musicpad tutorial

back to musicpad

OK, you heard about musicpad and you want to give it a try. But then you feel completely lost with the coding part. Here is your chance: a musicpad tutorial.

The musicpad idea was born when I stumbled on jens'polymath; I wanted to develop it further to fit my needs of a quick musical notepad that could reproduce your idea very quickly, and from any internet-connected computer. After the first basic version, many ideas came to me with additional commands, resulting in the musicpad you see now. But you don't need to know every command to use musicpad. I'll walk you through them one after the other, and everything will be clearer.

Notes

.

"I have a tune idea. Quick, I write down the notes names: ' C E G F eb c# c '". Well, that's exactly how you write notes down in musicpad:

```
C E G F eb c# c
```

Each note is separated with a whitespace (or a new line, or 50 whitespaces if you want, it doesn't matter). They can be written in uppercase or lowercase, as you which - musicpad is 99% not-case-sensitive.

Flat is indicated with 'b' or '-', sharp with '+' or '#'.

"Cool, but I want to add some timing to it". The first two notes should be quarter-notes (1/4), the G should be half-note (1/2) and the last notes 1/8th notes... In musicpad you add timing to a note by adding '/' and the timing. Easy ! (you note that the default duration is 1/4)

```
C/4 E/4 G/2 F/8 eb/8 c#/8 c/8
```

Another way to put in timing is using the '=' command, which means: play the previous note twice its initial duration. You can have multiple '=' for repeating the order. Note that you don't have to put whitespace around '='.

```
C/8 = E= G === F eb c# c
```

"OK. But the final 'C' should be one octave higher". You have two ways of doing it: adding the octave to the note just before the '/', like: 'C5/8' (read: C, octave 5, 1/8th note): (you note that the default octave is 4)

```
C/4 E/4 G/2 F/8 eb/8 c#/8 c5/8
```

... or by putting in a '/' all alone before the note (meaning: ramp up one octave). The other command '\' (one octave down) also exists of course.

```
C/4 E/4 G/2 F/8 eb/8 c#/8 / c/8
```

"OK, but it's a lot to write for a few notes..." Well, musicpad has an instant short-term memory, allowing you to be more elliptic in your writing. That means, if you don't indicate a paramater, musicpad uses the same as the previous note. So let's get rid of what musicpad doesn't need:

```
C/4 E G/2 F/8 eb c# / c
```

"How do I repeat exactly the same note ?" You can use the 'x' command :

```
C/4 E G/2 F/8 eb d# / c x x x
```

"I need a pause somewhere" The pause command is 'P' , and you can optionnally indicate its length with '/..' like normal notes

```
C/4 E G/2 P/4 F/8 eb d# / c x x x
```

you can use the '-' command for a pause of the same duration as the previous note:

```
C/4 E G/2 P/4 F/8 eb d# / c x x - x
```

Note that like the '=' command, you are not forced to use whitespaces around x and -

```
C/8 -x-xxx=x-x
```

Be careful though with the '-' for flat and '-' for pause... 'C---' is interpreted as 'C flat then 2 pauses'.

Another shortcut: if you just specify a number alone, by default it's a duration (except in guitar or toms mode, but we'll see that later). For example, '12' alone means 'play the same note as before, but as 1/12'

```
C/8 P 8 P 8 8 4 8 P 8 16 16 16 16 4
```

Oh, and let's learn right now the **tempo** command to speed it all up... Tempo is 60 quarter notes per minutes by default, but you can change that easily (usually at song beginning):

```
tempo100 C/4 E G/2 P/4 F/8 eb d# / c x x - x
```

OK, you're happy, you can write one-finger songs on the piano. Er, well, that's a start, right ?

Tracks and Instruments

Good news: with Musicpad you can have... more than one note playing at the same time ! And even on different instruments ! All that thanks to **TRACKS**.

Tracks are a way to say to Musicpad: "OK, now that you've played this, start over and play another instrument on top of it !".

Musicpad skip to the next track each time it encouters the **|** sign. Let's try it right away

```
C E G | E G B
```

What happened ? Well, we played C E G with one finger, and E G B with another finger. Kewl... but the sound is choppy, you can't hear all notes ? Let's try this:

```
ch1 C E G | ch2 E G B
```

Sounds better ? Well, that's because previously we used two tracks to play on the same midi channel... not advisable, sometimes midi doesn't handle it well. With the **ch** command, we indicate that the first track should be played on channel 1 (default channel), and second track should be played on channel 2. Think of it as indicating that we should have do different pianos & piano players to make sure all notes are played.

Wanna change instruments ? Sure, let's use the **i** command followed by the instrument number (check the help file to have a list of midi instruments codes). Let's use... an drawbar organ (17) and a nylon guitar (25):

```
ch1 i17 C E G | ch2 i25 E G B
```

There, the channel indication is really mandatory... If we had not indicated this, musicpad would have sent two "change to instrument x" orders on the same midi channel, resulting in midi only taking the latest order into account - we would have heard only one intrument playing two notes, probably the guitar.

Another cool feature of tracks is that you can indicate the track number you want to jump too by specifying a number right after the **|** - knowing that the first track is always number 1 (but you can use track 0 also if you want). Let's try this:

```
ch1 i17 C E G | ch2 i25 E G B |1 F E C | A G E
```

See the **|1** in the middle ? Note that you could have obtained exactly the same result with: ch1 i17 C E G F E C | ch2 i25 E G B A G E... So what is it used for ? Well, for long songs, you might want to write the intro as a whole, then the first verse, then the chorus, etc... With "track jumping" you can do this easily. It is also useful when using macros.

However you have to remember the golden rule of tracks:

Each Track Is Independant From All Other Tracks

OK, what does that really mean ? Simply that you must not base your track timing on other tracks... and if you want synchronized tracks you have to adjust timing and duration yourself. Let's take an example:

```
ch1 i17 C E G | ch2 i25 E G B B |1 F E C | A G E
```

What happened ? We added a fourth note to the first part of the second track, but not to the first part of the first track. Musicpad played each track independantly, so the second part of the second track is played one note later than the second part of the first track. It's perfectly OK with Musicpad... If this is really the result you want to get !

Want to add some drums ? Easy. The **i** command can also change to drum instruments (SD for Snare, BD for Kick, HH for HiHat, etc... check the help file to have a list of midi drums codes). What the **iSD** command really does is change midi channel to 10 (because for midi, Drums are played on channel 10) and select the correct default note for Snare Drum. Don't try to change that by indicating your own channel, you'd be surprised of the result. Don't specify the note either -

a **x** is enough. Also, each drum instrument (Bass Drum, Snare Drum, HiHat) must have its own track. Let's take an example:

```
  iHH -x-x-x-x
| iSD --x---xx
| iBD xx-xxx--
```

Cool, but slow. How to speed it up ? Well, indicate duration using the "number alone" shortcut is one way to do it... Another way is to use the **o** command; If you add an **o** to a note, that note will NOT be played, but its parameters will be remembered for the next notes. Let's turn our rythm to 1/16th notes:

```
  iHH o/16 -x-x-x-x
| iSD o/16 --x---xx
| iBD o/16 xx-xxx--
```

OK, but how to repeat this rythm without copy/pasting code ? Well, that's the subject of the next part:

Structure

All music - especially modern music - use repetition of patterns, for rythm as well as for melody. With Musicpad you can deal with this using two commands:

The first is the ***** (star = multiply) command. For example, ***4** means: repeat 4 times. Its use is twofold:

- If you put a ***** after a command, musicpad will repeat that command - and only that command. A command is delimited by whitespaces. For example: iBD 8 *8 is the same as iBD 8 8 8 8 8 8 8 8. Note that you can have a whitespace before the *****, but not between the ***** and the number of repetitions.
- If you put a ***** after a set of commands delimited by **( )**, musicpad will repeat that set of commands. For example: iBD (8 4) *4 is the same as iBD 8 4 8 4 8 4 8 4.

So... Let's take our rythm above and loop it !

```
  iHH o/16 -x-x-x-x*4
| iSD o/16 --x---xx*4
| iBD o/16 xx-xxx--*4
```

One interesting feature of ***** is illustrated here: I don't need whitespaces for x-commands, so the ***4** command repeats the whole -x-x-x-x string... I didn't have to use **( )**...

Another very useful tool are **MACROS** with the **m$** and **$** commands. Macros are a set of commands that you ask musicpad to store in its memory, and that you can call very easily

To set a macro, use the **m$*name*(...)** syntax. You can specify whatever *name* you want for the macro - can be a letter, a digit, or a combination thereof.

To call a macro, use the **$name** command. A simple example:

```
m$1(C/16 E G) m$secondmacro(F F F) m$3rdMacr(C C C C)
$1 $secondmacro $1*3 $secondmacro $3rdMacr
```

Note that you can combine macros and repeats. Note also that macro names are case-sensitive.

A warning though: macros are neat, but tend to require more computation time. If your song relies very heavily on macros, you might end up with a "time-out error" - your browser telling you he thinks he's lost contact with the server which was too slow to answer...

Another usefull command... You can add comments to your song by using **#**. Everything after a **#** will be ignored until line end.

Now you have all elements in hand to begin to write your own tunes. But... where to begin ?

The structure of a song depend of the song itself. I'll illustrate below a musicpad code structure for a typical modern song.

Musicpad Code

Comments

```
# My cool song
```

- Name your song - in comment !

```
tempo70
```

- First the general declarations - like tempo

```
#Instruments
```

- Comment to structure the code

```
|0 iHH o/16 | iSD o/16 | iBD o/16
|5 ch1 i01 o4/16 | ch2 i34 o2/16
```

- Instruments definitions. This is where I define what the tracks will be made of, starting with track 0. I have the habit of putting drums first, in drum tab reading order, then leave some space in case I want to add another drum instrument, then the other instruments starting on track 5, with specification of the midi channel to use.
  Note also the definition of default duration and octave.

```
# Macros

m$rythm(
|0 xxxxxxxxxxxxxxxx
|  --x---x---x---x-
|  x---x---x---x--- )
```

- Then the macros. Don't forget the closing **)**...
  To be useful each macro usually

includes track information, to indicate where this is to be played. Drum patterns are a lot more readable when put on multiple lines with whitespaces to align everything. Note the repeat inside the 'piano' macros...

```
m$bass1( |6 E*16 )
m$bass2( |6 A*16 )

m$piano1( |5 (P G# G# G# A G# F# G#)*2 )
m$piano2( |5 (P G# G# G# A B / C# E \ )*2 )
```

- Then the song in itself. Repeats can of course be used. Macros can be specified in whatever order we want, it doesn't matter as each macro specifies the track to be used. The only thing is not to get lost in measure count.

```
# Song Structure

$rythm*4 ($bass1 $bass2 $piano1 $piano2)*2
```

play that song

Chords

How can you play chords in Musicpad ?

If you read carefully this tutorial, you already figured out that you can do it by adding tracks on top of tracks. If you want to be very precise in your complex chord playing, it's the only way to do it. However, if you're only looking for plain simple chords playing, Musicpad can do it for you.

Chords in Musicpad are indicated by brackets **[...]** with no whitespace inside the brackets. There are 4 different chord syntax available:

- **[#,#,#,...]** : the basic chord definition. Each figure represents a note inside the chord, counted in half-tones relatively to the actual default note. For example if the default note is C4, and you play chord [0,4,7,12]... Here is the result: oA4 [0,4,7,12] .
- **[G:#,#,#,#,#,#]** : If you add a 'g:' in front of the detailled chord, musicpad will consider it's a guitar chord, relative to guitar tuning. Of course there must be 6 notes indicated. Use a '-' for not-played string, and a figure to indicated fret number (0 or more). Here is for example a standard Amaj chord: i25 [g:-,0,2,2,2,0]
- **[Amin]** or **[min]** : standard keyboard chord. If you don't indicate the base note inside the chord, the default note will be used. Chords have variations,

which you can call using **:** and a number after the chord definition. For Keyboard chords, variations are usually :1 and :2, in some cases up to :5 - try for example [Amin], [Amin:1], [Amin:2]. Check out the chords list for Chord syntaxes.

- **[G:Amaj]** : If you add a 'g:' in front of the chord, musicpad will use a guitar chord instead of a keyboard chord. Guitar Chords also have varations indicated with ':' - let's try for example [g:Amin], [g:Amin:1], [g:Amin:2], [g:Amin:3]. Note that guitar chord base note is to be written only with 'A-G' and 'b' for flat (don't use sharp '#','+' or flat '-') (hey, don't complain, you still have 900+ guitar chords to choose from !!!). The chords list lists the syntaxes.

Want to change guitar tuning ? **tuning[E3,A3,D4,G4,B4,E5]** defines guitar tuning to be used for all guitar chords. Watch out though, if you change the tuning and play a standard guitar chord on top of it, you might end up with surprises...

You noted that the guitar chords example above are... not sounding like a guitar. That's because when you play a real guitar chord, notes are not played each at the same time - you strum the strings one after the others. Well, Musicpad can do that too, using the **strum** command ! The idea is to have a delay between chord notes, and also the possibility to define a minimum time delay that can trigger an "upstrike" strumming, where notes are played in reverse order and possibly at another level. **strum*a,b,c*** uses up to 3 parameters:

- a = delay between notes (strings) in milliseconds
- b = maximum time to trigger upstrike in milliseconds - optional parameter.
- c = upstrike volume in % - optional parameter.

For a rythm guitar try strum10,300,80 : i25 o/16 strum10,300,80 [g:Amin]*16

You can also use this for any other instrument: i25 o/2 strum100 [Amin]*4

Advanced Commands

There are some other commands we haven't talked about yet.

First the "figure only" command... Remember, I told you if your command is only one figure (like: 8), musicpad uses it as a new duration (1/8) to play the same as previous note. Well, there are two other modes that you can activate track by track to change that:

- The Guitar Mode, activated by command **guiton**. In that mode, a number indicates a temporary transpose in half tones compared to the previously defined note. And that is very useful for a guitar track ! Each track in guitar mode can represent a string of the guitar, and you can import almost directly guitar tabs ! Let's have a try:

```
tempo100

|0 ch1 i25 guiton | ch1 guiton | ch1 guiton | ch1 guiton
| ch1 guiton |

(
|0 oE5/16
------5=7=====7=8=====8=2=====2=0=========0=====-------------
|  oB4/16
----5==---5====---5====---3====---1===1=====1===0=1=1========
|  oG4/16
--5====-----5==-----5==-----2==-----2=========2=0=2=2========
|  oD4/16
7=======6=======5=======4=======3===============-------------
|  oA3/16
------------------------------------------------2=0=0===0===8
)*2
```

play that song

Pretty neat, huh ? And you can even still read the guitar tab ! See the clever use of **=** and **-** to hold notes or simulate chord change... The drawback is of course that to indicate note duration I can only use the **o/** command (specifying a note but not playing is), which is very useful to indicate string tuning and default duration.

Note that I could have spared myself the **guiton** order on each track by using a **globalguitmode** command at song start, which activates guitar mode for all tracks by default. And for those tracks which should not be in guitar mode by default ?... Well, there is also the **guitoff** command of course...

For clarification, note that the Guitar Mode is totally independant from the Guitar Chords and the **tuning** command.

- Another mode is the Tom Mode, activated by the **tomson** command (and cancelled by the **tomsoff** command). It's used very specifically for toms in rythm tracks; instead of using a different track for each tom you use (... up to 6 different toms, so 6 different toms tracks... argh...), just use one track in tom mode; a figure alone indicate the tom number to be used (from 1 for highest-pitch tom to 6 for lowest-pitch tom). Let's try it out:

  ```
  tomson o/16 4 3 5 1 1 5 6 6 4 5 4 2 1 1 1 6
  ```

  Note that the **tomson** command automatically selects midi channel 10 for drum track. Same remark as for guitar mode, duration should initially by indicated by a **o** command.

In musicpad, there are also some other commands to play notes or influence the tone of the note played:

- The **t+#** or **t-#** command is use for permanent transpose, in half-tones; further notes will be played with the tone modification specified. For example, **t+2** means that all further notes will be played one whole tone higher than their specification. Let's try it with a macro:

  ```
  o4/16 m$1(C D E)
  $1 t+2 $1 t-2 $1 t+4 $1 t-4 $1 t+6 $1 t-6 $1 t+8 $1 t-8
  $1 t0 $1
  ```

  To return to normal (no transpose), use the **t0** (t-zero) command.

- The **N** command use used to specify note tone with a number. It has 3 different syntaxes :
  - **N##** indicates a note to be played by its midi number. 'C5' for

example is midi note 60, so N60 will play a C5.
  - **N+#** or **N-#** orders musicpad to play a new note with # half-tones more (or less for N-) compared to previous note play. It changes the reference note to that new note of course.
  - **NT+#** or **NT-#** has the same effect as the N+/N- commands, except that it DOESN'T change the reference note to that new note.

OK, let's have a few tries:

```
n50/16 n+1 n+1 n+2 n-1 n-1 n-2
```

```
n50/16 nt+1 nt+1 nt+2 nt-1 nt-1 nt-2
```

There is still a bunch of commands left to explain:

R*a*/*b* : next notes duration should be multiplied by a/b (allows playing triplets with 4/3 or 2/3 etc. Also used for polyrythms)
R1 : back to normal duration
mrnd$*xx*(...) : define random macro $xx; inside the macro, list all possible macros. Everytime you process the song, one macro will be randomly chosen from the list to remplace the random macro. (NB: mrnd$ lowercase only)
u*xx* : (duty) note is played over xx% of normal length - 10 for staccato, 98 by default, 100 max
duty*xx* : define default duty of notes (by default 98)
v*xx* : set velocity to xx (max 127)
velocity*xx* : define default velocity of notes (by default 100)
stress*xx* : stressed notes will be xx% stronger
soft*xx* : soft notes will be xx% softer
pitch+*x* or **pitch-*x*** : change pitch of +/- x% (usually 100% = 2 half-tones).
pitch0 to center pitch.
ctrl*a,b* : send midi control parameters a,b
sysex*a,b,c,...* : send midi sysex values a,b,c,...
dEbUg (case sensitive command!) : debug mode, will print out processings - allow you to debug your song... also **dEbUgMaX** is an even more insanely complete debug mode...
loose*w,q* : add "human" feel to playing; notes hit will vary of +/- w milliseconds around exact time. The q parameters defines how centered notes are: with a q of 1 repartition is random, with a higher q notes are more centered on exact time (q=10 is almost always exact time). A standard value example would be loose20,2
globaloose*w,q* : define default loose parameters for all tracks (see loose above)
velvar*w,q* : add another "human" feel to playing; notes velocity will vary of +/- w% around standard velocity. The q parameters defines how centered velocity is: with a q of 1 repartition is random, with a higher q notes are more centered on exact velocity (q=10 is almost always exact velocity). A standard value example would be velvar10,2
globalvelvar*w,q* : define default velvar parameters for all tracks (see velvar above)

... Stay tuned & watch this space, I'll come to it.

musicpad v2.2 © loic prot 2004. largely inspired by polymath.cgi, © jens johansson 2001