

The KMid Handbook

Antonio Larrosa Jiménez



The KMid Handbook

Contents

1	Introduction	1
1.1	KMid's features	1
2	General usage	3
2.1	Opening songs	3
2.2	Playing songs	3
2.3	Displaying lyrics	4
3	Collections	5
3.1	Creating a collection	5
3.2	The Temporary Collection	5
3.3	Adding songs to a collection	6
3.4	Removing songs from collections	6
3.5	Playing in order or at random	6
3.6	Selecting a song from a collection	7
3.7	Deleting a collection	7
4	Midi Maps	8
4.1	What is a midi map ?	8
4.2	Do I need a midi map ?	8
4.3	Creating a midi map	9
4.3.1	The PATCHMAP section	10
4.3.2	The KEYMAP section	10
4.3.3	The CHANNELMAP section	11
4.3.4	The OPTIONS section	11
4.4	Using midi maps	12

The KMid Handbook

5	Advanced features	13
5.1	The Channel View	13
5.1.1	Changing instruments	13
5.1.2	Changing the look mode	13
5.2	The Tempo LCD	14
6	Key bindings	15
7	Frequently Asked Questions (FAQ)	16
8	Final notes	18
8.1	Some tips and tricks	18
8.2	Hall of KMid's friends	19
8.3	About the author	19
8.4	Copyrights and License	20
A	Installation	21
A.1	How to obtain KMid	21
A.2	Requirements	21
A.3	Compilation and Installation	21

Abstract

KMid is a midi/karaoke multimedia player

Chapter 1

Introduction

KMid is KDE's midi and karaoke multimedia player. It features some features not found in any other UNIX® midi player, such as realtime graphics and karaoke text highlighting among others.

KMid has been reported to run on Linux® and FreeBSD operating systems. It uses the OSS sound driver, so it should run on every system where KDE and OSS compile. KMid also supports the Linux® Ultrasound Project Driver , which is required to get sound in GUS cards. I plan to support the ALSA driver as soon as it supports a sequencer device.

KMid shows the lyrics in the screen changing its color at the same time the music is playing, so it is very easy to follow the tune of the songs.

Hope you find KMid as fun to use as I found developing it.

Antonio Larrosa Jiménez larrosa@kde.org

1.1 KMid's features

These are some of KMid's main features:

- It has a very *friendly user interface* to display karaoke text with *realtime highlighting* of lyrics.
- It features a graphical view of what is being played on each midi channel, by highlighting the keys pressed in (virtual) keyboards.
- The most powerful *Midi Mapper* that you will ever find in any operating system.
- *Drag & drop* so you can drop in KDE any midi file from a Konqueror window.
- You can *change the tempo* of songs to play them slower or faster at your wish.

The KMid Handbook

- It shows lights to follow the rhythm of the song.
- *Customizable fonts* for karaoke text to be displayed.
- Supports the two standards to introduce lyrics in midi files, that is, lyrics or text events (and guess which one a song uses automatically).
- Session Management. If a song is playing while you logout from KDE, the next time you login, the same song will start playing.
- *Adjustable volume* in realtime.
- It can play broken midi files which make other players core dump!
- It can open *gzipped midi/karaoke files* just as any other file.
- Consumes approximately *0.1%* of my CPU (depends on the complexity of the song).
- Supports external midi synths, AWE, FM and GUS cards (for the latter you need the LUP driver and gusd installed).
- Runs on Linux® and FreeBSD (maybe also other unices ...).

Chapter 2

General usage

2.1 Opening songs

You can open a song several different ways.

First, you can select `Open...` from the File menu, then you are presented with a standard open dialog, with which you can select the song you wish to open.

You can drag a file from a Konqueror window and drop it in the KMid window. You can also Drag & Drop multiple songs at the same time.

If you specify a song in the command line when running KMid, it will also be opened.

And the final way is by selecting the song from the list of songs of the active collection.

2.2 Playing songs

To play a song, first open it, and then press on the Play button of the toolbar, choose the Play entry of the Song menu, or just press the **Space** key.

Note that when you open a file using Drag & Drop, KMid will start playing it automatically (if you drop more than one file, they will be added to a collection and they will be played sequentially).

Once KMid is playing a song, you can move the time slider, by pressing with the middle mouse button mouse button, to go to a specified position.

If a song is playing too fast or too slow for you, you can press on the arrows at both sides of the tempo LCD and make it play faster or slower. To get back to the default tempo, just do a double click on the tempo LCD.

The **Space** key is used for two things, when music is playing, and you press the **Space** key, it will act as when you press on the pause button or the Pause entry of the Song menu, that is, it will pause music. If you press the **Space** key when no music is being played, KMid will play it.

2.3 Displaying lyrics

There are two methods to store lyrics in a song, by using Text events or Lyrics events, some songs use the first, some the second, some use both of them, and some don't include lyrics :-)

KMid lets you choose which events to display, and even better, it has an option to automatically select the type of events that a song uses, so that you don't have to change the type manually. That way, if you activate the Automatic Text Chooser entry of the Settings menu, the karaoke text will be automatically selected, but you can still change them if you prefer to see the other type.

To select which type to see, you can use the appropriate entries in the Settings menu, or just press the **1** and **2** keys of your keyboard to see the Text events or Lyrics events respectively.

Chapter 3

Collections

A collection is a list of midi files that you put in a set, and which are played one after another. This section will help you to use them, and will give you some useful tips to make a good use of them.

3.1 Creating a collection

To create a collection, first open the Collections Manager dialog, by selecting the Organize ... entry of the Collections menu . Then click on the New button, and enter the name you want the collection to have.

You can also copy a complete collection by selecting it and then pressing the Copy button, which will ask you for the name of the new collection that will have initially the same songs as the selected collection.

Once you have more than one collection, you can change the active collection from the Collections Manager, by selecting it.

3.2 The Temporary Collection

The Temporary Collection is a collection that is used to hold songs you want to play but that you don't want to add to any collection.

This collection is *not saved* on exit of the application, so keep it in mind when you add lots of songs to it.

Keep on reading this section for a better understanding of the Temporary Collection.

3.3 Adding songs to a collection

How to use AutoAdd to a collection There are some different ways to add a song to a collection.

First of all in each method, you must have selected the collection you want to add songs to in the Collections Manager. Then you can press on the Add button to add a song, there will appear an open file dialog so that you can choose which song to add.

The other methods to add a song depend on the state of the AutoAdd to Collection option.

If AutoAdd to Collection is enabled, when you open a song (using File → Open... or Drag & Drop) it (they) will be added to the active collection without user intervention.

If AutoAdd to Collection is not enabled, when you open a song the Temporary Collection will be activated and cleared, and the opened songs will be added to it.

3.4 Removing songs from collections

To delete a song from a collection, just open the Collection Manager, select the appropriate collection, and the song you wish to delete, and then click on the Remove button.

3.5 Playing in order or at random

You can select the order in which songs will be played. When you select In order mode from the Play Order submenu of the Collections menu, songs will be played in the same order in which they were added to the collection.

When you select Shuffle mode, KMid will generate a random variable with a discrete uniform distribution to really play randomly the songs in the collection. It will give values to that random variable while generating the list in which order the songs will be played (you surely want to play random songs, but don't want to play twice the same song, and you want to play the last played song when you press on the Previous Song button, don't you ? :-).

The random list in which the collection will be played will be regenerated each time you add or remove a file in the active collection, and when you press on the Shuffle mode entry of the menu.

3.6 Selecting a song from a collection

You can select a song to play in the `Collection Manager`, or by using the combo box over the karaoke text.

You can also change to the next song by using the `Next Song` entry of the `Song` menu, the `Next Song` button of the toolbar, or pressing the `right arrow` key.

To change to the previous song, use the `Previous Song` entry of the `Song` menu, the `Previous Song` button of the toolbar, or press the **left arrow** key of your keyboard.

3.7 Deleting a collection

To delete a collection, simply open the `Collection Manager`, select the collection you want to delete, and click on `Delete`. Easy, isn't?

Please keep in mind that you cannot delete the `Temporary Collection`, but it doesn't matter as it is not saved when you quit KMid.

Chapter 4

Midi Maps

4.1 What is a midi map ?

A Midi Map is something that maps midi events in other midi events.

This is totally needed if a synthesizer doesn't understand the standard events (that is, if a synthesizer is not General Midi compliant), in this case, a midi map will translate General Midi events in the events that synthesizer understands.

For example you can make a midi map that converts all the Change patch to Bright Piano events, to Change patch to Trumpet events, and so when a song tries to play a piano, it will play a trumpet instead.

This may sound odd, (why playing a trumpet when the song is made to play a piano?), but it is very useful. The GM standard specifies that when a midi keyboard receives an event to change patch to 0, it will change the current patch to Grand Piano, but older synthesizer will change for example to a Electric Guitar when it receives a 0. This old keyboard, needed to receive a 3 (for example) to change to a Piano. And here comes the midi map in action, changing all change patch to 0 to change patch to 3 and thus really playing the correct instrument when it has to.

4.2 Do I need a midi map ?

In short, if you don't have an external synth, *no!*

If you only have a sound card, midi maps are not needed because all the sound cards are GM compliant (this include AWE cards, GUS cards, FM devices and so on).

If you are playing music through an external synthesizer, and it is not GM compliant, you will need to make a midi map for your midi keyboard . Although you will perhaps be a whole afternoon doing your map file and trying

different values for all the options, you will be fully rewarded when you finish it, because then you will find all the hidden possibilities of your keyboard. For example, I have a low-cost Yamaha PSS-790, which is not GM compatible, and doesn't have as many instruments as a GM synthesizer, but with KMid's midi mapper, it sounds even better than many soundcards (including AWE :-)), due to the sound quality found in external synths (even on non GM compliant ones).

4.3 Creating a midi map

There isn't any program to generate midi maps, so you will have to edit a file by hand (using your favorite text editor).

A Midi map is a text file that keeps all the needed translations there will be made when playing music.

It consists of four sections: PATCHMAP, KEYMAP, CHANNELMAP and OPTIONS.

Each section must appear only once, except the KEYMAP section that can appear as many times as needed, provided that each appearance use a different TextID (continue reading for details).

The general structure of a map file is:

```
DEFINE PATCHMAP
...
END

DEFINE KEYMAP "Name of Keymap"
...
END

DEFINE KEYMAP "Another Keymap"
...
END

DEFINE CHANNELMAP
...
END

OPTIONS
...
END
```

You can see that the DEFINE word is used to specify which section is going to be started (except for OPTIONS), and END is put at the end of each section.

You can put comments by starting the line with a # character.

Please, don't forget to send me your map file by email, so that future releases of KMid will include support for more non General Midi compliant keyboards.

4.3.1 The PATCHMAP section

This section is used to specify how patches are going to be mapped, from GM to your keyboard specs. The general usage is:

```
(Name of GM Patch name)=(N) [AllKeysTo M]
```

Where N is the number that your keyboard needs to receive to change the patch to the same that the GM standard does .

Please note that the left side of the equal sign is ignored, so GM patches are supposed to be in order (from 0 to 127) , and so you are not allowed to change the order of the lines nor to omit any of the 128 instruments.

The optional `AllKeysTo M` is used to map all notes that use that instrument to the M key. For example, suppose that your midi keyboard doesn't have a Gun Shot sound (GM patch 127) so you want to map it to a percussion drum (i.e. key 60), which sounds similar to a gun shot, then you can put in the 127th line of the `PATCHMAP` section:

```
Gunshot =100 AllKeysTo 60
```

So when a midi file tries to play a note with the patch 127 (gun shot), it will be mapped to the patch 100 (your keyboard's percussion patch) and play the note 60 (independently of the key that was going to be played).

Please note that when I use the expression 'Percussion patch', I mean the patch in which each key plays a different drum, cymbal, tom, maracas and so on, and not to a possible sound which some keyboards have and which plays a different tone of the same drum with each key.

4.3.2 The KEYMAP section

The `KEYMAP` section is used to specify how keys are going to be mapped, within a given channel or instrument. The usage is:

```
DEFINE KEYMAP "Name of Keymap"  
C 0 =0  
C#0 =1  
D 0 =2  
...  
END
```

As with the `PATCHMAP` section, it is very important the order of the lines, and that they are all there (the 128 keys).

As you can define multiple keymaps for different channels and instruments, you must give a different name to each one in the first line.

Keymaps are mainly used to map keys in the percussion channel. Have a look at the distributed maps to see some examples.

4.3.3 The CHANNELMAP section

This section can be used to map some channels to different ones. For example, if you want to swap the first and second channels, you can easily do it within the CHANNELMAP section.

However it is more useful for keyboards that need the percussion channel to be in a given channel (the GM standard use the channel 10, others use the channel 16 and others use channel 9).

Note that midi devices use 16 channels, so the CHANNELMAP section, has 16 lines, from 0 to 15 , as this one:

```
(N) = (M) [Keymap "Name"] [ForcePatch x]
```

Where `N` is the channel which is mapped to the `M` channel. If the `Keymap` option is used, the `Keymap` with name `Name` will be used in this channel (this `Keymap` should be defined earlier in the map file !). If the `ForcePatch` option is used, all events that try to change the patch which is used in this channel will be ignored, and patch `x` will be used instead.

The `ForcePatch` option may be useful for example to always use the percussion patch on the percussion channel.

4.3.4 The OPTIONS section

The OPTIONS section has some general options that can be very useful:

```
OPTIONS  
PitchBenderRatio = r  
MapExpressionToVolumeEvents  
END
```

You can specify both options, only one, or none of them.

The `PitchBenderRatio` `r` value, has the ratio by which pitch bender events will be multiplied. That is, when a midi file tries to send a pitch bender event with a `n` value, the real value that will be sent is $n * (r/4096)$ (the 4096 value is for not having to put decimal points in the map file).

This is used because the GM standard says that when a midi keyboard receives a Pitch Bender event with a 4096 data value, it should bend the note to a higher one , but some midi keyboards try to bend the initial note by two or more higher notes (even an octave higher!) when they receive a 4096. This can be easily fixed by trying different values so that instead of sending a 4096, KMid sends the appropriate value.

When the `MapExpressionToVolumeEvents` option is set in the map file, and a midi file try to send an expression event, KMid will send a volume event which is understood by more non-GM keyboards, and which has a similar effect. There are many midi files which use expression events to fade out at the end of a song, so if you feel that music should be heard softer and softer, you

can turn on this option and see if this is what you need, because your midi synthesizer could be ignoring the expression events because it doesn't understand them.

4.4 Using midi maps

To use a midi map, simply open the Midi Setup dialog by selecting the Midi Setup ... entry of the Settings menu.

Then click on Browse ..., select the map file within the open file dialog and enjoy the music ! :-)

Chapter 5

Advanced features

5.1 The Channel View

The Channel view is a window where you are shown a heap of keyboards (one for each midi channel). In these keyboards, the notes that are being played with each instrument are highlighted so that you can see what is each instrument playing.

5.1.1 Changing instruments

You can use the Channel View to change the instrument that each channel is playing. In each channel there is a combo box where you can select it. Once you change it, the green button next to it will change to red to indicate that this is not the default instrument.

If you want to set again the default instrument, click on the red button, and it will be automatically set.

5.1.2 Changing the look mode

The Channel View has two different ways (for now) to display the played notes, you can select them from the Channel View Options... item in the Settings menu.

You can choose between a mode in which played keys are pressed, as if it were a normal piano (3D look), or a mode in which keys are also filled with red color, so that pressed keys are easily recognized (3D - filled). If you play the piano, or any other music instrument, you can use this view to learn to play a song by yourself. I've used this technique and it (along with a tempo reduction) is great to learn new compositions.

5.2 The Tempo LCD

This shows the tempo in which a song is played, that is, the velocity of the song. The higher this number is, the faster the song will play.

You can also change the tempo of the song, so if a song plays too fast for you to follow the lyrics, you can make it play slower. To change the tempo, you can use the arrows that appear at each sides of the LCD.

Once you have changed the tempo, you can get back the default one by doing a double click on the LCD.

Chapter 6

Key bindings

Key	Action
Space	Play the loaded song, if it isn't playing, or pause it, if it's already playing.
Backspace	Stop playing
Right Arrow	Next song in current collection
Left Arrow	Previous song in current collection
Up Arrow	Scroll lyrics one line up
Down Arrow	Scroll lyrics one line down
Page Up	Scroll lyrics one page up
Page Down	Scroll lyrics one page down
1	Display text events
2	Display lyric events
Ctrl+O	Open a song
Ctrl+Q	Quit KMid
F1	Open this document

Chapter 7

Frequently Asked Questions (FAQ)

1. *What is exactly a midi file?*

A Midi file is a file that contains the information on how to play a song, that is, it contains simply the notes, the rhythm, velocity, etc. This implies that the same midi file, when played in two different devices, can produce very different results, as well as a given staff can be played very differently by two different musicians.

2. *I can get better sound with a mp3/wav player, why should I use KMid?*

Well, I cannot force anyone to use KMid, but a typical midi file occupies 50 Kb. while a typical mp3 file occupies 4 Mb. (and that is a 1:80 compression ratio :-). And with a good synthesizer device, you can get a comparable sound quality. Even more, with a midi file, you can change individual instruments, change the velocity of a song, etc. so you have more overall control.

3. *I cannot get my AWE card to work with KMid, what can I do?*

This can happen when you get a binary distribution (rpm, deb) of KMid. It happens because KMid was compiled without awe support. If it doesn't work, then you must download a source code distribution (for example, from [KMid's homepage](#))

4. *I want to add a whole folder to a collection, but having to add the midi files one by one is not funny.*

I agree, that's why KMid supports Drag & Drop. Just open, in Konqueror, the folder you want to add, select all the files, drag them and drop them in KMid. Be sure to set the AutoAdd to Collection option before, so that the files will be added to the current collection. If you don't do this, files will be added to the Temporary Collection.

The KMid Handbook

5. *I cannot follow the lyrics, it's playing too fast!*

You can press the left arrow of the tempo LCD to make it play slower. Remember that you can do a double click on the LCD to get the default tempo.

Chapter 8

Final notes

8.1 Some tips and tricks

I will include some tips so that you can take fully advantage from all the features of KMid:

Opening files I always keep a KDE desktop with a Konqueror window in my root midi folder, and KMid in this desktop (if playing a midi file) or sticky (if playing a karaoke file :-)). This way, when the active collection finishes, or I want to play some file, I just go to the Konqueror window, select the desired files and Drag & Drop to the KMid's window.

Suppose that you want to play some midi files, but don't want to add them to any collection, well, just turn off the `AutoAdd to Collection` option in the `Collections` menu, and open the files, they will be added to the `Temporary Collection`.

Another method to create a new Collection Suppose that you have midi files `A.mid`, `B.mid` and `C.mid`. At first you only want to play the first midi file, so you unselect `AutoAdd` and open `A.mid`. You get then a `Temporary Collection` with only one midi file.

Then you decide to play also B and C, and make a collection with all them, what do you do?

Easy, select `AutoAdd` and open `B.mid` and `C.mid` (by any of the multiple methods), they will be automatically added to the `Temporary Collection`, that will then have `A.mid`, `B.mid` and `C.mid`. At this point, you can open the `Collections Manager` dialog, select the `Temporary Collection`, and click on the `Copy` button, enter the name of the new collection, and you are done. You already have a new collection, which holds the A, B and C midi files, and that is not deleted when you close KMid.

8.2 Hall of KMid's friends

These are some folks who have sent me midi files or a postcard, thanks to everyone! Hearing those songs and watching those postcards will keep me programming more and more on KMid.

- Ola Sigurdson - Taking Care of Business (Bachman Turner Overdrive)
- EG Lim - A really very nice postcard from Penang.
- Guenther Starnberger - Amadeus (Falco) and Schrei Nach Liebe (Die Aerzte)
- Leandro Terrés - All That She Wants and The Sign (Ace of Base)
- Nick Stoic - Two midi files

8.3 About the author

KMid has been made by Antonio Larrosa Jiménez, in Málaga (Spain). I am a student of Mathematics at the University of Málaga, currently I am doing the third course, so I don't have much free time for my hobbies, but I always try to get some :-). My hobbies include: Programming, collecting midi files, playing music and proving theorems ;-).

If you want to know where to download midi/karaoke files, you have any question, a bug to report, an idea or a feature you'd like to see in KMid or just want to make me happy, feel free to send me an email to: larrosa@kde.org or antlarr@arrakis.es

or write to me by snail-mail at:

Antonio Larrosa Jimenez
Rio Arnoya 10 5B
Malaga (Spain)

You will really make me happy if you send me a postcard from where you live, or a midi/karaoke file from a local music group of your country . Everyone who sends me a postcard or a midi file will have his/her name in the Hall of KMid's friends of this file (provided they don't oppose to this). Please contact me before sending me any midi files because I may have it already.

I'd like stress that KMid has been done completely on free time, without any monetary support from any company nor particular. So please keep in mind when you use it, that the only think that keep me working on this is getting some feedback from its users (a postcard, or just an email).

I would like to thanks the following persons their help in developing KMid:

- Paul J. Leonard PJ.Leonard@bath.ac.uk - Support for AWE cards

The KMid Handbook

- Sebestyén Zoltan szoli@digoo.inf.elte.hu- FreeBSD port and AWE testing
- Christian Esken esken@kde.org - For organizing the KDE multimedia efforts
- Stephan Kulow coolo@kde.org- Configure scripts and help with **automake** and **CVS**
- Jaroslav Kysela - Help in doing the Linux® Ultrasound Project driver support
- Takashi Iwai and Joseph H. Buehler - Fix for AWE cards pitch being too high
- Adrian Knoth - For giving me good news and many suggestions
- Kevin Street - Patch to support FreeBSD 3.0
- Thanks go also to Jose Luis Sanchez for his testing of GUS support, Ignacio Garcia for testing the AWE support, Hans Petter Bieker, Ola Sigurdson, Marc Diefenbruch, Peter Gritsch, Magnus Pfeffer, Urko Lusa, Peter-Paul Witta, Thorsten Westheider, Ulrich Cordes and everyone that sent me a patch, bug report or just an email to give me encouragement.
- And of course to all the fabulous musicians over the net that keep giving us those wonderful midi and karaoke files.

8.4 Copyrights and License

KMid is copyright Antonio Larrosa Jiménez, 1999-2001

Documentation is copyright Antonio Larrosa Jiménez 1999, 2001

This documentation is licensed under the terms of the [GNU Free Documentation License](#).

This program is licensed under the terms of the [GNU General Public License](#).

Appendix A

Installation

A.1 How to obtain KMid

KMid is part of the KDE project <http://www.kde.org/> .

KMid can be found in the kdemultimedia package on <ftp://ftp.kde.org/pub/-kde/> , the main FTP site of the KDE project.

Additionally, KMid can be found on its homepage, which is at <http://www.arrakis.es/~rlarrosa/kmid.html>. In the homepage, you can follow its development, see some information about it, some screenshots, a list of sites from where you can download more karaoke songs, etc.

A.2 Requirements

KMid requires to work:

- KDE. Well, you probably already have this :-)
- A sound card. A good soundcard and/or external synthesizer are recommended, as the sound quality depends greatly in your soundcard, it's not the same to play the music using an FM device, than using an AWE card.
If you don't have a soundcard, you can still compile KMid with `MODE_DEMO_ONLYVISUAL` defined and it will run as if you had one (but you'll get no music, of course :-).

A.3 Compilation and Installation

For detailed information on how to compile and install KDE applications see [Building KDE4 From Source](#)

The KMid Handbook

Since KDE uses **cmake** you should have no trouble compiling it. Should you run into problems please report them to the KDE mailing lists.

I've included some examples that are installed in `$dollar;KDEDIR/share/apps/kmid`

If you run into any problem, don't hesitate to contact any of the KDE mailing list, or send a report directly to me.