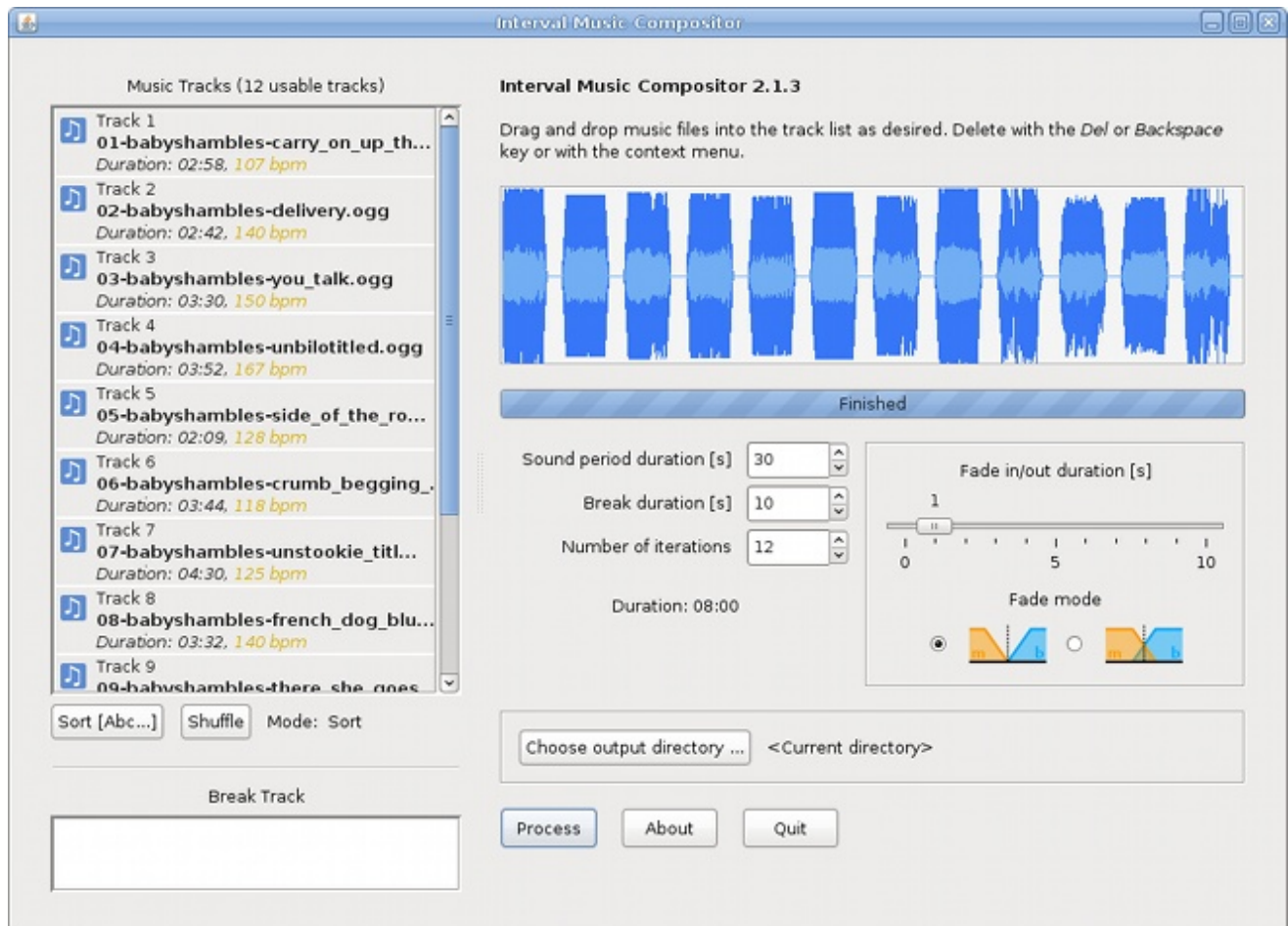


Interval Music Compositor

Last update: 04. March 2011, 16:08

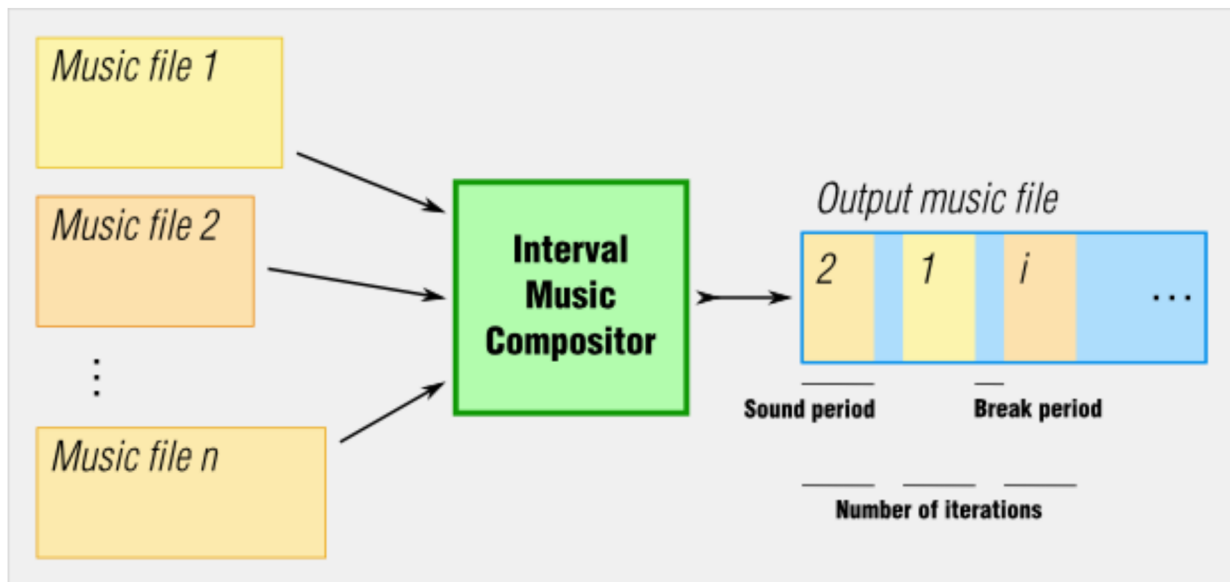
Keywords: *interval training, circuit training, physical education, PE, sports, music, exercise music, break, iteration, workout, automated generation, compilation, bpm*



Interval Music Compositor is a small, multi-platform software which generates a random interval sequence out of an arbitrary number of music files. One can choose the duration of the music parts and the intermediate breaks as well as the number of iterations. The output is a single, large audio file to be processed further or burnt to cd.

The german-language professional journal [mobile](#), a periodic publication by the swiss [Federal Office of Sports](#) about physical education, published a short review on the **Interval Music Compositor** in its 1/2010 issue: [mobile 1/2010 \(p. 39\); "Für Sie gelesen - Intervallmusik einfach zusammenstellen"](#)

The main purpose is the generation of music compilations for the use with circuit or interval training in physical education or sports in general. Certainly it can be used for any purpose where a compilation is desired but the exact position of the extracts does not matter. For example one can create a representative audio overview over a music CD in seconds.



Features

Last update: 05. March 2011, 00:33

The retorte.ch **Interval Music Compositor** has the following properties.

What it does

- Reads audio files (tracks) in current directory
- Via drag'n'drop more tracks can be added to the list
- The track list can be sorted or shuffled
- The speed of a track (bpm) is loaded from the metadata or calculated (identifies speeds between 60 and 185 bpm). It can also be entered manually or determined by tapping. The value can be stored back into the original file.
- Ignores the first and the last 5 seconds of a track
- Discards tracks which are too short
- Takes a random extract of the tracks
- Normalizes the volume of the extract
- Composes them together to one big output file (compilation)
- If there are not enough tracks to fill the compilation, it starts from anew
- Stores the playlist data in a textfile
- Displays the audio envelope of the compilation for test purposes

What it needs to be run

- Lots of diskspace (the temporary files and the output file need 10MB per minute music): 1GB
- Lots of main memory
- Installed Java Runtime Environment (Version 6)

Supported input file formats

- WAV
- MP3
- OGG
- FLAC

Output file format

The resulting sound file is a 16bit, 44.1KHz, stereo WAV file, or, if a MP3 converter is present, a corresponding 256kbit MP3 file.

Instructions

Last update: 03. March 2011, 17:16

The **Interval Music Compositor** comes as a single executable file. No installation is needed therefore. However, it is based on *Java* technology, thus a *Java Runtime Environment* (JRE) must be present to run it.

If you don't have one installed, you are prompted to do so on the first start of the software on the Microsoft Windows version. Users of Linux and Mac OS X operating systems are expected to have one installed already.

Usage

1. Start the **Interval Music Compositor** by double-clicking it
2. (If necessary, follow the instructions for installing the *Java Runtime Environment*)
3. Drag all the audio files you'd like to have processed into the music track list
4. If you want a track being played during breaks, drag one into the break track list
5. Wait until all tracks have been imported properly. This is indicated by a blue icon in front of each track
6. If you're not sure about the speed (bpm) of a track, determine it manually by right-clicking on the track and choosing 'Change BPM...'
7. Determine the order of the track by dragging them around or pressing the 'sort' or 'shuffle' buttons.
8. Choose a target directory for the audio compilation by clicking the according button and choosing one in the dialog
9. Enter the desired values for the the length of the sound samples (in seconds, *i.e. 30 seconds*), the fade-in/out mode and period (*i.e. 1 second*), the length of the breaks (in seconds, *i.e. 10 seconds*) and the number of sound samples you'd like to have in the end (*i.e. 12 samples*)
10. Press the 'Process' button
11. Wait some time ...
12. If it says 'Finished' in the progress bar, the output sound file (in this example '30_10_12_imc_out.wav'i (resp. '30_10_12_imc_out.mp3'), the numbers match the entered values) and the according tracklist ('30_10_12_playlist.txt') have been generated (this is, unless there were some serious problems). You can check the audio envelope to determine if the output file is ok.

Legend

These are the meanings of the icons displayed in front of imported tracks:



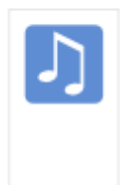
EN: The file is being imported or processed.

DE: Die Datei wird gerade importiert oder sonstwie verarbeitet.



EN: The file is too short for the this settings. If you start the process now, it is omitted.

DE: Die Datei ist für die Einstellungen zu



EN: The file was successfully imported and is now ready to process.

DE: Die Datei wurde erfolgreich importiert und steht nun zur Verwendung bereit.



kurz. Sie wird so nicht verwendet.

EN: There was a problem with the encoding of the file. It can't be used by the software.

DE: Die Kodierung wird nicht verstanden. Die Datei kann nicht verwendet werden.

These are the meanings of the colors of the displayed speed (bpm) measurements:

- 120 bpm** *EN: The value was read from the original file's meta data and is considered reliable.*
DE: Der Wert wurde aus den Metadaten der Originaldatei gelesen und gilt als zuverlässig.
- 120 bpm** *EN: The value was calculated from an extract and is of unconfirmed reliability.*
DE: Der Wert wurde aus einem Ausschnitt berechnet. Die Zuverlässigkeit ist unbestätigt.
- 120 bpm** *EN: The value was entered manually and is considered reliable.*
DE: Der Wert wurde manuell eingegeben und gilt als zuverlässig.

MP3 Generation

Since the encoding of MP3 files is covered by software patents, it is easier to skip it in the software and let it add through the user himself.

To add the ability to directly encode MP3 output files, you need to add the converting tool [Lame](#) to the Interval Music Compositor directory. A small guide for it follows the [download](#) instructions.

Download

Last update: 11. March 2011, 19:00

The current version 2.1.3 was issued on 11th March 2011.
([Changelog](#))

The software is provided free of charge. Please give me [feedback](#) if you encounter errors or strange behavior.

I hereby declare: The software is free of malicious and snoop parts in any way.

The following variants are available for download:

[Linux](#)

[Mac OS X \(>= 10.5\)](#)

[Microsoft Windows \(XP/Vista/7\)](#)

[Legacy versions for older systems](#)

[Source code](#)

Linux version



Version 2.1.3: [IntervalMusicCompositor-2.1.3-lin.tar.gz](#) (1.3 MB)

The archive consists of two files; a start script 'IntervalMusicCompositor' and the Java file. Start the program by either double clicking on the start script or calling it from the shell:

```
user@localhost $ ./IntervalMusicCompositor &
```

If you are one of the rare species without JRE installed, call this to install one (on Debian or Ubuntu):

```
user@localhost $ sudo aptitude install sun-java6-bin  
user@localhost $ sudo update-java-alternatives -s java-6-sun
```

How to add MP3 support?

Under Linux, the 'lame' binary has to be either existent in the current Interval Music Compositor directory or in the system directory for binaries - '/usr/bin'. Install it by calling:

```
user@localhost $ sudo aptitude install lame
```

Mac OS X (>= 10.5) version





Version 2.1.3: [IntervalMusicCompositor-2.1.3-mac.tgz](#) (1.4 MB)

After the download you can unzip it by double clicking on it. You can then drag the program into a folder of your choice.

How to add MP3 support?

Get an executable version of Lame and either install the package in your system or place the executable binary 'lame' in the same directory as the Interval Music Compositor lies. You can obtain such installable version i.e. here: www.thalicttrum.com (LAME Audio Encoder).

Microsoft Windows (XP, Vista, 7) version



Version 2.1.3: [IntervalMusicCompositor-2.1.3-win.zip](#) (1.3 MB)

This is just a simple executable in a ZIP file which can be unpacked and then placed in the folder of your choice and executed directly.

How to add MP3 support?

Get an executable version of Lame and place the executable binary 'lame.exe' in the same directory as the Interval Music Compositor lies. You can obtain such version i.e. here: www.rarewares.org (Lame bundle).

Legacy version

For users with older systems, where there still runs Java 5 (version 1.5) i prepared a stripped down older version of the Interval Music Compositor which is supposed to run on these:

Linux: [IntervalMusicCompositor-0.0.9legacy.tar.gz](#)

Mac OS X: [IntervalMusicCompositor-0.0.9legacy.dmg](#)

Microsoft Windows: [IntervalMusicCompositor-0.0.9legacy.zip](#)

Source code

The source code is being beautified at the moment (comments ...) and will be presented for download at this place soon. If you want to have it right now, please send me a mail. I will send it.

Feedback / Support

Last update: 09. March 2011, 10:55

Feedback

Write me if you've found a bug, have a suggestion for improvements, need help or just want to drop some lines. If you want an answer, state your e-mail address.

Name*

E-Mail

Comment*

Billie the away

Type the two words:



Known problems

- If a filename has some nasty characters in it, the file is ignored.
- On Mac OS X computers it may be that the wrong Java version is selected. The program will then only run for a second and close again immediately after being started. Start the 'Java Settings' (Applications -> Utilities) and move Java Version 6 to the top of the list to fix it. If there is no Version 6 of Java you probably have to update your operating system to obtain it.
- MP3 files in exotic encoding (sampling frequency lower than 44.1 kHz) are in some cases played faster in the output file. Such only available in mono are reduced to one channel.
- On Mac OS X systems it currently seems not possible to drag and drop more than one files at the same time into the music track list.
- On Linux, the pre-listen function only plays music if there is no other audio source active on the system.

Et cetera

Last update: 05. March 2011, 00:36

Legal etc.

The software is distributed under the GPL.

License: [GNU Public License](#)

You're using it at your own risk! :)

Donate

Support the ongoing developement of the project.

Will follow ...

Credits

This project uses the following libraries:

Tritonus

For OGG Vorbis decoding and signal processing.

<http://www.tritonius.org/>

License: [GNU Lesser General Public License](#)

JCraft Jorbis

For OGG Vorbis decoding.

<http://www.jcraft.com/jorbis/>

Lizenz: [GNU Lesser General Public License](#)

JavaZoom JLayer and MP3SPI

For MP3 decoding.

<http://www.javazoom.net/javalayer/javalayer.html>

<http://www.javazoom.net/mp3spi/mp3spi.html>

License: [GNU Lesser General Public License](#)

JFLAC

For FLAC decoding.

<http://jflac.sourceforge.net/>

License:

Copyright (C) 2000,2001,2002,2003,2004 Josh Coalson

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Parts of Dr. Simon Dixons BeatRoot

For determination of the BPM values.

<http://www.eecs.qmul.ac.uk/~simond/beatroot/>

License: [GNU Public License](#)

JAudioTagger

For reading and writing of the TBPM tags in in MP3, FLAC and OGG files.

<http://www.jthink.net/jaudiotagger/>

License: [GNU Lesser General Public License](#)