All of us here work hard to provide powerful but affordable software. From the day in 1987 when Cakewalk for DOS 1.0 first shipped, we have never used copy-protection. We thank you for your purchase of this product!

Glossary

activate
Activating a window means to make it the topmost, current window. The window caption changes to show that it is the active window. If an application's main window is active, then one of its "child" windows may be active, too.

event filters

Event filters let you specify criteria that an event must meet in order to be affected by some operation. Each row specifies a kind of event to include or not. If you check including a kind of event, the rest of the row indicates certain values that an event must have to still be included.

velocity
In MIDI, how fast you strike a key is called the velocity. Each note event has a velocity stored with it. Usually the velocity controls how loud and/or "bright" the sound of the note is, just like with a real piano. But this depends on the synthesizer or sound card: velocity can sometimes have no effect (as with an "organ" patch) or set to control other characteristics of the sound.

patch changes
A type of MIDI command that tells an instrument to use a different sound. Numbers from 0 to 127 each select a different sound -- *what* sound depends on the synthesizer: MIDI doesn't require that certain numbers mean certain kinds of sounds. This flexibility has many benefits. But an optional "General MIDI" specification does assign sounds (like "Grand Piano") to particular numbers, making it easier to distribute MIDI Files that will sound similar on different equipment.

> Note that manufacturers use an incredible variety of jargon for "patch", including "program", "voice", "preset", and other terms. We stick with "patch" because it's one of the oldest phrases and one which doesn't have other meanings for electronic music.

synchronization
A mode of playback where Cakewalk follows an external source of timing instead of its internal clock.

From and Thru markers
This pair of times let you tell most Cakewalk editing commands what region of time you want changed.

Now marker
Shows the current time. As you play or record, this will change to reflect the current time. While not playing or recording, as you move through screens you will change the current time. Some editing commands -- like Paste -- also propose using the Now marker for a time.

CAL
Cakewalk Application Language.   CAL is an interpreted language for writing custom editing commands.   CAL programs can get user input, display messages, and insert, delete, and modify events in tracks.

controller
MIDI uses controller messages for a variety of things. Each controller message has a **number** for what it controls. The MIDI spec assigns many controller numbers to standard functions. For example, controller 1 is the Modulation Wheel, controller 7 is Volume, controller 10 is Pan, and controller 64 is sustain pedal. However, not every synthesizer responds to every assigned controller number. And some synthesizers have special functions assigned to other numbers. You need to check the manual for a synthesizer to see what it responds to. Each controller message also has a **value**. For instance, 0 for Modulation Wheel means no modulation, and 127 means full modulation. The Sustain Pedal is an "on/off" controller, so 0 means no pedal and anything else means the pedal is down.

System Exclusive

MIDI data that is understood only by a particular brand/model of synthesizer and ignored by everyone else. Each manufacturer gets a unique ID code with which to tag their System Exclusive messages. Typically used to save/restore data describing patches (sounds) and/or configuration information for a synthesizer. Cakewalk can receive this data, store it, and transmit it, but cannot understand what it means.

layers (Staff view)
When a staff contains other than chordal harmony -- that is, overlapping notes with different durations -- separate layers must be created. Each layer has its own stem direction, beaming, and rests. Cakewalk has no internal limit to the number of layers, but aesthetics will impose a practical limit. If too many notes, too close in pitch are crammed together; Cakewalk cannot resolve these "aesthetic errors". You can help by moving parts to different tracks, so they can have a staff to themselves.

Contents

Welcome to Cakewalk Professional for Windows!

While using Cakewalk, you may get context-sensitive help -- help about what you're currently working on -- at any time.   Press the F1 function key, or, press the Help button of any dialog box.   This help system will pop up and take you directly to the relevant topic.

Point and click on any of these items for help:

Answers to Common Questions (Save yourself a call to Tech Support!)
Keyboard Shortcuts
Glossary

Help on Menus:

File   **Edit**   **View**   **Realtime**   **Mark**   **GoTo**   **Track**   **Settings**   **Window**

Descriptions of differences between Cakewalk for Windows and Cakewalk 4.0 for DOS are in this style.

Keyboard shortcuts

Function Keys

F1	**Help** (context-sensitive)
CTRL+F1	**Run CAL**
ALT+F1	**Undo**
F2	**Copy**
CTRL+F2	**Cut**
SHIFT+F2	**Paste**
CTRL+SHIFT+F2	**Paste To One Track**
SHIFT+F3	**Step record**
CTRL+F4	Close child window
ALT+F4	**Exit**
F5	**(GoTo) Time**
ALT+F5	**Search Next**

(The F6 keys work only with ***View/Options/DOS F6 Keys*** *on:)*

F6	Switch to Track pane of **Track/Measure view**
SHIFT+F6	Switch to Measure pane of **Track/Measure view**
CTRL+F6	Open or switch to **Event List view**
ALT+F6	Open or switch to **Piano Roll view**
F7	**(GoTo) From**
F8	**(GoTo) Thru**
F9	**From=Now**
F10	**Thru=Now**

Other Keys

CTRL+O	**Open**
CTRL+S	**Save**
CTRL+Z	**Undo**
CTRL+C	**Copy** (Also copies text in notepads/dialogs)
CTRL+X	**Cut** (Also cuts text in notepads/dialogs)
CTRL+V	**Paste** (Also pastes text in notepads/dialogs)
CTRL+SHIFT+V	**Paste To One Track**
CTRL+TAB	Next child window.
SHIFT+CTRL+TAB	Previous child window.

README.WRI

This file contains important information about changes since the printed documentation went to press.   You should read this file.

Cakewalk SETUP installs an icon for README.WRI that you can double-click on.

Or, choose the "Help/View README.WRI file" command.

Save changes?

You are quitting Cakewalk or loading a new file, but your changes to the current file have not been saved.

- To save your changes before quitting or loading a new file, press **Yes**.
- To simply quit or load another file, press **No**.   Your changes will be lost.
- To back out of quitting or loading a new file, press **Cancel**.

Cannot Undo later

The action you are about to take cannot be reversed later using the **Undo** command.

- If you are unsure about proceeding and you haven't saved your work, press **Cancel**. Use **Save** to save your workfile now, then take this action again.

- To proceed anyway, press **OK**.

WINCAKE.INI

This file is located in the directory where you installed Cakewalk Professional (if you accepted the directory which Cakewalk SETUP proposed, this is C:\WINCAKE). It contains information that Cakewalk needs to store between sessions, for example, your preferences. Many items are stored here from choices you make in Cakewalk's menus and dialog boxes. But some items can be changed only by directly editing this file using the Windows Notepad application, or, for items in the [WINCAKE] section, by using **Initialization Files**.

If you make changes to this file, they will not take effect until the next time you start Cakewalk. Also, when you quit Cakewalk it writes some information about your preferences to this file. So if you need to directly edit WINCAKE.INI, it is best to quit Cakewalk, make your changes, then restart Cakewalk.

WINCAKE.INI is divided into sections. The section beginning with the line

```
[WINCAKE]
```

is where you should add information if no particular section is mentioned. For example, if you are advised to, "add the line AskSysxAutoSend=1 to WINCAKE.INI", you should put it on the line under [WINCAKE], like this:

```
[WINCAKE]
AskSysxAutoSend=1
```

notepad

When you're using a view that has a standard Windows notepad, remember that you can cut, copy, and paste text between Cakewalk and other programs by using the standard shortcut keys:

CTRL+C	Copy
CTRL+X	Cut
CTRL+V	Paste

These keys work even with small, single-line text boxes in any Windows program.

File menu

The **File** menu contains mostly commands for relating to files, plus a few others.

New	Wipe out the whole work and start fresh.
Open	Load a work from a disk file
Save	Save a work to a disk file
Save As	Save a work to a disk file under a new name
Merge	Load a work's events into the Scrap buffer
Extract	Save the Scrap buffer into a work file
Info	View comments and other information.
Print Preview	Print the current view to the screen
Print	Print the current view
Printer Setup	Change the current printer and/or printer settings
Exit	Exit the program

New (File menu)

If your work has changed since it was last saved to disk, you will be given a chance to save it first.   Then this lets you pick from a list of file templates to create a new work.

See Also:
File Menu
"New" in the *User's Guide* for help with templates.

Open (File menu)

The **Open** command lets you load Cakewalk workfiles or Standard MIDI Files.

Loading a file replaces the file currently in memory.   If you have made changes since it was last saved, you will be given a chance to save it before loading the new work.

System Exclusive auto send banks.

If you load a Cakewalk workfile containing **System Exclusive auto send banks**, you will be asked whether you want to send them.   Answer "No" if you are loading a file from someone else that may contain unknown System Exclusive data.   That way you will avoid accidentally changing your synthesizer's data.   If you want to defeat this prompt, add the following line to your **WINCAKE.INI**:

```
AskSysxAutoSend=0
```

Dropping files from the File Manager

You can also load files by dragging them from the Windows 3.1 File Manager and dropping them anywhere on Cakewalk's window.   If Cakewalk is playing when you drop the file, playback will stop, the new file will be loaded, and playback will be turned back on again.

See Also:
File Menu

Save (File menu)

Use **Save** to save a changed file using the same name.

If you started the work using **New**, it has no name.   You're asked for a name by **Save As**.

Cakewalk for Windows uses the same file format as Cakewalk 4.0 for DOS unless you have tracks with more than 32,000 events, System Exclusive banks with more than 64K bytes, or other things not supported by Cakewalk 4.0 for DOS.   In such an event, Cakewalk for Windows stores the new feature in a new "file chunk".   Cakewalk 4.0 for DOS will ignore and discard that   portion of the file when loading.

See Also:
File Menu

Save As (File menu)

Lets you save the work in memory as a Cakewalk workfile or as a Standard MIDI File.

The ".WRK" or ".MID" file extension will be automatically appended to the name, depending on the type of file you've selected.

Note: Standard MIDI File format cannot hold all the information that is stored in a workfile. Save a file in this format only when you want to use the file with another program that reads MIDI Files.

Cakewalk for Windows uses the same file format as Cakewalk 4.0 for DOS unless you have tracks with more than 32,000 events, System Exclusive banks with more than 64K bytes, or other things not supported by Cakewalk 4.0 for DOS.   In such an event, Cakewalk for Windows stores the new feature in a new "file chunk".   Cakewalk 4.0 for DOS will ignore and discard that   portion of the file when loading.

See Also:
File Menu
"MIDI Files" appendix in the *User's Guide* for more information about MIDI Files.

Merge (File menu)

File Merge lets you load a Cakewalk workfile into the Scrap buffer.   From there, you can **Paste** the material to the location you want.

See Also:
File Menu

Extract (File menu)

Extract saves the current Scrap buffer into a Cakewalk workfile.

Hint: The **Copy** and **Cut** commands move a group of events into the Scrap buffer.

See Also:
File Menu

Info (File menu)

Info lets you set and view special text to describe your work.

Title: The title for your work. This text appears in the top of the **Staff view** and **Event List view** print-outs.

Subtitle: Also referred to as the "dedication". This text appears in the Staff view centered below the Title on the first page.

Instructions: These are performance instructions. This text appears in the Staff view print-out left-aligned near the top of the first page.

Author: If you created the work, you probably want to put your name here. If this field is blank in a file, Cakewalk will use whatever name you provided SETUP. This text appears in the Staff view print-out, right-aligned near the top of the first page.

Copyright: A copyright notice. A default Copyright is proposed based on the name you provided SETUP (like the Author field) and using the current year (e.g. 1993): "Copyright 1993 by <username>"). This text appears in the Staff view print-out below the Author.

Key Words: Supply any phrases you wish to describe the work or its style. For example, "RAGTIME JOPLIN". This will assist searching for files based on the Key Words, using a future feature of Cakewalk or other programs. This text does not appear when you print, except for printing this file information using the **Info** dialog **Print** button.

Comments: Free-form comments you can write using a notepad. This text does not appear when you print, except for printing this file information using the **Info** dialog **Print** button.

Pop Up on Open: If this is checked, this **Info** dialog will be displayed when a user loads the file.

Statistics

This button provides some additional informaton, which is calculated each time you save the file. A file that has not yet been saved (or not saved with Cakewalk 2.0) won't have this information. If you use **Save As** to create a new file -- even if you're writing over an existing file -- this information will be reset.

Editing time: The total time that you've had the file open in Cakewalk -- from the time it was created to the last time it was saved. This does not include time spent editing the file since you last saved it; if you want to update this value, save the file.

Revision: Each time you save a file that's been changed, Cakewalk increases this number. If you load a file, do *not* change it, and save it, Cakewalk does *not* increase the revision number. You can tell whether a file has changed by looking for an asterisk "*" after its name in the Cakewalk window caption.

Creation time and date.

See Also:
File Menu

Print Preview (File menu)

Some Cakewalk views -- like the **Staff view**, and **Event List view** -- can be printed. First, activate the window. Then choose this command.

See Also:
File Menu

Print (File menu)

Some Cakewalk views -- like the **Staff view**, and **Event List view** -- can be printed.   First, activate the window.   Then choose this command.

See Also:
File Menu

Printer Setup (File menu)

Choose the printer and printer options you want using this command.

See Also:
File Menu

Exit (File menu)

If your work has changed since you last saved it, you will be asked if you want to save it now.   If you do not save your work, it will be lost.

See Also:
File Menu

Edit menu

The Edit menu contains commands relating to moving and changing regions of your work. Regions are events (a) from selected tracks and in the time range specified by the From and Thru markers

Undo	Undo the most recent edit or Undo.
Copy	Copy a region to the Scrap buffer.
Cut	Move a region to the Scrap buffer.
Paste	Move scrap buffer back to the work.
Paste To One Track	Move Scrap buffer back to only one track.
Quantize	Round off the starting times of events.
Length	Change temporal length of region.
Slide	Slide a region ahead or back in time.
Interpolate	Scale values; "search-and-replace".
Retrograde	Reverse the order of events in region.
Transpose	Transpose pitches of Note and/or KeyAft events.
Controller Fill	Insert a series of controller events.
Velocity Scale	Change note velocities over time to a set value or percentage.
Fit to Time	Adjust tempos to make region last specified amount of time.
Fit Improvisation	Insert tempos to fit free-form playing to time grid.
Run CAL program	Run an existing CAL program.

Undo (Edit menu)

Reverses the effect of the most recent command from the Edit menu.   If the last Edit menu choice was **Undo**, then this command will be called "Redo", plus the name of the command that was undone.

See Also:
Edit Menu

Cakewalk Windows Clipboard Formats

The **Cut**, **Copy**, and **Paste** commands work with music data on the Windows Clipboard in several formats.

Standard MIDI Files

One format is MIDI File format: the data on the Clipboard is a memory image of a Standard MIDI File. Any other application capable of understanding MIDI Files and recognizing this Clipboard format can exchange data with Cakewalk. You can copy and paste data to and from Cakewalk and the other application. This is faster than creating an actual Standard MIDI File on disk, and there is no temporary file for you to delete later if it was a temporary transfer.

> Note to programmers: Cakewalk calls **RegisterClipboardFormat()** with the string "Standard MIDI File". The handle that you put on or get from the Clipboard is a block of memory from **GlobalAlloc()** that contains nothing more or less than the exact image of a Format 0 or 1 Standard MIDI File. Please help our mutual customers by supporting this simple protocol for data exchange under Microsoft *Windows*.

Private format

A private format is recognized by Twelve Tone Systems applications. For example, if you are running two copies of Cakewalk for Windows at once, and copy-and-paste music data between them, the recipient will use the private format simply because it is faster.

Copy (Edit menu)

This command copies any combination of track events, tempo changes, and meter changes in the time region From..Thru to the Clipboard.

If you check Track events, then events from selected tracks are copied.   You can opt to use an Event Filter to further define what events will be copied.

You cannot cut meter changes unless the From and Thru markers are set to cover a time range that is an exact number of measures long.   Since a meter defines how long a measure is, by definition you cannot cut other than in units of whole measures.

See Also:
Cut
Paste
Paste To One Track
Edit Menu
Cakewalk Windows Clipboard formats

Cut (Edit menu)

Cut moves any combination of track events, tempo changes, and meter changes in the time region From..Thru to the Clipboard.

If you check Events, then events from selected tracks are cut. You can opt to use an Event Filter to further define what events will be cut. But note that the **Delete hole** and **Use Event Filter** options are mutually exclusive: unless you delete *all* the events in the region, there won't be an empty region to delete.

You cannot **Cut** meter changes unless the From and Thru markers are set to cover a time range that is an exact number of measures long. Since a meter defines how long a measure is, by definition you cannot cut other than in units of whole measures.

See Also:
Copy
Paste
Paste To One Track
Edit Menu
Cakewalk Windows Clipboard formats

Paste (Edit menu)

Paste inserts a copy of the Clipboard data in your work at the current position.   Events are pasted to the same tracks they were originally from which they were originally **Cut** or **Copied**.   The pasted material can be repeated any number of times, with the repetitions inserted one after another.   This lets you create repeating themes quickly.

See Also:
Paste To One Track
Edit Menu
Cakewalk Windows Clipboard formats

Paste To One Track (Edit menu)

Works just like **Paste**, but *forces all the events to be pasted into a single track.*

Use this to **Cut** or **Copy** something from one track, then paste it to an entirely different track.

With respect to tempo and meter/key signature changes, **Paste To One Track** works no differently than regular **Paste**.   The tempo map and the meter/key signature map are still treated as separate "tracks".

Hint: To paste multiple tracks to multiple different tracks, try using the Drag and Drop feature of the **Measure pane** in the Track/Measure view.

See Also:
Edit Menu
Cakewalk Windows Clipboard formats

Quantize (Edit menu)

Rounds off the starting times and/or durations of note events to a resolution that you pick.

Durations are given a value of the resolution amount minus one.   That way, notes do not run into each other, legato-style, causing problems on some synthesizers. Durations smaller than the resolution amount will be rounded *up* to the resolution minus one -- not down to zero.

Normally operates only on note events in the selected region (selected tracks in the time range From..Thru).   If you use an Event Filter, however, it will operate on whatever kinds of events you have specified there.

See Also:
Edit Menu

Interpolate (Edit menu)

Interpolate is a powerful command that lets you search for certain events and replace their parameters with new, interpolated values.

You will be given two Event Filter dialogs to fill in.   The first box works like a normal Event Filter, except how you specify the ranges is important. The second Event Filter is used only for its range fields.   The replacement scales parameter values from the first range to the second range.

See Also:
Edit Menu

Length (Edit menu)

Lets you stretch or shrink the starting times and/or the durations of note events by a percentage that you pick.   You specify this in a percentage of the original values.   For example, 200% would make things twice as long, 50% would make them half as long.

This command does *not* change the tempo map.   Instead, it modifies the actual starting times and durations of the events.

See Also:
Edit Menu

Retrograde (Edit menu)

Reverses the order of events.   The first shall be last and the last shall be first.

See Also:
Edit Menu

Transpose (Edit menu)

Transposes the pitches of Note and/or KeyAft events up or down by the amount you specify.

Diatonic math option...	The **Amount** to transpose by is...
Checked	The number of diatonic steps (based on the current key signature)
Not checked	The number of half-steps (chromatic)

Unless you check the **Use event filter** option to exclude one of the two kinds, both Note and KeyAft events' pitches are affected by **Tranpose**.

See Also:
Edit Menu

Slide (Edit menu)

Moves a region earlier or later. This is similar to the **Time+** track parameter, but this permanently changes the event times.

Note: You cannot slide a region earlier than 1:1:0. The From time must be at least as big as the amount of time you are sliding. Otherwise you will simply get an error message.

> *Example:* If From is 1:1:30, you cannot give an amount less than -30.

See Also:
Edit Menu

Controller Fill (Edit menu)

Lets you insert a series of MIDI controller or pitch wheel events. The data values can range from one value to another over time.

In the dialog box, you pick a controller type, a MIDI channel, a starting and ending value for the controller, and a starting and ending time (the From and Thru markers are proposed as a default). Cakewalk inserts a series of controller events so that the value changes smoothly over time from the starting to the ending value. **Controller Fill** never inserts more than one event on the same tick. If there are any existing controllers of that type in the time region, they are deleted before the new ones are inserted.

You can also create controller changes by drawing in the **Controllers view**. Drawing is more flexible: you can draw any shape, whereas **Controller Fill** creates only linear (straight line) changes. However, **Controller Fill** can create an even change over a very long period of time, which might be awkward for you to draw by hand.

See Also:
Edit Menu

Velocity Scale (Edit menu)

Lets you modify note event velocities over time. The **Begin** and **End** values are either velocity values or percentages by which to change existing velocities, depending on whether the **Percentages** option is checked.

If the **Percentages** option is *not* checked, then the **Begin** and **End** values are the starting and ending velocities. A note at the beginning of the time range will have the **Begin** velocity and a note at the end will have the **End** velocity. Notes in between will be changed to values interpolated between **Begin** and **End** over the time range.

If the **Percentages** option *is* checked, then the **Begin** and **End** values are percentages by which to multiply the notes' present velocities. A note at the beginning of the time range will have its velocity multiplied by the **Begin** percentage and a note at the end will be multiplied by the **End** percentage. Notes in between will be multiplied by percentages interpolated between **Begin** and **End** over the time range.

You can also edit note velocities by drawing in the velocity pane of the **Piano Roll view**, which lets you draw shapes other than straight line changes.

See Also:
Edit Menu

Fit to Time (Edit menu)

This lets you make a region last a specific amount of real-world time. **Fit to Time** changes the duration of the region by changing the Tempo map to speed up or slow down, without changing the musical (measure:beat:tick) times of the events in the region.

See Also:
Tempo view
Length
Edit Menu

Fit Improvisation (Edit menu)

Lets you fit a free-form improvisation into the Measure:Beat:Tick timing grid, without altering how it sounds, by both changing the event times and inserting offsetting tempo changes.

See Also:
Edit Menu
"Fit Improvisation" in the *User's Guide* for a full description of the steps needed to prepare for this command.

Run CAL Program (Edit menu)

Lets you pick and execute a CAL program file.   CAL programs work on events the same way as built-in Cakewalk editing commands: they change events from selected tracks in the From..Thru time region.

See Also:
Creating and modifying CAL programs
Edit Menu

View menu

Cakewalk Professional for Windows has a variety of views.

Track/Measure	View of track parameters and measures.
Tempo	Graphical view of tempo changes.
Meter/Key	List of meter (time signature) changes.
System Exclusive	System Exclusive data generic librarian.
CAL	CAL program editor.
Markers	Markers for times.
Faders	Graphical faders.
Play List	Play a list of songs automatically.
Big Time	Display a window with current time.

Views to edit events in tracks:

Piano Roll	Graphical view of a track's notes.
Event List	List of a track's events (all kinds, not just notes).
Controllers	Graphical view of a track's controller data.
Staff	Notes from multiple tracks transcribed on a staff.

View options:

Auto Activate	Graphical view of a track's notes.
DOS F6 View Keys	Use Cakewalk for DOS F6 shortcut keys
Remember Layout	Save your window positions and sizes.
Colors	

Track/Measure view

The Track/Measure view is your main window in Cakewalk.   It is divided into two halves.

On the left is the **Track pane**, which lets you view and change track paramaters.

On the right is the **Measures pane**, which   is useful for getting an overview of your work and performing "global" editing operations.

When you move the cursor over the border between the two panes, it changes to a left/right arrow.   Click and drag the border left or right.

If you double-click on the border, you can toggle the parameters section size between its maximum and whatever other size you had last set by dragging.   (This is similar to maximizing *vs*. restoring a window.)

See Also:
View Menu

Track pane

The Track pane lets you view and change track parameters.

To Select Tracks

You may select tracks by clicking on the track number in the leftmost column. A track is selected when its number is displayed in reverse video. Double-clicking selects or un-selects all tracks.

To Enter New Values

With the mouse Double-click on a field

Wiht the keyboard Use the arrow keys to move the highlight to the field, then press ENTER.

This will let you type in new information directly on the screen, or, it may call up the **Track parameters dialog**.

To Increase/Decrease Values

With the mouse Click on a field and drag the mouse toward you or away from you.

With the keyboard Press **+** and - to add/subtract by 1.
Press **]** and **[** to add/subtract by 10.

To Customize the Layout

You can arrange the position and width of the columns using the mouse. Click in the header row to drag a column to a new position. If you click near the right edge of a column, you can change its width.

To Edit a Track

Click on a track with the right button, and a popup menu will appear listing the different kinds of Cakewalk track editing views. Select the type you want, and the view will open for that track. The menu also contains other common commands.

See Also:
Track parameters
Measures pane
View Menu

Measures pane

The Measures pane lets you see in which measures a track has events.   This is useful for getting an overview of your work and performing "global" editing operations.

To Change the Now Marker and/or the Current Track

Click on a cell to set Now (the current position) and the current track (you'll see the highlight in the Parameters pane move, too).

NOTE: If you click in selected measures -- drawn in reverse video -- you should click and release quickly so you don't start Drag and Drop (see below).

To Open a View Window

Click on a track with the right button, and a popup menu will appear listing the different kinds of Cakewalk track editing views.   Select the type you want, and the view will open for that track.   The menu also contains other common commands.

To Select Measures

Click and drag left and right to set the From and Thru markers.   Selected measures will be drawn in reverse video.   SHIFT+CLICK to extend any existing selection to the point you click.

NOTE: If you click in selected measures -- drawn in reverse video -- you should click and release quickly so you don't start Drag and Drop.   Tapping quickly will clear the existing selection; then you may drag to make a new selection.

To Drag and Drop Selected Measures

You can drag selected measures to a new position.   Click in the selected measures and hold the button: after a moment, the cursor changes.   Drag to the point where you want the new new upper-left corner to be, and release the mouse button.   You can move measures horizontally and/or vertically this way.

To copy instead of moving, hold down the CTRL key while clicking.   The original selected measures will remain.

To clear the selection without starting Drag and Drop, quickly click and release the button.

When you drop measures, the **Drag and Drop Options** determine whether the dropped measure replace, blend with, or slide over the existing measures on which you're dropping them.

Drag and Drop can be undone by **Undo**.

See Also:
Parameters pane
View Menu

Drag and Drop Options dialog

This dialog lets you choose what should happen to existing measures when you Drag and Drop existing material on top of them.   The options are similar to those for the **Paste** command.

By default, whenever you Drag and Drop you are presented with this dialog.   However, if you always want to drop using the same method, and find this dialog box annoying, simply turn off the **Ask This Every Time** option.   From then on, whichever method you've selected will be used, without you being asked.   Should you ever want to change these options, press the right mouse button in the Measure pane and choose **Drag and Drop Options** from the menu.

See Also:
Measures pane

Track parameters dialog

This dialog box lets you change many related **track parameters**. Double-clicking or pressing ENTER on the Port, Channel, or Patch fields brings up this dialog.

Changes here are effected in realtime. This dialog is handy for "orchestrating" while a song plays back, because you can hear the results while you change the "destination" -- port, channel, patch -- and also parameters like key and velocity offset, pan and volume. You can freely experiment with all of them because pressing Cancel restores all of the original track parameters. Press OK to keep your changes.

Press the **Configure** button (next to the **Patch** item) for the **Patch names** command.

Event List view

The Event List view lets you see and edit events from one or more tracks in an alpha-numeric list format. It is not graphical, but it packs a lot of information into a small space and is the only view that lets you see all types of events -- notes, controllers, patch changes, sysx meta-events, MCI commands, wave events, text events, etc.

Multiple tracks

Each Event List includes whatever tracks were selected (including the current track) when you opened the window. To view a different group of tracks, select them, then open a new Event List window.

To Step Play Events

With the keyboard
Press SHIFT+SPACEBAR to play the highlighted event. If it's a Note or Wave event, the sound will sustain as long as you hold down the SPACEBAR. When you release the SPACEBAR, you are automatically advanced to the next event.

With the mouse
SHIFT+click on an event. If it's a Note or Wave event, the sound will sustain as long as you hold down the mouse button.

See Also:
View Menu

Patch change dialog

The list of patch names you may pick from depends on the port and MIDI channel of the event.   If the track has a forced MIDI channel, the channel stored for the event is ignored.

You can change which patch names are used for various ports and channels by pressing the **Configure** button, which calls up the **Patch names** command.

See Also:
Event list
View Menu

Event kind dialog

This lets you change an event from one type to another.   If you are inserting events in the **Event list** view, typically you press the INSERT key to duplicate an existing event, then change its values.

There are various kinds of events.   Cakewalk provides you with standard MIDI events like Notes, Patch changes, and so on.   Plus there are special events that aren't part of MIDI.

MIDI events contain only a small amount of data.   For example, a Note event has a key number (pitch), velocity, and duration.   Cakewalk stores normal MIDI events in such a way that you can cycle among the various kinds without losing data peculiar to a certain kind.   In fact, by temporarily changing a Note event into a KeyAft event, you can get a different view of the same data: you can see the raw MIDI key number instead of Cakewalk's pitch name like "C#4".   You can do this using the plus or minus keys or dragging with the mouse, and avoid calling up this dialog box.

But the special kinds of events require more data -- Wave events, for example, require very large amounts of memory!   Cakewalk cannot preserve this data and must release it when you change to another kind of event.   To make sure that's what you intend, Cakewalk makes you press ENTER and use this dialog box if you want to change an event to or from one of these special event types.

See Also:
View Menu

Wave edit dialog

Wave event data cannot be directly edited in Cakewalk.   But you can load and save data in standard .WAV files.   Also Windows has a standard "Wave" Clipboard format.   You can copy and paste the wave event data between Cakewalk and other programs that understand wave data.

The Sound Recorder applet included with Windows 3.1 can Copy wave data to the Clipboard, but it cannot Paste data.   Either save the data to a .WAV file to load in Sound Recorder, or use a more full-featured wave editing application.

Wave events vs. MCI command events

You can play wave data by using a Wave event, or, an MCI command event with the command "PLAY <wavefile>".   There are pros and cons to each:

	Advantages	Drawbacks
Wave events	Because the Wave event contains the data itself, wave playback is triggered with virtually no delay. Also, you don't have to keep track of one or more separate .WAV files.	Waveform data can grow very large, even for short samples at a low frequency and resolution, making your work and file very large.
MCI command "PLAY *.WAV"	The size of the Cakewalk workfile will be much smaller, because it is storing only the MCI command text, not the wave data itself. Also, MCI commands can be used to control other media devices.	It takes the MCI interpreter time to parse an MCI command and load the .WAV file. Putting the event early to compensate for the delay may not help if the delay is unpredictable.

See Also:
Event list
View Menu

Piano Roll view

The Piano Roll window has two main panes (sections): note and velocity.   When you first open a Piano Roll window, the note pane may be full size and the velocity pane hidden at the bottom.   You can double-click on the the gray border bar to "maximize" the velocity pane or drag the gray border to whatever position you wish.

The Piano Roll view shows you whatever was the current track when you first opened it.   To change which track you're viewing, press ⇅ or the **T** key.

Working in the note pane

There are two tools, selection and edit.

Selection mode lets you drag to set the From and Thru markers.   Press ⬚ or the **S** key.

Draw mode lets you insert brand-new notes, change existing notes, and "scrub" notes. Press ✎ or the **D** key.   The **Snap-to** button lets you pick a round-off amount for when you drag and change note start times and durations.

To change an existing note: Which part of the note you click on -- left, middle, or right -- determines what you can change.

To change...	click on...	
Starting time	left third	A left-right cursor appears, and you can drag the note left or right to a new time without affecting the pitch.
Pitch	middle third	An up-down cursor appears, and you can drag the note up or down to a new pitch without affecting the starting time. In this mode, the note will sound as you move to each new pitch step.
Duration	right third	A left-right cursor appears, and you can drag left or right to change the length of the note without affecting the starting time or pitch.

To insert a new note: Hold the CTRL key and click.   A note appears.   As long as you hold down the mouse button, you may drag the note to the desired position.

If you CTRL+click on...	the new note will be...
an existing note	a copy of that existing note (same pitch, duration, and MIDI channel).
empty space	a copy of the last note that you dragged or deleted.

To delete a note: While dragging a note, press the DELETE key.

To "scrub" notes: If you click anywhere a note does not exist, you can drag side-to-side over time to hear notes.   This can be handy to find the time where a bad note is, to test how an entire chord sounds after changing the pitch of one of its notes, or to test notes after adjusting their velocities in the velocity pane (see below).

To change "hidden parameters": Click on a note using the RIGHT mouse button. A dialog box lets you edit all note parameters, including some that the Piano Roll view doesn't display.

Working in the velocity pane:

The velocity pane graphs the velocities of notes appearing above in the note pane. To modify velocities, click and drag the shape you want. Remember that velocities are not a separate type of event, they are an attribute of note events, just like pitch and duration. So in the velocity pane you are shaping the velocities of existing note events, not creating or destroying "velocity events" (there's no such thing).

To draw a straight line: Hold the SHIFT key while clicking, and you can draw a straight line.

See Also:
View Menu

Controllers view

The Conrollers view is for kinds of events like Control, Pitch Wheel, and Channel Aftertouch. Usually there are groups of these events that describe a shape, for example, a series of Pitch Wheel events with increasing values that create an upward pitch bend.

⇅ The Controllers view shows you whatever was the current track when you first opened it. To change which track you're viewing, press the track button or the **T** key.
There are three tools, or modes, you can work with: selection, draw, and erase.

Selection mode lets you drag to set the From and Thru markers. Press ⬚ or the **S** key.

Draw mode lets you draw new events of the type and MIDI channel you have selected. Any existing events of that type are replaced by what you draw. Press ✎ or the **D** key.
To draw a straight line: Hold the SHIFT key while clicking, and you can draw a straight line.

Erase mode lets you simply erase events you see on the screen. Drag over the events you wish to erase. Press ▨ or the **E** key.

See Also:
View Menu

Staff view

The Staff view is a view of sequencer data as staff notation, with interactive editing capabilities. It's not supposed to be a replacement for full-featured music publishing software that addresses the many possible nuances of conventional music notation. On the other hand, it does an excellent job of converting inexact human performances into notation displaying what you *meant*, rather than the exact timing of what you played.

To insert a new note: Hold the CTRL key and click; while you hold down the mouse button, you may drag the note to the desired position.

If you CTRL+click on...	the new note will be...
An existing note	A copy of that existing note (same duration, velocity, and MIDI channel).
Empty space	A note with a duration matching the size button(s) you've pressed, or, the duration of the last note that you dragged, deleted or changed. (The velocity and MIDI channel will be the same as the last note that you dragged, deleted, or changed.)

Note: When the **3** button is on to insert triplets, three notes are inserted at once for you -- a complete set of three triplets. Click where you want the first of the three to be located. Drag the two to your desired pitches.

To delete a note: While dragging a note, press the DELETE key.

To change "hidden parameters": Click on a note using the RIGHT mouse button. A dialog box lets you edit all note parameters, including some that the Staff view doesn't display.

Snap-to. The "Snap-to" checkbox determines whether you will be guided to inserting notes of certain durations at starting times that are multiples of that duration. For example, if you've selected the quarter-note button, and Snap-to is checked, your inserted notes will be forced to the nearest quarter-note boundary. This can be handy for entering many notes quickly, but of course you will need to turn this off to enter certain things: for instance, if you want to insert a half-note on the last beat of a measure.

"Simplifying" options.

Several options help you display inexact human performances without making the notation too terribly complex. You may not want to see very small noteheads (like 32nd notes) or separate layers. There are alternatives to using the **Quantize** command to round-off the actual times and durations of the notes: you can preserve the human feel without getting horribly complicated notation.

Fill. This option rounds up durations to the next beat (if there would otherwise be a rest). This is especially effective with chords, because the notes may be released at different times. However, you may want to turn this off if you are entering notes using the mouse. Otherwise, you may be confused when you insert an eighth-note in 4/4 time and it looks like a quarter-note, at least until you insert another eighth note immediately following it.

Trim. This cuts off durations if they extend only a little way past the start of the next note. This helps with legato parts. As with Fill, you may wish to turn off Trim if you're entering notes precisely using the mouse.

Resolution. This is similar to **Quantize**, but works on-the-fly without affecting the actual notes' timing for playback. Also, it uses a different approach to rounding-off the timing that is more appropriate for notation.

See Also:
View Menu

Staff printing configuration

Engravers have standardized nine sizes of the five-line staff. The vertical distance between the lines of each staff is called its *rastral* measurement. Each rastral size has a number and is used for a specific genre of musical composition by publishers.

The bigger the rastral number, the smaller the size.

	<u>Trade names</u>	<u>Genre usage</u>
0	**Commercial** or **Public**	Wire-bound manuscript
1	**Giant** or **English**	Elementary band and orchestra books; instruction booklets
2	**Regular, Common,** or **Ordinary**	Sheet music, concertos, classics
3	*Same as rastral 2*	Used for works which contain a greater density of symbols
4	**Peter**	Folios, works for organ, etc.
5	**Large middle**	Band/wind ensemble music; sheet music
6	**Small middle**	Chorals; condensed sheet music
7	**Cadenza**	Pocket editions; cues in piano parts; military marches
8	**Pearl**	Thematic advertisement; ossia clarification

See Also:
<u>Staff view</u>
<u>View Menu</u>

System Exclusive view

This provides 256 banks for MIDI System Exclusive data.

Each bank can hold any number of System Exclusive messages (up to available memory in Windows 386 Enhanced Mode; up to 1 MB in Windows Standard Mode).

A bank may be marked as **Auto**, which means it will be automatically sent whenever you load the file.   Usually you will want to mark your patch and configuration data bank as **Auto** so your instruments are set up automatically for the song.

See Also:
View Menu

CAL view

Lets you create CAL programs using a notepad.

Creating "macros" using CAL

You may also **Record** a CAL program, because CAL has many functions which correspond to Cakewalk commands. For example, press the **Record** button in the **CAL view**. The **Record** button changes to **/Stop/**, to indicate that you're recording a CAL program. Choose the Open command from the File menu, type in the name of a file ("DEMO.WRK"), and press OK. Now return to the **CAL view** and press **/Stop/**. A statement will appear in the notepad, replacing the current selection:

```
(FileOpen "DEMO.WRK")
```

If you now press **Run**, the Open command will be executed just as if you'd chosen it from the menu and typed in the name of the file.

To assign a shortcut key (either a computer keyboard key or MIDI keyboard key) to a Cakewalk command or to a CAL program file, use the **Key Bindings** command.

See Also:
CAL Function Reference
View Menu

Markers view

Markers are a way of associating text with a time.   The time can be a normal musical time, or locked to an absolute real-world time so that if the tempo changes, they will still refer to the same real-world time.   Locked times are useful for indicating cue points when creating music to accompany visuals (film scores, ad jingles, multimedia presentations, etc.)

Anyplace that you can enter a time in a Cakewalk dialog box, you may also select from the list of markers by pressing F5.   The predefined Now, From, and Thru markers are in this list along with any markers that you've defined.   Pick a marker and press OK.   The marker time will be supplied.

See Also:
View Menu

Faders view

The Faders view provides graphical faders (sliders and knobs) that generate MIDI controller events as you move them.

Faders and tracks

Each fader is associated with a particular track, which you can select.   This association has several roles:

1)   The fader will transmit controller events to the **Port** and **Channel** of the track.   (**Tip:** a track can be set to no particular MIDI Channel; if so the fader will not work: it has to transmit to *some* MIDI channel.)   If you record fader movements into a track -- see below -- then you should associate the fader with that track.

2)   If the track contains controller events of the fader's type (for example, Volume), then the faders will update as the current time changes: either during playback or when you manually move to a new time.   The fader shows the value for the most recent controller value in the fader's track.

3) You may create controller events in the fader's track in two ways: by taking a "snapshot" of all the faders or by interactively recording your fader movements (see below).

Moving the fader and generating controller events.

You can drag the fader thumb and slide it to a new position.   Appropriate MIDI Controller events are generated as you change the value.

Faders also have a "preview" feature.   Move the cursor directly over the slider's black slot or inside a knob.   The value will temporarily change to show you what value would take effect *if* you clicked.   This is handy for making instant jumps without generating lots of events with values in between the old and new positions.

Taking a "snapshot" of fader positions

If you have arranged your faders to your liking, you may take a "snapshot" of their positions by pressing 📷.   This inserts a controller event capturing the fader position into each fader's track, at the current time.   (If the current time is the beginning, 1:1:0, then faders set to Volume or Pan will change their track's corresponding Volume or Pan parameter instead.)

Recording fader-generated controller events.

You can record your fader movements into a track.   To do so, check **Record fader movements**.   Start playback.   Now, when you click on a fader, recording starts.   As you drag the fader, each position is recorded into the track as a controller event.   When you release the fader, recording stops.   (Playback continues, so that you may record another fader movement.)

This recording works in a special "punch-in" mode: controllers of that kind will be muted in the fader's track, and replaced by the new controllers you record.   (Other kinds of events in the track are not muted or replaced, so it's safe to mix controller events and notes in the same track.)   You can click on as many faders as you want; each one will be a separate recording segment for each fader's destination track.

See Also:
View Menu

Tempo view

There are three tools, or modes, you can work with: selection, draw, and erase.

Selection mode lets you drag to set the From and Thru markers.   Press ⬚ or the **S** key.   If you do this to prepare for using the **Edit Copy** or **Edit Cut** commands, remember to check Tempo changes in the dialog box, and possibly un-check Track events and Meter changes. These commands can work on any or all of these.

Draw mode lets you draw new events of the type and MIDI channel you have selected.   Any existing events of that type are replaced by what you draw.   Press ✎   or the **D** key.
To draw a straight line: Hold the SHIFT key while clicking, and you can draw a straight line.

Erase mode lets you simply erase events you see on the screen.   Drag over the events you wish to erase.   Press ▨ or the **E** key.

See Also:
View Menu

Meter/Key view

Meter and key signature changes can occur only on measure boundaries.   You can insert, delete, or change any entry in the meter/key map.   The map always has at least one entry, for measure 1, because there must always be some meter and key signature.

Meters

The meter describes how to divide time.   You pick the number of beats per measure, and the value of each beat.

Key signatures

The key signature controls how Cakewalk displays notes.

In the **Event List view** and some dialog boxes, Cakewalk converts the MIDI pitch number to labels like "C#2", or the C-sharp in the 2nd octave.   (Octave 0 corresponds to MIDI pitch number 0, Octave 1 to pitch number 12, and so on.   You can change this through the BaseOctave variable in the Cakewalk **initialization file**).

And of course the **Staff view**   needs to know the key signature.

Changing the key signature only changes how Cakewalk displays notes.   To actually transpose pitches, use the **Transpose** command.

See Also:
View Menu

Play List view

The **Play List** view lets you build a list of up to 128 songs.   While one song is playing the next song is pre-loaded from disk, so that it's ready to start without delay when the first one finishes.   Song files may be either Cakewalk files (*.WRK) or Standard MIDI Files (*.MID, *.MFF).

Be sure to check the **On** item if you want playback to follow the list.   Otherwise, Cakewalk will play the current song -- which appears in the caption of the main Cakewalk window.

Changes to songs are not saved

If you make any changes to a song while the play list is **On**, you will *not* be asked if you want to save the changes.   The play list is designed for live performance and will not interrupt list playback to ask this question.   Also, this enables you to make temporary changes to a song while it's playing back for a particular performance, without worrying about changing the actual song file.

For example, you may save a song with a certain section set to **Auto Shuttle**.   This section may be for an open-ended solo or dancing.   The song will repeat that section indefinitely, until you press the Loop button to turn off Auto Shuttle.   Then the song continues on past the section to finish.   Even though the song is changed, you don't want to save the song with looping turned off, and you don't want Cakewalk even to bother asking you about this. You want to know the original song file is unchanged, ready for your next performance.

Other options

You may set an initial **Delay** for each song.   The delay may be for a number of seconds or be a pause until you press a key to begin the song.

While one song is playing, you may stop it and immediately advance to the next song by pressing the **Next** button.

You may cause Cakewalk to move to a different spot in the list by double-clicking on a particular song.   Whatever song was already loaded and standing by (marked "READY") will be discarded, and the song on which you double-clicked will become the next song instead. If playback is not underway, then double-clicking simply loads that song as the current song. You may use the rest of Cakewalk to edit the song.

In addition to using **Add** and **Delete** to change the list, you may drag and drop songs to different places in the list.   Click and drag to *move* a song to a new position.   Hold the CONTROL key while clicking to drag a *copy* of a song to a new location.

See Also:
View Menu

Big Time view

This displays a window with the current time in a large font for high visibility.

To Change the Time Format

Point anywhere in the window and click the **left** mouse button. The format will change between musical (Measure:Beat:Tick) and SMPTE/MTC (Hours:Minutes:Seconds:Frames) formats.

To Change the Font

Point anywhere in the window and click the **right** mouse button. Enter the name, size, and color of the font. Other font attributes in this standard Choose Font dialog box are ignored.

See Also:
View Menu

Auto Activate (View/Options menu)

Normally you activate windows by clicking on them.   With this option enabled, you don't have to click: merely moving the mouse cursor over any window activates it.

You may want to turn off **Auto Activate** if you don't like this behavior, or if your machine is too slow redrawing the windows as they're activated.   If you are using Windows utility software that already provides auto activation ("floating windows") you will also want to add the line WinEat1stClick=0 to the Cakewalk **initialization file**.   You may want to put it below the line "WinAutoActivate=<0 or 1>", which is how Cakewalk stores your preference for **Auto Activate** being off or on.

See Also:
View Menu

DOS F6 View Keys (View/Options menu)

This option is intended for people familiar with Cakewalk for DOS. It makes the F6 keystrokes act like they do in Cakewalk in DOS. (This emulation isn't perfect because Cakewalk for Windows has more kinds of "views" and many of them can be visible at once.)

Keystroke	Action
F6	If the active window is an Event list or Piano Roll window, it is closed. Activates the parameters pane ("Track View") of the Track/Measure view (the "Track View" and "Measure View" are combined in Cakewalk for Windows).
SHIFT+F6	If the active window is an Event list or Piano Roll window, it is closed. Activates the measures pane ("Measure View") of the Track/Measure view (the "Track View" and "Measure View" are combined in Cakewalk for Windows).
CTRL+F6*	Opens an Event list window ("Event View"), or, if one is already open, activates it.
ALT+F6	Opens a Piano Roll window ("Note View"), or, if one is already open, activates it.

* Normally in Windows programs, CTRL+F6 takes you to the next window in an application. That standard meaning is ignored when this option is enabled.

See Also:
View Menu

Remember Layout (View/Options menu)

While this option is on, whenever you exit Cakewalk it saves the size and position of your windows. They are restored the next time you start Cakewalk. (Windows associated with particular tracks for editing -- the Staff, Piano Roll, Event List, and Controller views, are not saved. These change each time you load a different file.)

When you turn this option off, Cakewalk asks if you want to restore the "factory default" layout. If you answer *Yes*, then the "factory default" window positions will be used the next time you start Cakewalk. If you answer *No*, Cakewalk use whatever layout was last saved.

To set up a layout and protect it from changing:

1. Turn *on* **Remember Layout**.

2. Arrange your windows as desired.

3. Use **Exit** to quit Cakewalk.

4. Turn *off* **Remember Layout**, and answer *No* to the "Restore 'factory default' layout?" question.

See Also:
View Menu

Colors (View/Options menu)

You may choose what colors Cakewalk will use for various kinds of screen elements in many of its views.

The dialog box presents two lists. The **Screen Elements** are categories of things that Cakewalk displays on the screen. Select an element. Then use the list on the right, **Colors**, to select which color you want used for that element.

You may restore all of the screen elements to their factory default settings by pressing the **Default** button.

There are two kinds of colors: *direct* colors like Black, Blue, and Red, and *system* colors, like "Sys Color: Window text" and "Sys Color: Button Face". The system colors correspond to some of the colors you set using Windows Control Panel's "Colors" icon. For example, Cakewalk comes set up for the screen element "Window Text" to use the system color "Sys Color: Window text" . That way, Cakewalk uses the same color as all of your Windows programs.

Explanation of Screen Elements

Except where noted below, the views which follow this color scheme are the Track/Measure, Piano Roll, Controllers, Event List and Tempo views. Dialog boxes and other views use their own, hardwired colors, although these usually follow the colors you've set for those kinds of screen elements as defined by Windows in Control Panel.

Screen element	Explanation
"Window Background"	Background color
"Window Text"	Text color
"Rules"	Rules (lines used to mark values or time)
"Major Rules"	Major lines (lines used to mark larger intervals of values or time)
"Values"	Color of notes and velocities in Piano Roll view, controllers in Controllers view, and tempo changes in Tempo view.
"Drawing"	Color used to show your drawing in Piano Roll, Controllers, and Tempo views.
"Erasing"	Color used to show your erasing in Piano Roll, Controllers, and Tempo views.
"Event List Note"	Color of Note events in Event List view
"Event List KeyAft"	Color of KeyAft events in Event List view
"Event List Control"	Color of Control events in Event List view
"Event List Patch"	Color of Patch events in Event List view
"Event List ChanAft"	Color of ChanAft events in Event List view
"Event List Wheel"	Color of Wheel events in Event List view
"Event List Sysx"	Color of Sysx events in Event List view
"Event List 'Special'"	Color of other, 'Special' events in Event List view

Tip: Don't set any other screen element to the same color as that used by "Window Background", or you'll make the screen element invisible!

See Also:
View Menu

Realtime menu

The Realtime menu contains commands relating to recording and playback.

Play	
Record	
Rewind	
Step Record	Record notes one step at a time
Auto Shuttle	Set rewind, stop, and restart times
Overwrite Mode	Whether recorded material is blended with or overwrites existing material.
Reject Loop Take	Reject the last loop recording take.
Update Patch Cache	Update patches on sound cards which require "caching".
Panic	Turn off all notes, lift all pedals, and center all wheels.

Play (Realtime menu)

▶ You can also start/stop playback by pressing play button in the control bar, or by pressing the SPACEBAR.

The "panic" button on the control bar also stops playback -- plus it turns all notes off and zeroes all continuous controllers.

The loop button on the control bar is a short-cut for selecting **Auto Shuttle**, checking all the options (rewind/stop/start), and setting the rewind and stop times to the From and Thru markers, respectively. In other words, set From and Thru, then click the loop button. As long as this button is on, if you change From and Thru, the Auto Shuttle times will be automatically be updated. *Double*-clicking this button (when it's off) calls up the full **Auto Shuttle** dialog.

See Also:

Realtime Menu

Record (Realtime menu)

⊡ Starts/stops recording. You can also start/stop recording by pressing the record button in the control bar or pressing the R key.

Anytime during playback, you may press the **Record** button to begin recording. You may press it again to stop recording but let playback continue. You may repeat this indefinitely (starting and stopping recording while letting playback continue) to record multiple segments of material.

When you stop recording and playback, if anything was recorded it is automatically kept. You can change your mind using **Undo**.

Layered Loop Recording

If you have full looping set using **Auto Shuttle**, then you may accumulate many record takes, layering them on top of each other.

If you record anything during a loop, you will hear it play back immediately on the next loop. Until you record something new, that is the most recent take, and you may reject it using **Reject Loop Take** without stopping recording.

Changing tracks while recording

With both normal and loop recording, you may switch to a new track in the Track view. Cakewalk will place whatever you record during that segment on that track. For example, during loop recording you may move the cursor to a new track. All loop takes from that point on will be destined for that track. With normal recording, you may move the cursor to a different track for each time you turn on recording.

See Also:

Auto Shuttle
Overwrite Mode
Realtime Menu

Rewind (Realtime menu)

⏪ Rewinds to the beginning or to the optional **Auto Shuttle** rewind time. You can also rewind by pressing the rewind button on the control bar or pressing the W key.

See Also:

Realtime Menu

Step Record (Realtime menu)

Lets you record music one rhythmic step at a time.

Note that you step-record while *inside* the step-record dialog box.   Once you leave that window, you are no longer in step-record mode.

See Also:

Realtime Menu

"Step Record" in the *User's Guide* for details.

Auto Shuttle (Realtime menu)

Auto Shuttle provides various options for automatically stopping, rewinding, and restarting playback under certain conditions.

Rewind To rewinds to a time you specify whenever playback or recording is stopped.

Stop At stops playback or recording at a time you specify.

Restart restarts playback whenever it is stopped -- except when you stop playback manually.   For example, it restarts if playback was stopped automatically by **Stop At**, if the end of the work was reached, or if you pressed the Rewind button.

Full looping

Note that if all three are enabled, this results in looping of all tracks from the rewind time to the stop time.   The loop button on the control bar is a short-cut for setting this.   This looping may have a slight delay associated with it, especially if you have certain **MIDI Out** settings turned on.
If you **Record** with full looping, you may layer multiple takes.

Auto Shuttle and synchronization

If the clock is set to **MIDI Sync or SMPTE/MTC Sync**, checking the **Restart** option is handy for a "chase mode": when playback stops, Cakewalk will restart, which means waiting for sync mode.   You can start and stop playback on the tape deck or drum machine and Cakewalk will follow.   (The **Stop At** and **Rewind To** are ignored when you are using synchronization.

See Also:
Realtime Menu

Overwrite Mode (Realtime menu)

If the **Overwrite Mode** recording option is:
Off then **Record** blends new material that you record with any existing material in the track (recording works in "sound on sound" mode).

On then Record erases the existing material during the time you recorded, whether or not you actually played anything new.

See Also:
Realtime Menu

Reject Loop Take (Realtime menu)

If you **Record** with full looping (using **Auto Shuttle**), then you may accumulate many record takes, layering them on top of each other.

If you record anything during a loop, you will hear it play back immediately on the next loop. Until you record something new, that is the most recent take, and you may reject it using **Reject Loop Take** without stopping recording.

This command is enabled if there is a prior loop take to reject, and you haven't stopped recording and playback.   Once you stop, this command is disabled: then, you may use **Undo** to reject all of the material you recorded.

See Also:
Realtime Menu

Update Patch Cache (Realtime menu)

This command is enabled only if you're using one or more sound cards which require "patch caching".   After making changes to your song which result in patch change events being added, changed, or deleted, you may need to choose **Update Patch Cache** to force the sound card to load the required sounds from disk storage.

See Also:
Realtime Menu

Panic (Realtime menu)

This turns off all notes, lifts the sustain pedal, and centers pitch bend and modulation wheels.

See Also:
Realtime Menu

Mark menu

The Mark menu contains commands relating to setting the values of the From and Thru markers.

From Value	Enter a specific value for the From marker
Thru Value	Enter a specific value for the Thru marker
From = Now	Set the From marker equal to Now (the current time).
Thru = Now	Set the Thru marker equal to Now (the current time).
From = Start	Set the From marker to 1:1:0 (the beginning).
Thru = End	Set the Thru marker to the end of the work.

From Value (Mark menu)

Lets you enter in a specific value for the From marker.

From and Thru markers.

See Also:
Mark Menu

Thru Value (Mark menu)

Lets you enter in a specific value for the Thru marker.

The Thru marker is used by many commands, especially editing commands, to indicate the end time of a region.   The From marker indicates the start of the region.   You mark the region *before* choosing the editing command.

See Also:
Mark Menu

From = Now (Mark menu)

Sets the From marker to be equal to the Now marker.   Now always shows the current time. So, this is useful for setting the From marker, "live," while the work plays back -- especially if you use the F9 shortcut key.

Note: If the current window is the Track/Measure view, then the From time is rounded down to the first tick of the current measure.

From and Thru markers.

See Also:
Mark Menu

Thru = Now (Mark menu)

Sets the Thru marker to be equal to the Now marker.   Now always shows the current time. So, this is useful for setting the Thru marker, "live," while the work plays back -- especially if you use the F10 shortcut key.

Note: If the current window is the Track/Measure view, then the Thru time is rounded up to the last tick of the current measure.

From and Thru markers.

See Also:
Mark Menu

From = Start (Mark menu)

Resets the From marker to the start of the work (1:1:0). Useful when you want the whole work to be included in an edit operation, assuming you also set Thru to the end.

From and Thru markers.

See Also:
Mark Menu

Thru = End (Mark menu)

Resets the Thru marker to the end of the work (the time of the last event from all tracks). Useful when you want the whole work to be included in an edit operation, assuming you also set From to the beginning.

From and Thru markers.

See Also:
Mark Menu

GoTo menu

The **GoTo** menu contains commands relating to going to a different time in the work.

Time	Enter a specific time to go to
From	Go to the time in the From marker
Thru	Go to the time in the Thru marker
Beginning	Go to the start of the work (1:1:0)
End	Go to the end of the work
Previous Measure	Go to start of this or previous measure
Next Measure	Go to end of this or next measure
Search	Go to the next event that meets criteria you supply
Search Next	Find the next event that meets the criteria you specified in the last Search.

Time (GoTo menu)

Lets you enter in a specific time to go to.   The Now marker reflects the change.

See Also:
GoTo Menu

From (GoTo menu)

Takes you to the time indicated by the From marker.   In effect, sets Now equal to From.

From and Thru markers.

See Also:
GoTo Menu

Thru (GoTo menu)

Takes you to the time indicated by the Thru marker.   In effect, sets Now equal to Thru.

From and Thru markers.

See Also:
GoTo Menu

Beginning (GoTo menu)

Takes you to the beginning of the work (1:1:0).

See Also:
GoTo Menu

End (GoTo menu)

Takes you to the end of the work (the time of the last event from all tracks).

See Also:
GoTo Menu

Previous Measure (GoTo menu)

If you are in the middle of a measure, this takes you to the start of that measure.

If you are already at the start of the measure, takes you to the start of the previous measure.

See Also:
GoTo Menu

Next Measure (GoTo menu)

If you are in the middle of a measure, this takes you to the end of that measure.

If you are already at the end of the measure, takes you to the end of the next measure.

See Also:
GoTo Menu

Search (GoTo menu)

Finds an event at or later than the current time which meets certain criteria that you specify in an Event Filter.

See Also:
GoTo Menu

Search Next (GoTo menu)

Finds the next event that matches your previous **GoTo Search** command.

See Also:
GoTo Menu

Track menu

Most of the Track commands correspond to parameters in the **Track pane** of the Track/Measure view. The advantage of these command equivalents is that they let you change a parameter on many tracks at once. These topics explain what the track paramteters are:

Name
Status
Loop
Key+
Vel+
Time+
Port
Channel
Patch
Pan
Volume

The following commands do not correspond to track parameters. They are commands that operate on tracks:

Solo	Mute all tracks except the current track.
Un-solo	Restore the muted settings of tracks.
Clone	Back up a track, putting a clone of it on another track.
Wipe	Delete all events; don't reset track parameters.
Kill	Delete all events and reset all parameters to defaults.
Sort	Sort tracks.

Name (Track menu)

Each track has a name, which you may use for any purpose you wish.

The **Name** command lets you change the name of all the selected tracks at once.

See Also:
Track Menu

Status (Track menu)

Muted tracks are silent, but may be un-muted instantly during playback.

The **Status** command lets you toggle the muted status of all selected tracks at once. "Toggles" means that non-muted tracks become muted, and vice versa.

See Also:
Archive
Track Menu

Archive (Track menu)

Archived tracks are "super-muted": they can't be un-muted during playback.

Why would you want to do this? If you have a large number of muted tracks (for instance, with different versions of solos, or bits of thematic material), you may encounter timing problems. Cakewalk still processes muted tracks during playback so that it can be ready to un-mute them instantly at your command. Archiving them means you "promise" not to un-mute those tracks during playback. Thus they can be ignored, possibly improving performance.

The **Archive** command toggles the archived status of all selected tracks at once. "Toggles" means that non-Archived tracks become Archived, and vice versa.

See Also:
Status (muting tracks)
Track Menu

Loop (Track menu)

Tracks may play from 1 to 9998 times.

9999 has a special meaning: keep looping as long as other, non-looping tracks are still playing -- that is, keep repeating the track along with non-repetitive tracks.

The **Loop** command lets you change the Loop parameter for all selected tracks at once.

See Also:
Track Menu

Key+ (Track menu)

The **Key+** parameter is an amount added to the MIDI key number (pitch) of all note events in the track.   It is an on-the-fly transposition that does *not* change the key number that is actually stored for each note event.   **Transpose** lets you do the latter.

The **Key+** command lets you change the Key+ parameter for all selected tracks at once.

See Also:
Track Menu

Vel+ (Track menu)

The **Vel+** parameter is an amount added to the velocity of all note events in the track. This parameter is an on-the-fly transposition that does *not* change the velocity that is actually stored for each note event.

The **Vel+** command lets you change the Vel+ parameter for all selected tracks at once.

See Also:
Track Menu

Time+ (Track menu)

The **Time+** parameter is an amount added to the starting time of all the events in the track. This parameter is an on-the-fly transposition that does *not* change the time actually stored for each event.   (The **Slide** command lets you do the latter.)

The **Time+** command lets you change the Time+ parameter for all selected tracks at once.

See Also:
Track Menu

Port (Track menu)

The **Port** parameter controls which port tracks will be routed to.   Ports correspond to one or more MIDI Out devices that you've set up using **MIDI Devices**.

The **Port** command lets you change the Port parameter of all selected tracks at once.

See Also:
Track Menu

Channel (Track menu)

The **Channel** parameter forces all events in a track to be transmitted to a particular MIDI channel.   Each event has a MIDI channel stored actually stored with it, so the Channel parameter can also be "--" or "none", letting a track contain events for more than one MIDI channel.

The **Channel** command lets you change the Channel parameter of all selected track(s) at once.

See Also:
Track Menu

Patch (Track menu)

The **Patch** parameter is for transmitting a patch change before playback starts. Tracks may contain actual patch change events, for example to change patches midway through a track. For that, use the **Event List view**. But using the Patch parameter is easier when a track needs only a single patch for its whole length.

If you want to use the Patch parameter, you must also give the track a forced MIDI channel using the **Channel** parameter. Otherwise Cakewalk won't know what MIDI channel to use for the patch change.

The **Patch** command lets you change the Patch parameter of all selected tracks at once.

See Also:
Track Menu

Pan (Track menu)

The **Pan** parameter is for transmitting a MIDI Pan (10) controller event before playback starts.   Tracks may contain actual pan events, for example to change panning midway through a track.   For that, use the **Controllers view**.   But using the Pan parameter is easier when a track needs only a single pan setting for its whole length.

If you want to use the Pan parameter, you must also give the track a forced MIDI channel using the **Channel** parameter.   Otherwise Cakewalk won't know what MIDI channel to use for the pan event.

The **Pan** command lets you change the Pan parameter of all selected tracks at once.

See Also:
Track Menu

Volume (Track menu)

The **Volume** parameter is for transmitting a MIDI Volume (7) controller event before playback starts. Tracks may contain actual volume events, for example to change volume midway through a track. For that, use the **Controllers view**. But using the Volume parameter is easier when a track needs only a single volume setting for its whole length.

If you want to use the Volume parameter, you must also give the track a forced MIDI channel using the **Channel** parameter. Otherwise Cakewalk won't know what MIDI channel to use for the volume event.

The **Volume** command lets you change the Volume parameter of all selected tracks at once.

See Also:
Track Menu

Solo (Track menu)

Mutes all tracks except the current track.

The original track muted settings are restored when you select **Un-solo**.

See Also:
Track Menu

Un-solo (Track menu)

Restores the track muted settings that were in effect before you used **Solo**.

See Also:
Track Menu

Wipe (Track menu)

Deletes all events from all selected tracks.

Unlike **Kill**, **Wipe** preserves the track parameters like the name.

Like **Kill**, the events are not Cut to the scrap buffer, they are simply deleted, so *there is no way to get them back*.   The From and Thru markers are ignored -- *all* events in the selected track(s) are wiped out.

See Also:
Track Menu

Kill (Track menu)

Kills all events from all selected tracks and resets the track parameters to their default values. The events are not Cut to the scrap buffer, they are simply deleted, so *there is no way to get them back*. The From and Thru markers are ignored -- *all* events in the selected track(s) are killed.

See Also:
Wipe
Track Menu

Clone (Track menu)

Makes a copy of the selected track's events and/or all its parameters.

Useful for quickly making a back-up copy of a track before embarking on some major editing operation, or to double a part on a different MIDI channel.

See Also:
Track Menu

Sort (Track menu)

Sorts tracks based on the key value you pick.

Example 1: Pick *Name* for the key to **Sort By** and *Ascending* for the sort **Order**. Tracks will be sorted alphabetically by name.

Example 2: Pick *Size* for the key to **Sort By** and *Descending* for the sort **Order**. Tracks will be sorted in order from those with the most to those with the fewest events.

Sort uses a "stable" sort method, which means that it tends to preserve the ordering from the last time it was used. For instance, if you performed the two example sorts listed above, tracks would be sorted by decreasing Size, but any two tracks with the same Size would be sorted alphatbetically by Name. That's because the sort by Size tends to preserve the earlier sort by Name.

Blank tracks -- those with no parameters or events -- are moved to the end when sorting.

See Also:
Track Menu

Window menu

Tile in Rows
Tile in Columns
Cascade
Arrange Icons
Close All
Minimize All

Tile In Rows (Window menu)

Arranges all of the windows so that they're side-by-side (tiled) in rows.   This way, windows tend to be wider than they are higher.

See Also:
Window Menu

Tile In Columns (Window menu)

Arranges all of the windows so that they're side-by-side (tiled) in columns.   This way, windows tend to be higher than they are higher.

See Also:
Window Menu

Cascade (Window menu)

Arrange windows stacked on top of each other with at least the caption bar of each window visible.

See Also:
Window Menu

Arrange Icons (Window menu)

Arrange icons (minimized windows) so that they're in a neat row starting at the bottom left corner.

See Also:
Window Menu

Close All (Window menu)

Close all windows except the Track/Measure view.

See Also:
Window Menu

Minimize All (Window menu)

Minimize (make into an icon) all windows except the Track/Measure view.

See Also:
Window Menu

Settings menu

The Settings menu contains commands for setting various parameters and options.

- **Clock**
- **Metronome**
- **Timebase**
- **MIDI Devices**
- **Patch Names**
- **MIDI In**
- **MIDI Out**
- **MIDI Thru**
- **Record Filter**
- **Channel Table**
- **Time Format**
- **Key Bindings**
- **Initialization Files**
- **Auto Save**

Clock (Settings menu)

Change the clock source to one of the following:

Internal Clock

This follows the computer's internal clock.

MIDI Sync

MIDI synchronization makes Cakewalk follow another device that is sending MIDI Sync to the computer's MIDI In port.

SMPTE/MTC

SMPTE/MTC synchronization makes Cakewalk follow another device that is sending SMPTE/MTC timecode to the computer's MIDI In port. Some MIDI interfaces have a special SMPTE jack to read timecode directly. You may also use a device that converts SMPTE to MTC for sending over a MIDI cable.

See Also:

Settings Menu

"Synchronization" appendix in the *User's Guide*, for a full discussion of synchronization methods.

Metronome (Settings menu)

The left column of items lets you pick various options about when you want to hear the metronome.

The right column lets you specify a MIDI note that will be played as a metronome "tick". The default key number is that used by most drum machines for a closed hi-hat. MIDI channel 10 is a popular convention for drum machines, so that is also provided as a default.

To disable the MIDI metronome note entirely, specify 0 for the Velocity and/or Duration.

See Also:
Settings Menu

Timebase (Settings menu)

The timebase is also known as PPQ: "Pulses Per Quarter-note".   This is the number of clock ticks that a quarter-note is worth.   Generally, higher values provide more precision, but many people are not sensitive to much improvement using a timebase greater than 120 PPQ.   The computer has to work harder to provide higher timebases, so if you cannot hear the difference, it is probably best to stick with the default value of 120 PPQ.

See Also:
Settings Menu

MIDI Devices (Settings menu)

Input and Output devices

You may pick any number of available MIDI input and output devices. However:

- For **playback** to work, you must select at least one **Output** device.
- For **recording** to work, you must select at least one **Input** device.

The lists of devices show drivers which you have already installed using the "Drivers" icon of the Windows Control Panel. Cakewalk lists only MIDI input and output devices, not other kinds of devices.

MIDI Out devices are assigned to Cakewalk port numbers in the order they are listed. For example, if "MPU-401" is above "Sound Blaster" in the list and you select both, port 1 in Cakewalk is the MPU-401 and port 2 is the Sound Blaster. To rearrange the order, select one or more items, then press the **Move Selected to Top** button.

Sync Output Port

This is the port to which Cakewalk sends MIDI Sync, so that another device (for example, a drum machine) may slave to Cakewalk. If you are using such a device and want to sync it to Cakewalk, be sure to set this item to the number of the port to which the device is connected.

Timing Precision

If you have a 16 MHz 80386, you will want to change this option to **Low**. Otherwise, you should probably accept Cakewalk's default choice matching the class of your computer. Choosing a faster option will probably backfire, because it will make your computer work too hard and make the timing worse rather than better. Avoid the **Custom** option unless Twelve Tone Systems technical support has asked you to try specific values in response to a problem you are experiencing.

See Also:

Settings Menu

README.WRI file -- which you may view using **Help/View README.WRI** -- may contain specific instructions and tips for the kind of devices you're using.

Patch Names (Settings menu)

Cakewalk lets you assign names to patches. The default set of "names" are numbers from 0 to 127. You may also select numbers from 1 to 128, or lists of names that correspond to the factory presets of a number of popular synthesizers. If you don't see your instrument on the list, you may be able to use the "General MIDI" list, which is a standard layout that many newer instruments use.

Because you may have different kinds of instruments connected to various ports and channels, Cakewalk lets you specify which patch names to use for each port and channel. Once you set this up, the correct list will be used automatically for each port/channel.

Using the dialog box

You can select more than one port/channel at a time in the list on the left. If they share the same patch list, that will be selected on the right, otherwise nothing will be selected. To make a change, click on the patch list you want to use.

If you don't want to save your settings for the next Cakewalk session, remove the check from that option before pressing OK.

See Also:
Creating your own patch names
Settings Menu

MIDI In (Settings menu)

To ignore input on a particular MIDI channel, un-check the corresponding item.   When input is ignored, it won't be **recorded** or echoed by **MIDI Thru**.

See Also:
Settings Menu

MIDI Out (Settings menu)

This dialog lets you control various options related to information that Cakewalk sends to the MIDI Out port(s).

Send MIDI Start/Continue/Stop/Clock controls whether Cakewalk will transmit MIDI Sync. If you are not synchronizng another device like a drum machine to Cakewalk, leave this disabled.

Use Start, Never Continue lets you force Cakewalk to send a MIDI Start even when playback begins from other than the start (1:1:0).

Send MIDI SPP enables sending MIDI Song Position Pointer. This message lets devices synchronizing to Cakewalk jump to the correct spot.

Locate Delay for SPP Recipient can be set higher than zero to give slow devices more time to jump to the correct spot.

Zero Continuous Controllers sends messages every time playback stops to lift up the sustain pedal, center pitch bend wheels, and zero modulation wheels. It also sends a "reset all continuous controllers" message. Because this is sent for every MIDI channel of every port, it can take awhile.

Patch/Controller Searchback makes Cakewalk do some extra work before playback starts. It searches for the most recent patch change, pitch bend wheel, modulation wheel, and sustain pedal events on each MIDI channel of each MIDI port and sends them. This ensures that even if you jump to an arbitrary point in your song, any intervening patch, wheel, and pedal settings will be acted on. This option could take a long time if your song is very long.

See Also:
Settings Menu

MIDI Thru (Settings menu)

These options control how MIDI input is transformed and echoed to the MIDI output.   Auto mode, the default, is the most convenient.   It routes things based on the Port, Channel, Key+ and Vel+ parameters of the track you currently have the cursor on in the **Track/Measure view**.

See Also:
Settings Menu
"MIDI Thru" in the *User*'*s Guide* for a description of all Thru modes and options.

Record Filter (Settings menu)

This lets you filter out certain kinds of MIDI events.   By default, aftertouch events are filtered out.

See Also:
Settings Menu

Channel Table (Settings menu)

The Channel Table lets you optionally map certain MIDI channels onto certain tracks when recording.   Events on certain channels will end up on certain tracks.   This is most useful when "dumping" a sequence from another sequencer, or working with a guitar controller that transmits on separate channels at once.   (Note that Cakewalk also permits each track to contain events from different MIDI channels; you don't *have* to split things onto separate tracks.)

The dialog box has a checkbox for enabling/disabling the channel table and for assigning a track number to each MIDI channel.   The "Fill" button can save you typing: it looks at the number in the field you have the cursor on, and fills in the remaining fields with increasing values.

See Also:
Settings Menu

Time Format (Settings menu)

Select a SMPTE/MTC format.

The "Offset" option lets you specify a SMPTE time that will correspond to the beginning of your work.   In other words, this is the SMPTE time corresponding to musical time 1:1:0.

Hint: If you are syncing to SMPTE/MTC, you should stripe your tape beginning with SMPTE/MTC time 0:0:0:0 and set an offset of at least 2-3 seconds, or 0:0:3:0.   This gives the SMPTE tape reader enough time to establish a good sync lock-up before your song starts.

See Also:
Settings Menu
"Synchronization" Appendix in the *User's Guide*, for a full discussion of synchronization methods.

Key Bindings (Settings menu)

To save time, you may bind (assign) keys to commands and CAL programs that you frequently use.

There are two types of "keys" you may use: computer keyboard keys and MIDI keyboard keys. When you bind computer keyboard keys to commands, this is like the kind of "macro hotkey" feature you may be familiar with in other software. Using your MIDI keyboard is more novel, but the same basic idea.

You make both types of key bindings using the same dialog box: select which set you want to work with by picking one of the radio buttons at the top.

Computer keys

The available hotkeys include CTRL in combination with the A through Z keys, and the function keys (F1 through F12) by themselves or with CTRL or ALT pressed also. If you use choose a hotkey which Cakewalk has already assigned to a command, your binding will override Cakewalk's.

These bindings execute the command they're assigned to only when you could choose the command normally from the menu. If Cakewalk's menu is not available (for example when you're in a dialog box) or if the command is disabled, then the keys won't work either.

MIDI keys

The MIDI option uses a "shift" event to signal whether your MIDI keys should be treated as hotkeys. That way, you can still use all the MIDI keys to play notes. For this shift event, you may choose to use either a MIDI key (typically, the very lowest or highest key on your MIDI keyboard) or a controller event (typically, one of the pedals).

You may also temporarily disable MIDI macros by un-checking the **Enable** box, without having to clear all of the MIDI bindings.

See Also:
Settings Menu

Initialization Files (Settings menu)

This command lets you view and change certain options in Cakewalk intialization files. You may also change initialization files by opening and editing them in Windows Notepad, but you may find using **Initialization Files** easier.

To set or delete an option:

1. Pick the initialization file containing the option you wish to set or delete, by clicking one of the radio buttons in the **File** section of the dialog box.

2. Type the name of the option in the **Option** field.

   If the option already exists, you may pick it in the list of **Current Settings**, and that will fill in the **Option** and **Value** fields for you.

3. If you want to delete the option, press the **Delete** button.
   or
   If you want to set the option, type in the value for it in the **Value** field, then press the **Set** button.

4. When you are done viewing or changing options, press the **Close** button.

IMPORTANT: For most of these settings, Cakewalk looks at the initialization files only when it starts up. Therefore, to ensure that all of your changes take effect, you must exit Cakewalk and restart it.

See Also:
Settings Menu

Auto Save (Settings menu)

Cakewalk can periodically save your work to a special back-up file, named AUTOSAVE.WRK. You may control this using one or both of two options.

Use the **Minutes** item to say how many minutes may pass before Cakewalk automatically saves your file (assuming it's changed since last saved). Set **Minutes** to zero to disable this.

Use the **Changes** item to say how many changes you may make to the file before Cakewalk automatically saves it. Set **Changes** to zero to disable this.

If both **Minutes** and **Changes** are set to zero, no automatic save will occur. If both items are set to something other than zero, than whichever condition becomes true first will cause an automatic save to occur.

Your work is saved to a work file called AUTOSAVE.WRK. If you should need to retrieve your work after an error or system crash, use **Open** to load AUTOSAVE.WRK. You may then use **Save As** to save this using the name of the file you were originally working on. **Warning:** If you don't re-save soon, then **Auto Save** may write over AUTOSAVE.WRK! To avoid this possibility completely, you may wish to rename AUTOSAVE.WRK before restarting Cakewalk at all, using the Windows File Manager or the DOS RENAME command.

See Also:
Settings Menu

Answers to common questions

Here are answers to some of the most common questions about Cakewalk for Windows. Checking this list may save you a phone call to technical support!

Setup

- Why can't I hear anything on my synthesizer (no MIDI output)? Answer
- Why can't I record anything? (no MIDI input)? Answer
- How do I get a driver for my brand of MIDI interface? Answer

Customization

- How do I create my own set of Patch Names? Answer
- How and why should I use the Microsoft MIDI Mapper? Answer

Wave Audio

- How do I play back wave audio (.WAV files) files in my song? Answer
- Why can't I play wave audio with my Sound Blaster? Answer
- Why can't I play wave audio using my MPU-401 or Ad Lib card? Answer

Synchronization

- Why won't my drum machine won't sync to Cakewalk? Answer

Miscellaneous

- Why do I get the error message, "Cannot open timer", when I start Cakewalk? Answer

"Cannot open timer"

If you start Windows from a directory containing Cakewalk 4.0 (for DOS), then Windows will mistake Cakewalk 4.0's TIMER.DRV file for its own TIMER.DRV file located in the Windows system directory.   Windows silently fails to load its timer driver, and you discover the failure only when trying to use a program that needs the timing services, like Cakewalk for Windows.

To work around this problem with Windows, simply start Windows from another directory (for example, from the root directory or the Windows directory).

Other common questions....

Wave events and Sound Blaster

The Sound Blaster (ROM 2.0 and earlier) cannot do both MIDI input and wave output at the same time.   Thus, if you've selected "Creative Labs Sound Blaster" as a MIDI In device in Cakewalk's "Settings/MIDI Devices" dialog, wave audio won't work.   This includes Cakewalk's special Wave and MCIcmd events as well as wave audio attempted by any other Windows application running while the "Creative Labs Sound Blaster" MIDI In device is open. This is not a limitation of Cakewalk: while *any* program is using the "Creative Labs Sound Blaster" MIDI In device, wave audio won't work.

Note that MIDI *output* will work fine along with wave audio: you can select "Creative Labs Sound Blaster" from the list of MIDI *Out* devices.   The problem occurs only when you've selected the "Creative Labs Sound Blaster" as a MIDI *In* device.

Other common questions....

Required hardware for wave audio

There are basically three functions that multimedia audio hardware can provide:

(1) MIDI interface (link to external synthesizers)

(2) Tone generators (on-board synthesizers)

(3) Digital audio (wave form audio, like a CD or DAT recorder/player)

Various multimedia audio cards provide one or more of these three functions. The MPU-401 is "only" a MIDI interface. The Ad Lib is "only" a tone generator. The Sound Blaster and MultiSound cards combine all three functions.

Wave events require the third function, **digital audio**. To use Wave events, you must install a sound card with digital audio capabilities and install a wave driver for the card using Windows Control Panel. For more information, consult the manufacturer of the sound card you own or are considering purchasing.

Other common questions....

No MIDI output

If you aren't getting any MIDI output, please run through this checklist:

1. Make sure you've connected your cables properly. See the instructions in the Cakewalk *Installation Guide*.

2. Make sure you install a MIDI driver for your interface using Windows Control Panel. See the instructions in the Cakewalk *Installation Guide*. Make sure you specify the correct configuration information -- like IRQ and base port address -- in the driver's setup dialog box.

3. Now the driver is available for Windows programs to use. Next, you need to tell Cakewalk to use it. Choose the **MIDI Devices** command on the **Settings** menu and make sure the device is selected (highlighted) in the MIDI Out list. Click on the device name to be sure it is selected; if the name only has a dotted box around it but isn't drawn in reverse video, then it is not selected!

4. Try loading and playing a sample workfile included with Cakewalk. Make sure that each track is set to a **Port** and **Channel** to which you've connected a synthesizer. Because you can change the Port and Channel during playback, you can experiment with different settings easily until you hear a sound.

Other common questions....

No MIDI input

If you aren't getting any MIDI input, please run through this checklist:

1. Make sure you've connected your cables properly. See the instructions in the Cakewalk *Installation Guide*.

2. Make sure you install a MIDI driver for your interface using Windows Control Panel. See the instructions in the Cakewalk *Installation Guide*. Make sure you specify the correct configuration information -- like IRQ and base port address -- in the driver's setup dialog box.

3. Now the driver is available for Windows programs to use. Next, you need to tell Cakewalk to use it. Choose the **MIDI Devices** command on the **Settings** menu and make sure the device is selected (highlighted) in the MIDI In list. Click on the device name to be sure it is selected; if the name only has a dotted box around it but isn't drawn in reverse video, then it is not selected!

Other common questions....

Microsoft MIDI Mapper

The Microsoft MIDI Mapper is intended to help make MIDI song files be "portable" from one system to the next by mapping patch numbers, key numbers, and so on from a standard General MIDI layout to whatever synthesizer a particular user has. If you are not preparing or using MIDI song files for multimedia distribution purposes, you may not need to use the MIDI Mapper at all: it may be solving a problem that you don't have and needlessly complicating your setup.

If you *do* need the Microsoft MIDI Mapper, be sure to read Microsoft's documentation for it. Also, feel free to call Microsoft Windows Technical Support for assistance using their product.

To get you started, however: One key point is that you have Cakewalk use the MIDI Mapper instead of directly using the device. The whole idea is for the MIDI Mapper to sit between Cakewalk and the device and map data on the way. For example, in Cakewalk's **MIDI Devices** list you would select "Microsoft MIDI Mapper" but not select "Sound Blaster". Instead, select "Sound Blaster" in the MIDI Mapper, using the Windows Control Panel. So you set up a chain: Cakewalk sends data to the MIDI Mapper, the MIDI Mapper transforms the data and sends it to the Sound Blaster.

Other common questions....

Creating your own Patch Names list

Cakewalk lets you associate a list of patch names with each port/channel. You make this association using the **Patch Names** command. A number of lists for some popular synthesizers are provided.

If you wish to create your own list, you need to edit the PATCHES.INI file supplied with Cakewalk.

1. You may want to make a backup copy of this file before changing it.

2. Load the PATCHES.INI file into a text editing program. (*Note:* The Windows Notepad cannot handle text files as large as PATCHES.INI. Instead, use the EDIT command in MS-DOS 5.0 or higher, or, some other text editing application.)

3. Follow the instructions at the beginning of the file.

4. After adding your patch names to the file, save it and restart Cakewalk. Cakewalk will notice the PATCHES.INI file is newer, and it will "compile" the file into a PATCHES.BIN file that can be read much more quickly. As long as PATCHES.BIN exists and is newer than PATCHES.INI, Cakewalk will use PATCHES.BIN.

5. Choose **Patch Names**. You should see your new patch name list among the others. Associate your list with each port/channel that you desire.

Other common questions....

Devices and drivers

Windows 3.1 provides the Multimedia Extensions.   Manufacturers of MIDI interfaces, sound cards, and other multimedia hardware can create drivers.   Any Windows program can use these drivers to know how to work with a particular model of sound device.   Cakewalk uses these standard drivers.

We have supplied a couple drivers with Cakewalk only as an interim provision to help the owners of a few of the most popular MIDI interfaces.   Eventually even these drivers will be available directly from the maker of the MIDI interface.

If your MIDI interface is not in this list, you need to contact its manufacturer to obtain a Windows driver for it, otherwise Cakewalk will not know how to work with it.

Other common questions....

Using wave audio (.WAV files)

Cakewalk lets you trigger playback of digital audio at various times in your song. You must have installed a digital audio sound card like a Sound Blaster, *and* installed a wave driver for it using the Windows Control Panel "Drivers" icon.

You trigger digital audio playback by inserting a special kind of event in a track at the time you want. There are two kinds of events you may use, Wave or MCIcmd. There are pros and cons for each.

You may find it easiest to keep these special events on a track all by themselves. Here are step-by-step instructions:

1. Open an **Event List view** for the track in which you want to insert the event.

2. Press INSERT to add a new event.

3. Move the cursor to the **Kind** column and press ENTER (or, double-click on the **Kind** cell with the mouse.)

4. In the **Event Kind dialog**, choose **Wave Audio** or **MCI Command**, and press OK.

   - If you selected **Wave audio**, then you see the Wave edit dialog. Choose **Load Wave from File** and then pick a .WAV file.

     -or-

   - If you selected **MCI Command**, then you must type the text for the MCI command in the Event-list view. Move the highlight one cell to the right -- press the RIGHT key once -- and press ENTER. Enter the MCI command, "PLAY <*filename*>", where <*filename*> is a .WAV file. For example, you could use the command, "PLAY C:\WINDOWS\CHIMES.WAV", to play the CHIMES.WAV file included with Windows 3.1 and located in the C:\WINDOWS directory.

Tip: To test your new event, you may step-play it, as explained in the help topic for the **Event List view**.

Tip: During playback, only events from checked (non-muted) tracks are played. Make sure you change the track **Status** in the **Track/Measure view** to hear your Wave or MCIcmd event. If the track has an "m", it's muted, so double-click on the "m" to change it to a checkmark.

Other common questions....

Slaving drum machines to Cakewalk's MIDI Sync

Cakewalk can either slave to MIDI Sync, or be the master providing MIDI Sync for something else to sync to.   For the latter -- for example, if you are using a drum machine which you want to follow Cakewalk -- go through the following checklist:

1. On most models of drum machines, you can select whether or not it should slave to MIDI Sync.   If this can be switched off, make sure it's switched ON.

2. Double-check the connection from your computer's MIDI OUT jack to the drum machine's MIDI IN.

3. If you are using more than one MIDI output port on your computer, you need to tell Cakewalk which of these ports to send MIDI Sync info to.   Do this in the **MIDI Devices** dialog's **Sync Output Port** item: specify the port to which you've connected the drum machine.

4. Go into the **MIDI Out dialog**, and make sure the **Transmit MIDI Start/Continue/Stop/Clock** option is ON.   This must be turned on for each new workfile you create: it is off by default.

Other common questions....

CAL Function Reference

General CAL functions

- CAL data types
- Math functions
- Relational functions
- Boolean functions
- Assignment functions
- (forEachEvent)
- Event parameter variables
- Marker variables
- Buffer functions
- Musical time functions
- Output functions
- Input functions
- Control flow functions
- (include)
- (DLL)
- (sendMIDI)
- (delay)

CAL functions corresponding to Cakewalk user commands

- File menu
- Edit menu
- Track menu
- Settings menu

See Also:

CAL view

File menu CAL functions

- FileNew
- FileOpen
- FileSave
- FileSaveAs
- FileMerge
- FileExtract

These correspond to commands available on the **File menu**.

See Also:
CAL Function Reference

(FileNew)

```
(FileNew [<szTemplate>])
```

If *szTemplate* argument is omitted, then the user is prompted for the name of a template. The *szTemplate* argument is a string and should contain only the base name of the template, without an extension.   For example "Normal" is correct but "NORMAL.TPL" is not.

Returns: FALSE if the template was not found and loaded.   This could be either because of an error reading the file or because the user pressed Cancel in the dialog.   The "Normal" template always works: if no NORMAL.TPL file is found, a blank work is created with the normal program defaults.

See Also:
New
CAL Function Reference

(FileOpen)

```
(FileOpen [<szPathname>])
```

If   *szPathname* argument is omitted, then the user is prompted for the name.

Returns: FALSE if file was not loaded.   This could be either because of an error reading the file or because the user pressed Cancel in the dialog.

See Also:
Open
CAL Function Reference

(FileMerge)

```
(FileMerge [<szPathname>])
```

If *szPathname* argument is omitted, then the user is prompted for the name.

Returns: FALSE if file was not loaded.   This could be either because of an error reading the file or because the user pressed Cancel in the dialog.

See Also:
Merge
CAL Function Reference

(FileSaveAs)

```
(FileSaveAs [<szPathname>])
```

If *szPathname* argument is omitted, then the user is prompted for the name.

If *szPathname* already exists, then user is warned and given the ability to enter a different name.

Returns: FALSE if file was not saved.   This could be either because of an error writing the file or because the user pressed Cancel in the dialog.

See Also:
Save As
CAL Function Reference

(FileSave)

```
(FileSave)
```

Returns: FALSE if file was not saved due to an error writing the file.

See Also:

Save

CAL Function Reference

(FileExtract)

```
(FileExtract [<szPathname>])
```

If *szPathname* argument is omitted, then the user is prompted for the name.

If *szPathname* already exists, then user is warned and given the ability to enter a different name.

Returns: FALSE if file was not saved.   This could be either because of an error writing the file or because the user pressed Cancel in the dialog.

See Also:
Extract
CAL Function Reference

Edit menu CAL functions

- EditCopy
- EditCut
- EditPaste
- EditPasteToTrack
- EditQuantize
- EditSlide
- EditTranspose
- EditLength
- EditRetrograde
- EditControlFill
- EditVelocityScale
- EditFitImprov
- EditFitToTime
- EditInterpolate
- ResetFilter
- SetFilterKind
- SetFilterRange

These correspond to commands available on the **Edit menu**.

Most of the Edit menu command functions share certain arguments, which will be explained only here.

(An "argument" is one of the parameters passed to a function.)

tFrom corresponds to the "From:" item.

tThru corresponds to the "Thru:" item.

bEvents corresponds to the "Events" checkbox, either TRUE or FALSE.

bEvtFilt corresponds to the "Use Event Filter..." checkbox, either TRUE or FALSE.

bTempos corresponds to the "Tempo Changes" checkbox, either TRUE or FALSE.

bMeters corresponds to the "Meter/Key Changes" checkbox, either TRUE or FALSE.

All of the arguments to a command function correspond to items in the dialog box, and you can generally easily guess which by the name used to describe the argument here.

See Also:
CAL Function Reference

(EditCopy)

```
(EditCopy [<tFrom> <tThru> <bEvents> <bEvtFilt> <bTempos> <bMeters> <bMarkers>])
```

See Also:

Edit menu CAL functions

Copy

CAL Function Reference

(EditCut)

```
(EditCut [<tFrom> <tThru> <bEvents> <bEvtFilt> <bTempos> <bMeters> <bMeters>
<bDelHole>])
```

See Also:

Edit menu CAL functions

Cut

CAL Function Reference

(EditPaste)

```
(EditPaste [<tTo> <wReps> <nHoleKind> <bEvents> <bTempos> <bMeters><bMarkers>])
```

nHoleKind is

0	blend with existing material
1	replace existing material
2	slide over existing to make a hole for new material

See Also:
Paste
Edit menu CAL functions
CAL Function Reference

(EditPasteToTrack)

```
(EditPasteToTrack [<tTo> <wReps> <nHoleKind> <bEvents> <bTempos> <bMeters> <bMarkers>
<nTrack>])
```

nHoleKind is

0	blend with existing material
1	replace existing material
2	slide over existing to make a hole for new material

nTrack varies depending on nMode.

multi-track mode:	paste tracks back shifted by this amount
single track mode:	paste all track to this track number

See Also:

Paste To One Track
Edit menu CAL functions
CAL Function Reference

(EditQuantize)

```
(EditQuantize [<tFrom> <tThru> <bEvtFilt> <wRes> <nPct> <bTimes> <bDurs>])
```

See Also:

Quantize

Edit menu CAL functions

CAL Function Reference

(EditSlide)

```
(EditSlide [<tFrom> <tThru> <bEvtFilt> <nAmount> <bUnitTicks>])
```

If *bUnitTicks* is TRUE, then *nAmount* is treated as ticks; otherwise it is treated as measures.

See Also:
Slide
Edit menu CAL functions
CAL Function Reference

(EditTranspose)

```
(EditTranspose [<tFrom> <tThru> <bEvtFilt> <nAmount> <bDiatonic>)
```

See Also:

Transpose

Edit menu CAL functions

CAL Function Reference

(EditLength)

```
(EditLength [<tFrom> <tThru> <bEvtFilt> <nPct> <bStart> <bDur>])
```

See Also:

Length

Edit menu CAL functions

CAL Function Reference

(EditRetrograde)

```
(EditRetrograde [<tFrom> <tThru> <bEvtFilt>])
```

See Also:

Retrograde

Edit menu CAL functions

CAL Function Reference

(EditControlFill)

```
(EditControlFill [<tFrom> <tThru> <nCtrl> <nChan> <nBeg> <nEnd>])
```

nChan is from 1..16.

nBeg and *nEnd* must be 0..127.

See Also:

Controller Fill
Edit menu CAL functions
CAL Function Reference

(EditVelocityScale)

```
(EditVelocityScale [<tFrom> <tThru> <bEvtFilter> <nBeg> <nEnd> <bUnitsPercent])
```

bUnitsPercent if TRUE indicates that *nBeg* and *nEnd* are percentages, otherwise if FALSE that they are velocity values.

See Also:
Velocity Scale
Edit menu CAL functions
CAL Function Reference

(EditFitImprov)

```
(EditFitImprov [<nReferenceTrack>])
```

See Also:

Fit Improvisation
Edit menu CAL functions
CAL Function Reference

(EditFitToTime)

```
(EditFitToTime [<tFrom> <tThru> <tNewThru> <nMethod>])
```

See Also:

Fit To Time

Edit menu CAL functions

CAL Function Reference

(EditInterpolate)

```
(EditInterpolate)
```

This never presents dialog boxes to the user.   It presumes the search (1) and replace (2) filters have already been set as desired.

See Also:
Interpolate
ResetFilter
SetFilterKind
SetFilterRange
Edit menu CAL functions
CAL Function Reference

(ResetFilter)

```
(ResetFilter <nFilter> <bEverything>)
```

nFilter is one of the following:

0	Used in all Edit menu commands except Interpolate
1	Interpolate's Search filter
2	Interpolate's Replace filter

See Also:

Edit menu CAL functions

CAL Function Reference

(SetFilterKind)

```
(SetFilterKind <nFilter> <nKind> <bInclude>)
```

nFilter is one of the following:

0	Used in all Edit menu commands except Interpolate
1	Interpolate's Search filter
2	Interpolate's Replace filter

nKind is one of the following kind-of-event constants:

NOTE

KEYAFT

CONTROL

PATCH

CHANAFT

WHEEL

bInclude says whether to include this kind of event.

See Also:
Edit menu CAL functions
CAL Function Reference

(SetFilterRange)

```
(SetFilterRange <nFilter> <nRange> <bInRange> <wMin> <wMax>)
```

nFilter is one of the following:

0	Used in all Edit menu commands except Interpolate
1	Interpolate's Search filter
2	Interpolate's Replace filter

nRange is one of the following:

0	Note Key
1	Note Vel
2	Note Dur
3	Keyaft Key
4	Keyaft Val
5	Control Num
6	Control Val
7	Patch
8	ChanAft
9	Wheel
10	Chan,
11	Beat
12	Tick

bInRange is the *opposite* of the Not checkbox: FALSE means Not is checked, TRUE means it's not checked.

wMin, *wMax* are the minimum and maximum values for the range.

See Also:
Edit menu CAL functions
CAL Function Reference

Track menu CAL functions

- TrackName
- TrackLoop
- TrackActive
- TrackArchive
- TrackKey+
- TrackVel+
- TrackTime+
- TrackPort
- TrackChan
- TrackPatch
- TrackVolume
- TrackPan
- TrackSelect

These generally correspond to commands available on the **Track menu** which are related to specific track parameters.

All of these have a third argument, *nTrack*, which is optional. If omitted, then all selected, non-blank tracks are affected -- just as the Track menu commands work. If a specific track number argument is supplied, then only that track is affected, and if the track is blank, it is created prior to applying the setting.

Note: Track numbers are 0 through 255 when you're writing a CAL program, not 1 through 256.

Example 1: Set the Key+ parameter of all selected, non-blank tracks, to -10.

```
(TrackKey+ -10)
```

Example 2: Set the Key+ parameter of track 0 only to -10.

```
(TrackKey+ -10 0)
```

See Also:
CAL Function Reference

(TrackName)

```
(TrackName [<szName> [<nTrack>]])
```

szName is either a string literal or a string created by the (format) function.

nTrack is a number from 0 to 255.

Example: Set the name of a track to something including its number in the name:

```
(int nTrack 1)
(getInt nTrack "Number of track to name?" 1 256)
; Give the track the name "Track N", where N is the track number.
; Note that (TrackName) expects a number in the range 0..255, not 1..256.
(TrackName (format "Track " nTrack) (- nTrack 1))
```

See Also:

Name

CAL Function Reference

(TrackLoop)

```
(TrackLoop [<nReps> [<nTrack>]])
```

nReps must be 1..9999.

See Also:
Loop
CAL Function Reference

(TrackActive)

```
(TrackActive [<nActive> [<nTrack>]])
```

nActive must be be one of the following:

1	playing
0	mute
-1	toggle

See Also:
Status
CAL Function Reference

(TrackArchive)

```
(TrackArchive [<nArchive> [<nTrack>]])
```

nArchive must be be one of the following:

1	archive
0	un-archive
-1	toggle

See Also:
CAL Function Reference

(TrackKey+)

```
(TrackKey+ [<nAmount> [<nTrack>]])
```

nAmount must be -127..127.

See Also:
Key+
CAL Function Reference

(TrackVel+)

```
(TrackVel+ [<nAmount> [<nTrack>]])
```

nAmount must be -127..127.

See Also:
Vel+
CAL Function Reference

(TrackTime+)

```
(TrackTime+ [<lAmt> [<nTrack>]])
```

See Also:

Time+

CAL Function Reference

(TrackPort)

```
(TrackPort [<nPort> [<nTrack>]])
```

nChan must be 0..15.

See Also:
Port
CAL Function Reference

(TrackChan)

```
(TrackChan [<nChan> [<nTrack>]])
```

nChan must be 0..15.

See Also:
Channel
CAL Function Reference

(TrackPatch)

```
(TrackPatch [<nPatch> [<nTrack>]])
```

nPatch must be -1..127.

See Also:
Patch
CAL Function Reference

(TrackVolume)

```
(TrackVolume [<nVolume> [<nTrack>]])
```

nVolume must be -1..127.

See Also:
Volume
CAL Function Reference

(TrackPan)

```
(TrackPan [<nPan> [<nTrack>]])
```

nPan must be -1..127.

See Also:
Pan
CAL Function Reference

(TrackSelect)

```
(TrackSelect <nState> <nTrack>)
```

nState must be one of the following:

1	select
0	un-select
-1	toggle

See Also:

CAL Function Reference

Settings menu CAL functions

SettingsMetronome
SettingsMidiThru
SettingsMidiOut
SettingsRecordFilter
SettingsChannelTable

These correspond to some of the commands available on the **Settings menu**.

See Also:
CAL Function Reference

(SettingsMetronome)

```
(SettingsMetronome [<bPlay> <bRec> <bAcc> <nCount> <nPort> <nChan> <nKey> <nVel>
<wDur> <bBeep>])
```

See Also:

Metronome

CAL Function Reference

(SettingsMidiThru)

```
(SettingsMidiThru [<nMode> <nPort> <nChan> <nKey> <nVel> <nLocalOnPort>])
```

See Also:

MIDI Thru

CAL Function Reference

(SettingsMidiOut)

```
(SettingsMidiOut [<bTxMidiRt> <bSendCont> <bSendSPP> <nSPPDelay> <bCtrlZero>
<bCtrlChase>])
```

See Also:

MIDI Out

CAL Function Reference

(SettingsRecordFilter)

```
(SettingsRecordFilter [<bNote> <bKeyAft> <bControl> <bPatch> <bChanAft> <bWheel>])
```

See Also:

Record Filter

CAL Function Reference

(SettingsChannelTable)

```
(SettingsChannelTable [<bOn> <n1> <n2> <n3> <n4> <n5> <n6> <n7> <n8> <n9> <n10> <n11>
<n12> <n13> <n14> <n15> <n16>])
```

n1 through *n16* are track numbers, 0..255.

See Also:
Channel Table
CAL Function Reference

CAL Data Types

int

An int is a 16-bit signed integer. Values range from 32676 to 32767. (Just like an int in C or an integer in PASCAL.)

word

A word is a 16-bit unsigned integer. Values range from 0 to 65536. (Just like an unsigned short int in C.)

long

A long is a 32-bit signed integer. Values range from about negative 2 billion to positive 2 billion. (Just like a long int in C.)

dword

A dword is a 32-bit unsigned integer. Values range from 0 to over 4 billion. (Just like an unsigned long int in C.)

string

A string is text enclosed in quotes.

Variable declarations

Variables must be declared before they are used. Variables are declared in the form

```
        (<type> <name> [<initial-value>])
```

Examples:

```
(int <variable-name> [initial-value])
(word <variable-name> [initial-value])
(long <variable-name> [initial-value])
(dword <variable-name> [initial-value])
(string <variable-name> [initial-value])
(undef <variable-name>)                 ; un-defines a variable of any type
```

The optional *initial-value* part lets you initialize a variable. For example, this declares an int variable named "i" and initializes it to zero:

```
        (int i 0)
```

If you don't initialize a variable, its value cannot be predicted.

Because of the LISP-like syntax, all CAL needs to delimit variable names is a space. So, variable names may be any sequence of printable ASCII characters.

See Also:

CAL Function Reference

Math functions

```
(+ <op1> <op2>)            ; add
(- <op1> <op2>)            ; subtract
(* <op1> <op2>)            ; multiply
(/ <op1> <op2>)            ; divide
(% <op1> <op2>)            ; modulo (remainder after division)

(++ <variable>)            ; increment (add 1 to variable)
(-- <variable>)                  ; decrement (subtract 1 from variable)

(random <min> <max>)       ; returns pseudo-random # min..max inclusive
```

See Also:

CAL Function Reference

Relational functions

```
(< <op1> <op2>)            ; less than

(> <op1> <op2>)            ; greater than

(<= <op1> <op2>)           ; less than or equal to

(>= <op1> <op2>)           ; greater than or equal to

(== <op1> <op2>)           ; equal

(!= <op1> <op2>)           ; not equal
```

Relational functions compare two values. The result is either zero (false) or non-zero (true). For example:

```
(< 11 12)
```

is true. Eleven is less than twelve, and the function returns a non-zero number. On the other hand,

```
(> 11 12)
```

is not true. Eleven is not less than twelve, and the function returns zero.

See Also:
CAL Function Reference

Boolean functions

```
(&& <op1> <op2>)           ; logical AND.  If both operands are non-zero, returns non-
zero.
(|| <op1> <op2>)           ; logical OR.  If either or both operands are non-zero,
returns non-zero
```

See Also:

CAL Function Reference

Assignment functions

```
(= <variable> <expr>)
```

This is how you give a variable a value.   For example, to assign a variable named "foobar" the value 12:

```
       (= foobar 12)              ; foobar is now 12
```

To assign "foobar" the value of (+ 10 10):

```
       (= foobar (+ 10 10))       ; foobar is now 20
```

The following assignment operators let you perform an arithmetic operation and an assignment in one fell swoop:

```
(+= <variable> <value>)    ; (= <variable> (+ <variable> <value>))

(-= <variable> <value>)    ; (= <variable> (- <variable> <value>))

(*= <variable> <value>)    ; (= <variable> (* <variable> <value>))

(/= <variable> <value>)    ; (= <variable> (/ <variable> <value>))

(%= <variable> <value>)    ; (= <variable> (% <variable> <value>))
```

Example:

```
       (+= foobar 1)
```

is the same as:

```
       (= foobar (+ foobar 1))
```

See Also:
CAL Function Reference

(forEachEvent)

```
(forEachEvent

	<action>

)
```

The *action* is evaluated for (applied to) each event in the selected region.   In other words, the *action* expression is inside of an implicit loop cycling through the events in the From..Thru range of each selected track.   The event parameter variables hold the values for the current event.

Example: Ask the user for an amount, then add that amount to the velocity of all Note events in the selected region.

```
	(do

		(int amt 0)

		(getInt amt "Amount to add?" -127 127)

		(forEachEvent

			(if (== Event.Kind NOTE)

				(+= Note.Vel amt)

			)

		)

	)
```

See Also:
CAL Function Reference

Event parameter variables

CAL has some predefined "magic" variables that let you access events in Cakewalk Professional's track buffers. These variables work only in the (forEachEvent) expression, where they're automagically set to the value of the current event's parameters. This is how you get and set event parameters. You can use them in expressions to find out about the current event. If you assign them a new value, they automagically change the current event.

Different types of events have different parameters. For instance, a note event has a key number, a velocity, and a duration, but a patch event has only a patch change number. Consequently, some of these magic variables are defined only for particular kinds of events, while others -- e.g., the variable for the time of the event --apply to all kinds of events.

Variables for all kinds of events

```
Event.Chan
```

The MIDI channel of the event (0..15).

```
Event.Time
```

The starting time of the event. The type of this variable is dword. See Musical Time Functions for more information.

Kind (type of event)

```
Event.Kind
```

The kind (or type) of the event. One of the following:

```
        NOTE, PATCH, CONTROL, CHANAFT, KEYAFT, WHEEL, SYSX
```

Variables for particular kinds of events

The preceding variables apply to all kinds of events.

The following variables let you get or set an event parameter that a particular kind of event has. So, you should test the value of Event.Kind, then use only the variables that are meaningful for an event of that kind.

Variable	Event.Kind	Meaning	Allowed values
Note.Key	NOTE	Key number (pitch)	0 .. 127
Note.Vel	NOTE	Velocity	0 .. 127
Note.Dur	NOTE	Duration	0 .. 65535
Patch.Num	PATCH	Patch number	0 .. 127
Control.Num	CONTROL	Controller number	0 .. 127
Control.Val	CONTROL	Controller value	0 .. 127
KeyAft.Key	KEYAFT	Key number	0 .. 127
KeyAft.Val	KEYAFT	Value (pressure amount)	0 .. 127
ChanAft.Val	CHANAFT	Value (pressure amount)	0 .. 127
Wheel.Val	WHEEL	Wheel value	-8192 .. 8191

Example: Make sure the event is a note event. If it is, add two to the key number (raise the pitch two half-steps):

```
        (if (== Event.Kind NOTE)          ; Is it a NOTE event?
                (= Note.Key (+ Note.Key 2))              ; Yes: change it!
        )
```

See Also:

CAL Function Reference

Marker variables

There are some other "automagical" variables that equate to Cakewalk Professional's From, Now, and Thru markers:

```
From

Now

Thru
```

You can use them for their values anywhere in the CAL program, but you may not assign new values to them in a forEachEvent expressions.

Example:

```
(getTime From "Change From marker time:")
```

or, to save, change, and restore:

```
(dword oldFrom From)

(getTime From "Change From marker time:")

(forEachEvent

      ; ... do some stuff ....

)

(= From oldFrom)
```

Note: Cakewalk Professional never allows (> From Thru) or (< Thru From). Neither does CAL. The other marker is automatically adjusted, if necessary, to prevent this.

Note: Changing the Now marker has no effect on what events the forEachEvent operates on. Changing the From and Thru markers will.

See Also:

CAL Function Reference

Buffer functions

index

```
(index)
```

Returns the index of the event in the track buffer. Events are numbered starting from zero.

delete

```
(delete)
```

Deletes the current event. Indexes for subsequent events are automatically decremented.

insert

```
(insert <time> <chan> <kind> [ ...data parameters... ])
```

Inserts a copy of the event at the specified index. As you can see, this function takes a variable number of arguments, depending on the <kind> argument:

```
(insert <time> <chan> NOTE <key> <vel> <dur>)

(insert <time> <chan> KEYAFT <key> <amt>)

(insert <time> <chan> CONTROL <num> <val>)

(insert <time> <chan> CHANAFT <amt>)

(insert <time> <chan> PATCH <num>)

(insert <time> <chan> WHEEL <val>)
```

Example:

```
(insert (makeTime 1 1 0) 10 NOTE 60 64 TIMEBASE))
```

Inserts at time 1:1:0 on channel 10 a note event of MIDI key 60 with a velocity of 64 and a duration of TIMEBASE.

Hint: The TIMEBASE constant is described in the "Musical Time Functions" section.

The track that the event is inserted on depends on whether you use it in forEachEvent expression or not. If you use it in a forEachEvent expression, then the event is inserted in the track which that expression is currently evaluating. Otherwise, it is inserted in the current track (the track which the cursor is on in the Track/Measure view).

In other words, use (insert) in the forEachEvent for inserting events in any and all selected tracks. For example, you might want to add a controller event wherever you find a certain note event.

For commands where you want the user to be able to add events to a single track -- the track the highlight is on in the Track/Measure view -- use (insert) outside a forEachEvent.

Note: The inserted events do not become visible to the CAL program. Consequently, the forEachEvent is applied only to events existing in tracks before the command was selected.

Note: Because this command adds events, it is possible for it to exhaust free memory. If that happens, a CAL "out of memory" error message is displayed and the CAL program is terminated.

See Also:

CAL Function Reference

Musical time functions

Converting between Measure:Beat:Tick and "raw" times

```
(meas <time>)

(beat <time>)

(tick <time>)

(makeTime <measure> <beat> <tick>)
```

These are conversion functions for dealing with Cakewalk Professional times.

Although Cakewalk Professional displays times to the user in Measure:Beat:Tick format, internally they are stored as a single dword. CAL variables like Event.Time deal with these "raw" dwords too.

These functions let you convert between the two forms. (meas <time>), (beat <time>), and (tick <time>) return the measure, beat, and tick for a given raw dword time. (make time <meas> <beat> <tick>) makes up a dword time from a measure, beat, and tick. This conversion depends on the current meter, of course, as well as the timebase.

Important: Measures and beats are numbered from 1, not 0. You will get a "value out of range" error if you pass (make time) a 0 for the measure or beat arguments.

TIMEBASE constant

TIMEBASE is a constant that is equal to the current timebase as set by the **Timebase** command. Remember that the timebase is the number of ticks per quarter-note.

See Also:
CAL Function Reference

Output functions

message

```
(message <value> [ [value] ... ])
```

Takes a variable number of arguments, which can be int, word, dword, or string expressions (variables or values). Prints the result on Cakewalk Professional's message line (the bottom line of the screen).

Example:

```
(do
	(int foobar 10)
	(message "Foobar is " foobar " and the timebase is " TIMEBASE)
)
```

Assuming that Cakewalk Professional's timebase is set to 120, the preceding CAL code would display the following on the message line:

Foobar is 10 and the timebase is 120

pause

```
(pause <value> [ [value] ... ])
```

Works just like **(message)** but also pauses for a user keypress or mouse click. If the user presses escape, the CAL program will terminate. If the user presses any other key, the CAL program resumes.

Tip: This is handy for debugging CAL programs. Use **(pause)** to display a variable value; that way you can press Cancel to break out of long loops.

format

```
(format <value> [ [value] ... ])
```

Works just like **(message)** and **(pause)** but simply returns a string, which you may pass to a function like **(TrackName)**.

See Also:
CAL Function Reference

Input functions

These functions let you prompt the user for input. They display a standard Cakewalk Professional dialog box. After the user presses Enter, the CAL program resumes and the function returns the value input.

```
(getInt <int-variable> <prompt> <minimum> <maximum>)
```

Returns the int value that the user input.

int-variable is an int variable that will receive the input value. Its value is displayed as the default value in the dialog box.

prompt is a string item that will be displayed in the dialog box.

minimum and *maximum* are int values; CAL will not let the user input a number less than or greater than these values.

Example: Ask the user to input "A number from 0 through 100", suggesting a default of 50, and assign the result to a variable "i":

```
(do
        (int i 50)
        (getInt i "A number from 0 thru 100" 0 100)
)
```

```
(getWord <word-variable> <prompt> <minimum> <maximum>)
```

Returns the word value that the user input.

word-variable is an word variable that will receive the input value. Its value is displayed as the default value in the dialog box.

prompt is a string item that will be displayed in the dialog box.

minimum and *maximum* are word values; CAL will not let the user input a number less than or greater than these values.

```
(getTime <dword-variable> <prompt>)
```

Returns the dword value that the user input.

dword-variable is a dword variable that will receive the input value. Its value will be displayed as the default value in the dialog box.

prompt is a string item that will be displayed in the dialog box.

Example: Ask the user to input a "Starting time", suggesting as a default the time of the current event, and assign the result to a variable "t".

```
(do
        (dword t Event.Time)
        (getTime t "Starting time:")
)
```

See Also:
CAL Function Reference

Control flow functions

do

```
(do
             <expr1>
      [      <expr2>
             ...
             <exprN> ]
)
```

Evaluates each expression in sequence, once and only once.

Tip: Use this whenever CAL expects a single expression but you want to use two or more.

if

```
(if <condition>
      <true-expression>
      [<false-expression>]
)
```

If *condition* is non-zero, evaluates the *true-expression*, else evaluates the *false-expression*. The *false-expression* is optional.

while

```
(while <condition>
      <action>
)
```

So long as *condition* is non-zero, *action* is evaluated. If *condition* is zero the first time, *action* is never evaluated.

Example: Count-down from 100 to 1:

```
(do
      (int count 100)
      (while (!= count 0)
             (do
                   (message "Count-down is now " count)
                   (-- count)
             )
      )
)
```

switch

```
(switch <index>
             <case1> <case1result>
       [     <case2> <case2result>
             <case3> <case3result>
             ...
             <caseN> <caseNresult> ]
)
```

Evaluates *index*, then compares it to the first of each of the following pairs of expressions. When it finds the first match, if any, it executes the second half of the pair and returns the result.

CAL doesn't have an explicit "default" or "else" case statement. But you can achieve the same thing by making *index* itself the very last of the case pairs. This works because **switch** matches the first pair it finds, then quits. If nothing matches before the *index* case, why then, *index* always equals itself -- so its consequent will be evaluated if and only if no other pairs matched. For instance:

```
(switch TIMEBASE
       120                 (pause "120")       ; Is it 120?
       96                  (pause "96")        ; Is it 96?
       TIMEBASE     (pause TIMEBASE)    ; It is anything else not already matched
above
)
```

See Also:
CAL Function Reference

(include)

```
(include <szPathName>)
```

Runs another CAL program as if the content of that file were inserted in the program at the point where the (include) function is placed.

Example: Include the file NEED20.CAL, which is one of the supplied example files. The purpose of NEED20.CAL is to check that CAL version 2.0 or higher is being used, and if not, warn the user and stop the CAL program.

```
(do

        (include "NEED20.CAL")     ; require VERSION >= 20

        ; ... proceed with CAL program

)
```

See Also:
CAL Function Reference

(sendMIDI)

(sendMIDI <nPort> <nChan> <kind> <data1> [<data2>])

This function lets you transmit a MIDI event.

nPort is 0..15, or -1 for "all ports".

nChan is 0..15, or -1 for "all channels".

kind is one of the constants like NOTE, PATCH, and so on.

nData1 is 0..127.

nData2 is 0..127, and is ignored unless *kind* is NOTE, CONTROL, or KEYAFT.

See Also:
CAL Function Reference

(delay)

(delay <dwMilliseconds)

dwMilliseconds is the number of milliseconds (1000th of a second) that **delay** will wait before returning.

See Also:
CAL Function Reference

(DLL)

```
(DLL <szLibName> <szProcedureName> [ <arguments> ])
```

This powerful feature lets you make calls to functions in Windows DLLs (Dynamic Link Libaries).

The return value is of type dword, but may be assigned to any other type.

CAL string literals may be passed for LPCSTR arguments.

WARNING: Use of this function requires great care.   If you call a DLL function with the wrong number of arguments, you will corrupt the stack and subsequent operation of CAL and Cakewalk will be unpredictable.   There is no error-checking or other protection for this.

Example:

```
(do
       ; Call the Windows USER library function MessageBeep( UINT uBeep ).
       (DLL "USER" "MessageBeep" 0)

       ; Call the Windows USER library function
       ;    MessageBox( HWND hWnd, LPCSTR lpcszTest, LPCSTR lpcsz Caption, UINT
uFlags )
       ; with a NULL HWND argument and 0x40 == 64 decimal for MB_OK flag.
       (DLL "USER" "MessageBox" 0 "This is a message" "Caption" 64)
)
```

See Also:

CAL Function Reference