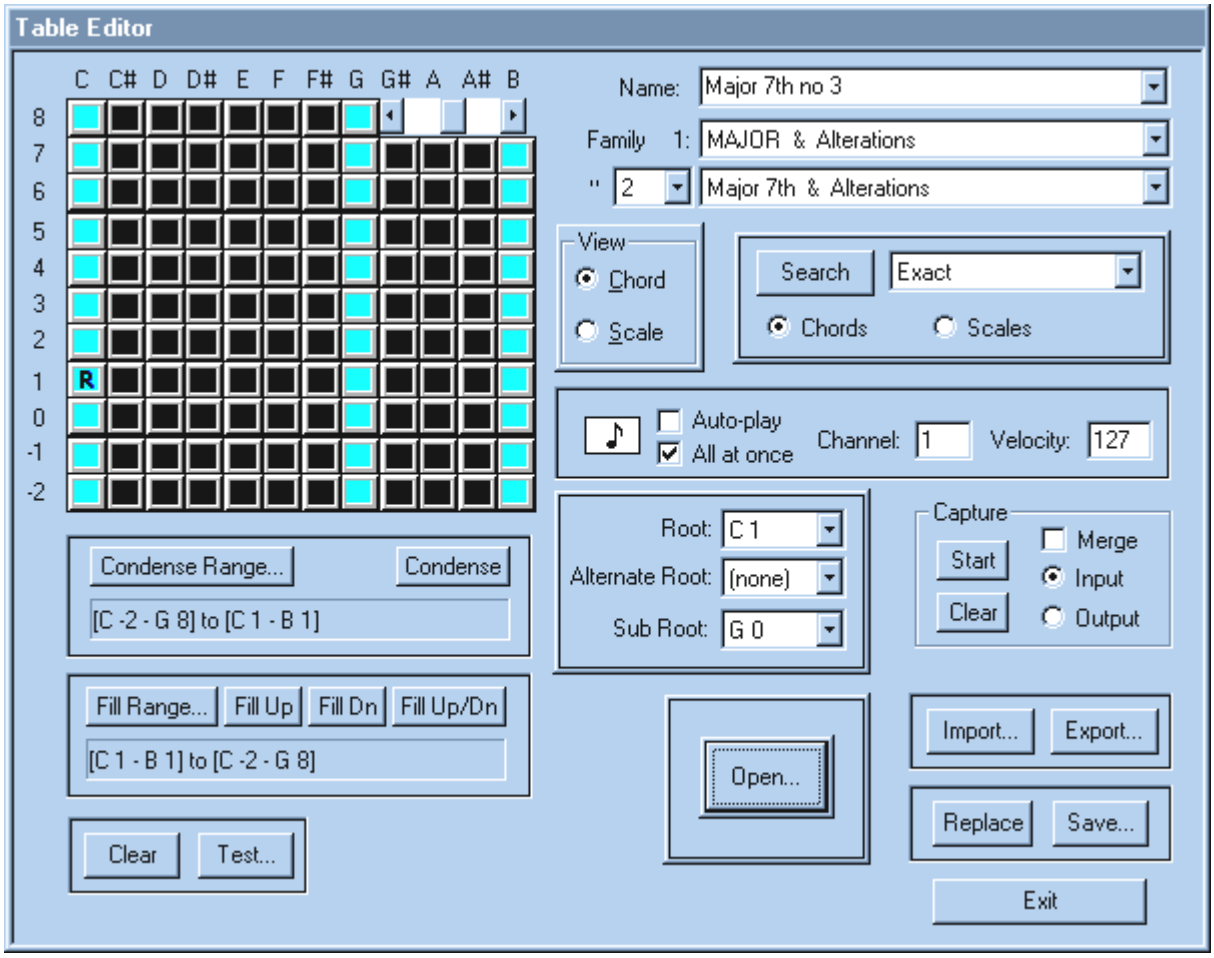
As a piece of music progresses and the chords and scales change, the translator triggers and reads the corresponding chord and scale tables. The following figure shows a *scale table* window open inside the Translator's main window. Behind the table are visible the audio mixing and audio transport (sequencer/CD) controls. To the left in the table window is the incoming MIDI notes mapping positions, with the notes (C, C#, D, D#, etc.) running across the top and the octaves (8 to -2) running down the left of the table. Every cell corresponds to a MIDI note value. On the right of the table window are the various table editing controls (see Advanced Features section).



At any instant in time, only the highlighted (lighter) cells are active. These notes are grouped together by the *Scale processor* from the zones that have been channeled to it. For example, in the Scale Table

above, notes of the major scale are mapped across all ten possible octaves. When the scale changes, the software reads in the next designated scale table and the same input notes then play the notes of the new scale. *Note: all structure Tables are programmed with C as the reference key (changes of key are controlled separately).*

An identical procedure applies to the *chord processor*. A Major 7th (no 3rd) *chord table*, derived from the same major scale, is shown below. Again, at any instant in time only the highlighted (lighter) cells are active, and these notes are condensed together by the chord processor in the zones that have been channeled to it.



The parallel *Melody processor* operates with tables in the same way. It can use the same table as the scale processor, or operate independently on another table. The melody processor allows customization of tables so that the player can play additional melodies more easily and with assured emphasis. For example, octaves may be played on top of another channel, and the required notes to do this are all condensed onto a dedicated zone on the keyboard.

The *thru processor* operates as the name implies. Any MIDI note values channeled to it via the input key map and the channel setting are delivered to the processor outputs directly.

Chapter 7: Hotz Translation Processing Overview

Grid Processing

Closely associated with tables are **Grids**. These can be applied to the entire dynamic sequence of scale, melody and chord tables and they act as an additional manipulator in the translation process. They can be viewed as an indexer into the **Chord/Scale Tables**. They greatly amplify the already formidable power of the **Chord/Scale Library** by allowing endless rearrangement, mode shifting, and hyper-transposition of all existing tables in a consistent manner. Grids can be dynamically selected from a list of 128 preprogrammed and user-configurable grids. Additional banks may be created if needed. Grids are selected/triggered during music sequences, or in live performance, in the same way that chord, scale, and melody tables are (as the following section discusses).

Triggering Table and Grid changes: the Conductor Processor

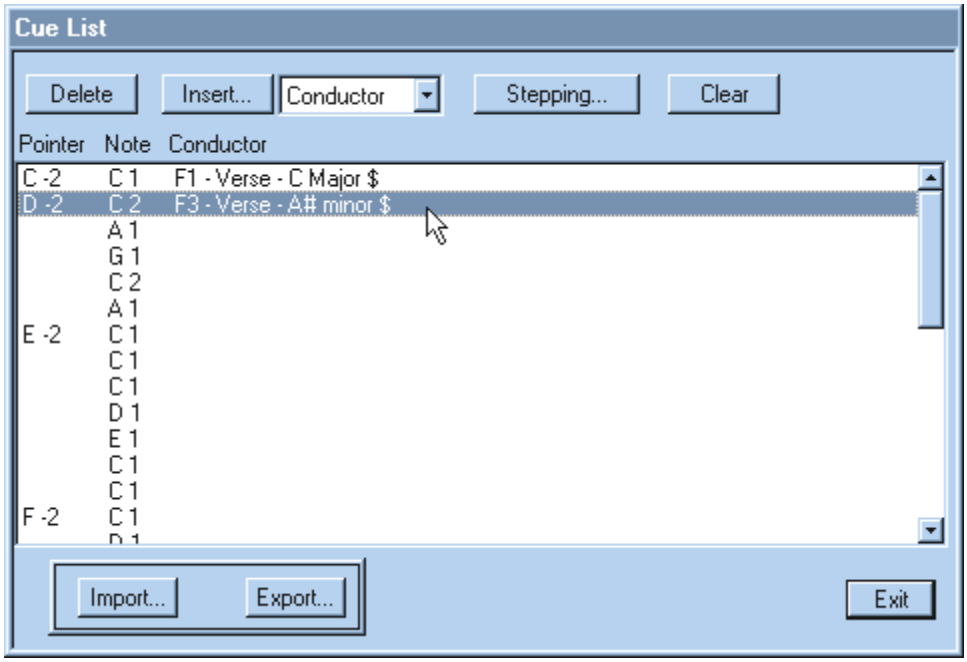
A crucial capability of the Translator software is its ability to instantly access **Chord/Scale Tables** and use these to translate the incoming MIDI data. The table changes are triggered in a variety of ways. Some are internal to the software and some are external, interactive methods. Techniques are:

Conductor keys - Interactive, external control
Sequencer - Internally generated 'Hotz' codes

Each trigger, table reading, and translation operation combined happens in about a millisecond and is not discernible by the player or listener. User-defined timing offsets can be programmed into the triggering process to compensate for the performer's anticipation of changes. For example: A slight 150 ms. pre-delay that ensures the correct chord and scale are already mapped and available when the music's chord/scale change occurs can allow a performer to push the beat and still have the correct structure available.

The *Conductor processor* has a unique and powerful capability. Input keys that are mapped and channeled to the conductor processor are then used to trigger table changes and are called Conductor keys. Up to 128 conductor tables, each with 128 functions, may be programmed in a single Cue List bank. Pressing a Conductor key instantly re-configures the underlying chord and scale tables, and so translates the input controller accordingly. Only keys that are sustaining their notes through the changes are unaffected by Conductor changes. These notes are translated after the key(s) are released (see section 3.2). Conductor keys can optionally be configured to sound the chord as it changes and so become an audible reference or performance function. In Translator groups, one performer can 'conduct' the others by pressing conductor keys that will translate multiple keyboards simultaneously.

Chapter 7: Hotz Translation Processing Overview

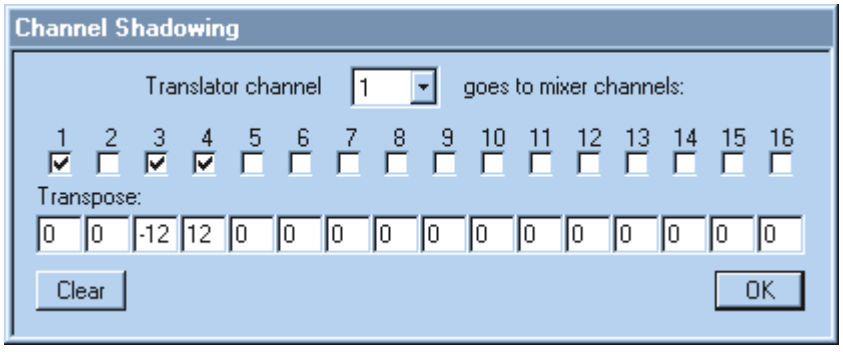


In another table triggering process, the integrated *sequencer* (described below) creates and saves *Hotz codes* to a file. Hotz codes provide a very concise control format for sequencing the translation tables used during a music passage. On sequencer replay, these codes are used to trigger any desired table changes. The user can record backing tracks, which include the generated Hotz codes, and accompany them using the dynamically translated controller(s).

Another powerful use of Hotz codes is writing codes to accompany pre-recorded music CDs. Files of Hotz codes can be written by any user familiar with the process. Alternatively, Hotz Technologies will work with music publishers to include Hotz codes with prerecorded music. When playing the Hotz Translator and music CD on a multimedia PC, the sequenced Hotz codes will trigger translation table changes in time with the music on the CD. The performer can then accompany the music, or improvise over it, using a keyboard that is automatically translated to the correct sequence of music structures.

MIDI Channel Shadowing and MIDI Out

The software offers another processor at the output of the translation engine**. MIDI Channel Shadowing** allows all MIDI data that is output on each of the 16 independent MIDI channels to be simultaneously Transposed and output on any combination of channels.



This process occurs directly before sequencer input and delivers the same data to the MIDI out port as the sequencer receives. Internal MIDI channels passing from the translation processors can be easily mapped by the user so that all desired channels receive the MIDI data. Again, the software provides a very clear and intuitive graphical editor for this process.

Chapter 8: Advanced

HotzCode Quality Assurance Procedure

Basics:

The #1 priority is to make sure that the musical structure signatures of the chord and scale changes (HotzCoding) match the musical structure signatures of the song.

It is mandatory to check the HotzCode (chord and scale changes) with an input device that is capable of full and simultaneous polyphony such as a MIDI Keyboard, Hotz Controller, Maxim keyboard, etc. A standard computer keyboard does not work for this purpose due to the limited number of voices that can be played on the standard PC keyboard simultaneously.

When checking Chord data it is mandatory to hear the entire structure including any alternate or sub roots. At least 7 consecutive events beginning at the root and progressing upward as well as any alternate or sub roots must be continually and simultaneous played in order to properly check the musical structure signature as the song progresses.

This is most easily accomplished by using the standard “AllChords” scene which should use the “AllChord” “Input Keymaps”. The “AllChord” “Input Keymaps” will transform C1, D1, E1, F1, G1, A1, B1, C2, and D2 coming in on channel #1 to the correct input notes for this procedure. This means that with a default grid the actual root (or alternate root) will appear at the E1 key with the structure progressing upward on the keys F1, G1, A1, B1, C2, and D2. The C1 key will generate the root(or alternate root) an octave down and the D1 will generate the sub-root. Playing the C1, D1, E1, F1, G1, A1, B1, C2, and D2 keys simultaneously will produce the overall signature of the structure for most circumstances.

Chord data should be checked from beginning to end a minimum of 2 times. First, play along using the Rhythm Engine set to a density of at least 16^{th} notes. It may be necessary to adjust the HotzCode Offset for some situations. Next, do free form play along with a HotzCode Offset that is appropriate for standard play on your system.

When checking Scale data it is mandatory that the song’s melodies and harmonies be reproducible with the HotzCoded scales. One must be able to continuously perform arpeggios up and down all of the notes of the HotzCoded scales without hearing notes that are harmonically inappropriate for the song.

Scale data should be checked from beginning to end a minimum of 2 times. First, verify that the melodies and harmonies are reproducible with the Scale changes and that the root of the scale does not inappropriately shift up or down. Next, verify that arpeggios do not produce notes that are harmonically inappropriate for the song.

Chapter 8: Advanced

Procedure for testing Chord Changes

1. Open the "**all chords**" scene (create it if it does not exist).

2. Under the options menu select "HotzCode Offset" and temporarily set it to a value of (-20 milliseconds).

3. Start the song and engage the Rhythm Engine by pressing the numeric keypads 5 key for 16th notes or the 6 key for 32nd notes.

4. Simultaneously play the C1, D1, E1, F1, G1, A1, B1, C2, and D2 keys, adjust the volume for optimum structure discernment and verify that the musical structure signatures of the "Chord Changes (HotzCoding)" matches the musical structure signatures of the song. Repeat if necessary.

5. Under the options menu select "HotzCode Offset" and set it back to it's original value of (-150 milliseconds (the default) or other previously optimized value).

6. Start the song, optimize the Volume for freeform play along, and verify that the musical structure of the "Chord Changes (HotzCoding)" provides a satisfactory play along experience throughout the song. Repeat if necessary.

Procedure for testing Scale Changes

1. Open a "**solo**" scene (create one if it does not exist).

2. Start the song and optimize the Volume for melodic discernment.

3. Verify that the melody and harmonies may be played throughout the song and that only notes that match the musical structure of the song are available. Repeat if necessary. Scales should not jump to different roots throughout the song unless true transpositions occur. Very Basic Example: If a song in C Major (C, D, E, F, G, A, B) has a passage where C, D, E, F, G, A, Bb is used, the scale should become C Mixolydian (Major Mode 5) not F Major unless the entire song seemed to do a literal transpose up 5 half-steps.

4. Start the song, continuously perform arpeggios up and down all of the notes of the HotzCoded scales and verify that arpeggios do not produce notes that are harmonically inappropriate for the song. Repeat if necessary.

Chapter 8: Advanced

There are several ways HotzCoders can interpret certain sections in a song and one way is not necessarily better than another. However you should verify that none of the problems listed below are present in a properly Coded Hotz file.

Critical Items

- The most obvious - BAD NOTES!
- Chords played in Octaves out of the proper range.
- Structure changes that do not take into account the root note of the bass.
- Scales that have root changes when only a scale change is necessary.
- Sub-roots that are out of key because of improper coding use of an Alternate- root and/or Grid.

Additional Procedure for testing CD coded material

(1) The delay should be set so that the CD is lined up with the tick. If it is not, temporarily change the delay so that you may test the chord and scale data.

(2) The beat align must be satisfactory. Some songs are clearly easier to beat align then others. Songs written to a click, such as dance music, will be the simplest to beat align and will have a more accurate outcome. Beat aligning a progressive rock song or any live band will be much more challenging because their beats are never consistent, therefore in this case we look for a satisfactory beat align. The “one” beat in every bar must always fall in place.

(3) Check the chord and scale data the same way as show above.

Verify proper HotzCoding of the Chord and Scale changes before proceeding to the Song/Scene checklist.

Any discrepancies discovered in the procedures listed above should either be corrected by the QA person or documented and forwarded to the person responsible for correcting the coding. Once these changes are made, the song should once again be QA’ed for Chord and Scale changes before proceeding to the Song/Scene checklist.

Final Song/Scene Checklist

Chapter 8: Advanced

Make sure that you are using the final version of the song and that it has passed the QA procedure for verifying Chord and Scale changes.

For each Scene open and verify the following for each channel:

- Automation Enable – verify proper state of "Use Automation"
  This state is stored with each scene. Most XG files require that "Use Automation" be enabled. Automation data should be verified on songs and scenes that use it.
- Channel settings - Generally the defaults should be as follows:
  - Highest Note = G8
  - Lowest Note = C-1
  - MIDI Channel = each channel should default to itself 1=1, 2=2, etc.
  - Transpose = 0 This should only be modified in octaves –12, 0, 12, 24
  - Velocity Scaling Tables
    "Soft Touch Compensation" for channels 1-4,
    Channels 5-16 should use tables appropriate for the song,
    Channel 10 is usually set to "default".
- Channel Shadows – appropriate for the scene.
- Input Channel Settings – appropriate for the scene.
- Pan Placement – appropriate for the scene.
- Reverb setting – appropriate for the scene.
- Track Mutes – *****Very Important****** verify that all **HotzCode tracks** are **not muted** in any of the scenes. Other track mutes should be appropriate for the scene.
- Volume – appropriate for the scene.
- Channel 10 should normally be set to Patch #1

For each Song verify the following:

- Locate Position
- CD Offset – verify for ".hcd" files
- CD Track– verify for ".hcd" files
- Clock setting – ".hcd"=CD, ".htz"= Internal
- Locate Position – make sure the song starts at the appropriate position.
- Loop On/Off – in most cases this should be off
- Meter - verify correct Meter and/or use of the Meter track.
- Tempo - verify correct Tempo and/or use of the Tempo track.
- Song Information – review and update the song information including the latest revision #, the date you perform these modifications and your initials.
- Under the options menu select "Record Enable" and make sure that the HotzCode Master Enable is unchecked. If it is checked, uncheck it unless the intended use of the file you are creating is a template for writing or HotzCoding.

Final Steps

- Delete any unnecessary Scenes

- Select the Scene that the song should default to when loaded (this is usually “All Chord Notes”)
- Save the Song with its “Gold” name
- Reload the Song and verify that it seems O.K. – **It is now Gold!**

Chapter 8: Advanced

MIDI & Required Equipment

The **Hotz Translator** requires a MIDI sound card, and preferably, a multi-timbral MIDI keyboard or sound module as the sound source. As with any musical instrument, good sound quality greatly enhances the satisfaction one experiences while making music with the Translator. Users new to MIDI and music software should consult any of the many excellent books and articles that have become widely available (special issues by *MUSICIAN* and *Keyboard* are highly recommended, as is the work of experts such as Craig Anderton and Jeff Rona). Since both MIDI and the Windows operating system can require many pages of explanations, this preliminary manual will not attempt to cover the intricacies of either; a basic working knowledge of both is assumed.

The Hotz Translator can import **Standard MIDI Files**. Most software and hardware sequencers currently available support this format (make sure that disks used to transfer such files are formatted for PC compatibility). The **General Standard** MIDI convention is also supported. A brief description of the **GS** standard follows:

Channels 1-6 and 10 are recommended as main channels for data. There should be only one program change per channel (if a change of tone is needed, this data should be recorded as a new instrument on an additional channel: 7-9 or 11-16).

MIDI Channel	Part Type	Notes reserved (total = 24)*
10	Rhythm (drums)	2
1	Piano	6
2	Bass	2
3	Chord	2
4	Melody	2
5	Sub Chord	2
6	Sub Melody	2
7	(Lower Part)	2
8	(Harmony Part)	2
9	(Melodic Part)	2
11	(Percussive Keyboard)	

* The number of notes per channel can exceed the number shown, but be careful not to exceed the total limit of the sound source being used.

Setting your multi-timbral sound source to these assignments will facilitate MIDI file exchanges with other musicians using completely different systems. Enable **Receive Program Changes** on your module so you can control its instrument sounds from the Translator. Remember that some modules could change from a multi-timbral combination sound to a single channel combination sound when sent program changes on their **Global channel**. To minimize confusion, such modules should have their Global channel set to a channel used only for sending combination sound program changes (*Korg users take note*).

Appendix

Examples

Simple translation examples

Below are several examples of the basic translation process at work. The fingering patterns required on a regular keyboard for several structures are compared with the fingering patterns used for translated versions of the structure. In this example incoming white notes (C1, D1, E1, F1, G1, A1, B1, C 2, etc.) will output the scale notes of the structure; incoming black notes (C#1, D#1, F#1, G#1, A#1, C#2, etc.) will output the chord notes of the structure. Of course, the same principles illustrated here apply to whatever tables are used in translation. The following example uses a G2 Minor structure:

The white notes play the G Minor scale notes:

input - C1, D1, E1, F1, G1, A1, B1, C2, D2...

output - G2, A2, Bb2, C3, D3, Eb3, F3, G3, A3...

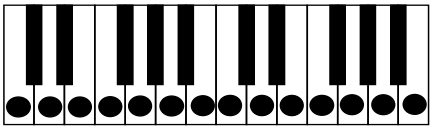
The black notes play the G Minor chord notes:

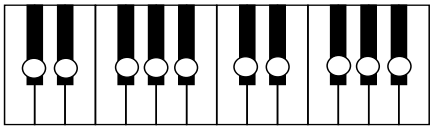
input - C#1, D#1, F#1, G#1, A#1, C#2, D#2...

output - G2, Bb2, D3, G3 , Bb3, D4, G4...

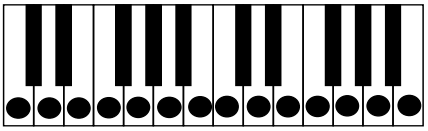
The figures below show how the G Minor translation table applied to a MIDI keyboard zone compares with the notes of a regular keyboard.

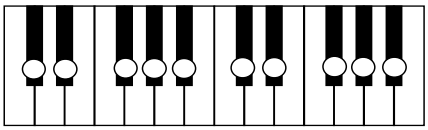
Notes played on a regular keyboard:





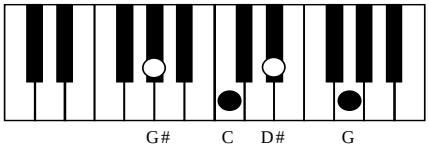
Note output after being processed through G minor translation table:



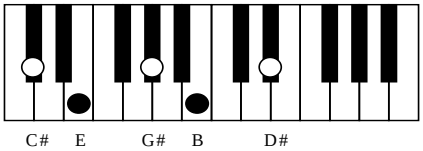


Appendix

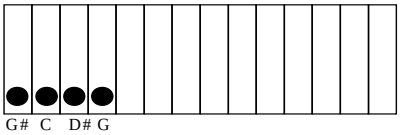
G# major 7th chord on a conventional keyboard



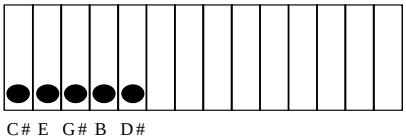
C# minor 9th chord on a conventional keyboard



G# major 7th chord on a translated zone



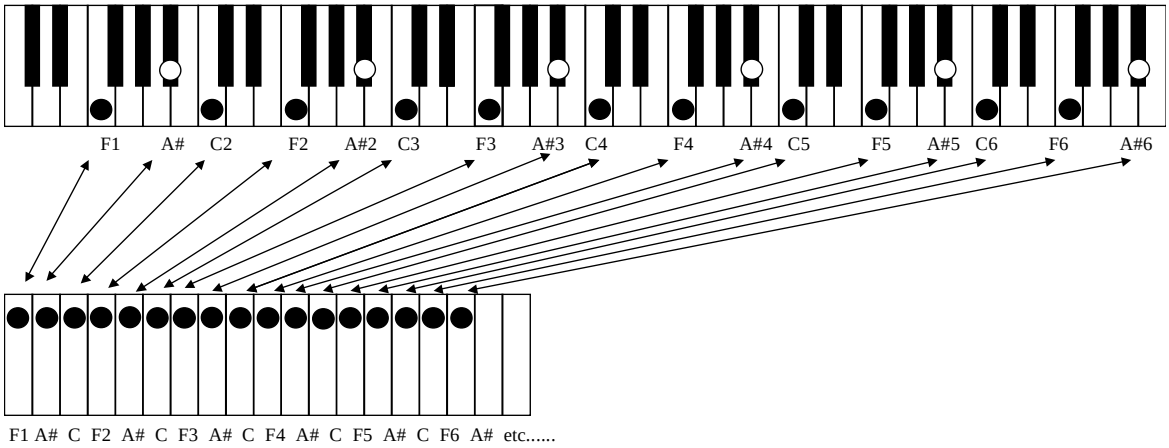
C# minor 9th chord on a translated zone



Notice how much the chord and scale notes of the structure are compressed into a small region of the keyboard. This is because the unused notes are not available, making the keyboard 100% efficient. This not only makes the playing of 'incorrect' notes impossible, but makes much wider voices possible. Also, more keys are available for additional functions (such as Conducting).

The patterns above are shown in only one octave. In reality, structures are mapped out in all octaves across the keyboard, as in the next example.

Full expansion of notes for an F suspended 4th chord on a 6 octave keyboard

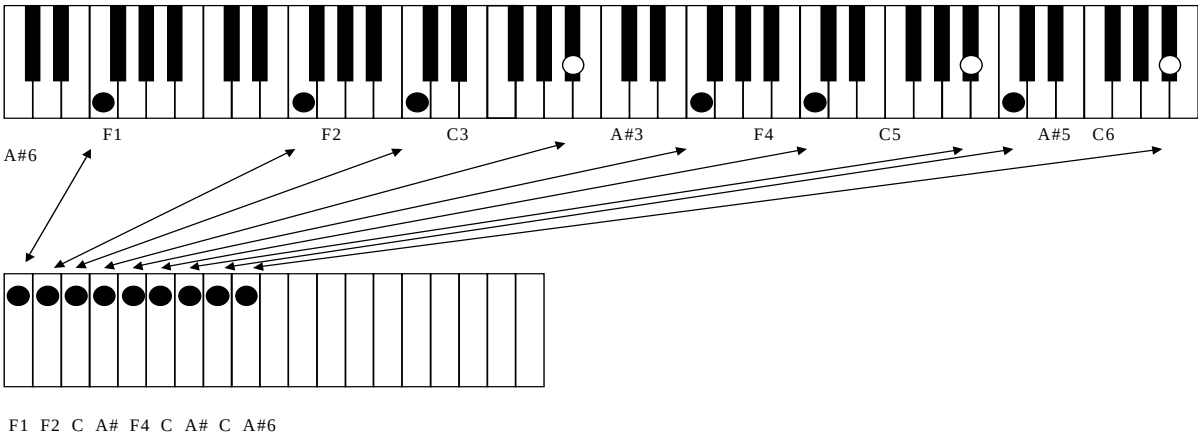


Full expansion of notes for the same 6 octaves of an F suspended 4th chord on a Hotz Translated Instrument. Note the 100% note layout, with all possible voicings available.

Appendix

Super Voicing example

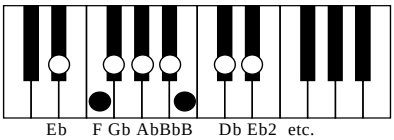
It would not be possible for a single person to strike all of the keys below at the same instant to create this super voicing. However, Hotz Translator technology makes it easy to play, even for novice musicians.



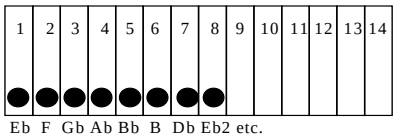
Basic Scale example

The complex fingering required to recall and play any scale is avoided with the Translator. All scales are mapped in similar fashion on the Translated keyboard.

Eb natural minor scale on
a conventional keyboard



Eb natural minor scale on
a translated zone

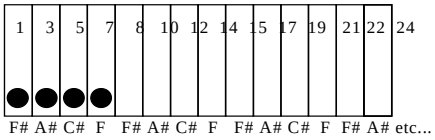


Appendix

Advanced “Chord and associated Scale” Example

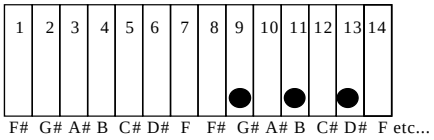
In addition to setting up a particular zone to lay out the chord notes of a song, another zone may be set up to contain the scales the current chords of the song were derived from. This makes it easy to add scale note extensions to the base chord structure.

In the following example, the current chord of a song is F# Major 7th. Zone 2 is assigned to play the notes of the scale which the current chord was derived from. Although the current chord is F# Major 7th, and all of the voicings of that chord are available on Zone 1, you may want to add extensions (such as the 9th, 11th, or 13th). For example, you may easily and confidently add a 9th (G# in this case) to the structure by playing the 9th position on zone 2. Whether the current chord is a major, minor, b9, #9 or whatever, the 9th key of that zone will provide the 9th function (b9, 9, #9), depending on the type of chord in the Table. It is not necessary to think of a different positioning for different keys. Another advantage of such a layout is the ease with which you can add scalar runs that are exact complements to the current chord. This provides tremendous power to players of all musical skill levels.



Zone 1

F# Major 7th Chord Notes



Zone 2

F# Major 7th Chord-scale extensions 9,11,13

Appendix

Basic Troubleshooting

NO SOUND?

After installation, if you do not hear sound when you press a key on your computer keyboard:

1. Check that your audio system is ON (speakers, amplifier, etc.).
2. Make sure the volume of your sound card is turned up (refer to your sound card documentation for more information); under Windows 95/98 you may use the Windows Volume Control accessory to adjust the level (see your Windows documentation for more information).

If you still cannot hear any sound, you may need to change the selected MIDI driver. To do this:

1. Double-click on the Options button and select MIDI Output Device.
2. When the dialog appears Double-click on MIDI Output to select a MIDI device.
3. You should select YOUR sound card or external MIDI device from this list.

If you still do not hear any sound, try selecting a different MIDI output device.

CLEARING A STUCK NOTE

Occasionally, your sound card will get stuck playing a sound. To clear the stuck note simply click on the All Notes Off button, the blue note symbol(it highlights red when your mouse is over it) with a slash through it, near the Play button. This control works whether Play is active or not.

If this problem occurs frequently, close all other applications which are open or which may be running automatically in the background (such as e-mail, fax software, and so forth).

Hotz OnLine

Visit the Hotz Store Web site at www.hotzstore.com for the latest information on Hotz products and troubleshooting.

Index

Index

Index