

Maestro Frankenstein 0.4 beta

The ReadMe

Table of Contents

Credits	1
What is this monster?	1
Basic Steps for Success	2
Other Neat Stuff	4
About VST plugins	4
System Requirements	4
Version History	5
The Future	7
If you find a bug (that is not listed below)	7
Known issues (PLEASE READ IF YOU VALUE YOUR SANITY!)	7
License	9

Credits

Maestro Frankenstein created by Arvid Tomayko-Peters

<http://arvidtp.net>

arvidtp@gmail.com

Based on ideas in the original Frankenstein created by Gabriele Rosseti

Made possible by a grant from Loccioni Corporation, Italia

<http://loccioni.com>

Many thanks to [Alessandro Montanari](#), Gabriele Rosseti and AEA/Loccioni

Portions © IRCAM, jasch and Peter Elsea and the Regents of the University of California

For more information and to contribute to Maestro Frankenstein's documentation yourself, visit our Wiki at

<http://studios.brown.edu/~arvid/mfwiki/pmwiki.php>. Also check out <http://arvidtp.net>.

Developed using [Cycling'74](#)'s Max/MSP environment and the [FTM library](#) from [IRCAM](#).

Icon image thanks to the [USGS](#).

What is this monster?

Frankenstein is a multi-track data sequencer. Each track corresponds to one x-y pair of data - usually one data file. Frankenstein considers on column to be time and one to be Y-values (you get to chose which is which). However, Maestro Frankenstein supports the importing of data files with any number of columns. Those with one column, such as maybe a file of tide data or a digital audio file, are assumed to be

sampled at a constant rate and the user is prompted to assign a sampling period or rate and a start value. Then data appears in the window, can be manipulated, mapped to a score, played back, spectrally analyzed and mapped to various parameters of VST and VSTi plugins for sound generation and effects, or mapped to various MIDI controllers. Each track can be exported as a MIDI file, or your whole composition can be bounced to an audio file, with 8 aux inputs for any external synthesizers you are using via MIDI.

Basic Steps for Success

1. Upon launch Maestro Frankenstein presents the user with a window containing transport controls with a scrollbar and a blank page. To create your first track, choose either 'New Track' from the 'Project' menu or the button on the toolbar.
2. Once the track appears it has a row of buttons on the bottom. Click 'Import Data'
3. From the resulting window, click the big yellow Import button and choose a file. Files must be tab-delimited text, comma-delimited text, (both easily savable from Excel, OpenOffice.org or your chosen spreadsheet software) or a raw binary data stream such as an uncompressed mono audio file.
4. If your data has only one column (such as mono audio), Maestro Frankenstein will ask you various questions about it. If not, then you must set which column contains the independent variable (used for time in Maestro Frankenstein) and which contains the desired dependent variable (primary data column). Also you have the ability here to specify where the data starts and ends (for instance in meters, millions of years, seconds, picometers, etc). Most of the time this will be done for you. You can input the correct units in adjacent the text field. A display on this window tells you how many of the data units one second in the Maestro Frankenstein's timeline is equivalent to. This should always be 1 when you first import data.
5. From this window you can also set the display color of the track or view the data in spreadsheet form (the 'View Data' button).
6. Next click the 'Edit Data' button and check out all the neat stuff you can do to that data, with a 10 step undo – that is 10 steps independently for every track! Sorry, redo is coming soon... ie when I get around to it.
7. Click the 'Create Score' button. Chances are you will already have a score computed (it is auto-updated when the track data is changed, but can also be updated manually). In this window you will be able to choose a key, scale and range of notes to map your data to. Make your choice and click 'Create/Update Score' (changes in this window do not automatically update the score). The colored progress-bar will show that your score has been created! Data is mapped to notes by superimposing the scale over the specified note range on the total range of data Y-values and placing a note for every data point (except with skipping enabled) in the scale degree closest to which it falls. Larger and smaller gaps between different scale degrees ARE taken into account.

8. You can now click 'View Score' to see your score on a piano roll. Hidden feature: if you right-click or Ctrl-click (on Mac if you don't have a multi-button mouse) you will get a contextual menu with the option to 'Inspect...' Choose this and a window will appear with different view modes to select from, such as viewing the score on staves.
9. Click the 'Instrument' button on the track to select one or more playback options (MIDI, [VSTi](#), Internal synthesizer).
10. Press play on the transport bar and listen!
11. If you are using internal instruments to play back scores in Maestro Frankenstein, choose 'Mixer and Recording' from the 'Display' menu to see your master bus, 2 master VST effect slots, aux inputs, DSP on/off switch (to turn audio processing off if you don't need it, or it is taking up too much of your CPU during non-audio related tasks). Here there is also a bounce to disk command which will ask you for a save location, then record a stereo audio file of your Maestro Frankenstein project to disk in realtime. Bit-depth selection coming soon.
12. Explore the other buttons at the bottom of the track.
 1. 'Map Data To Parameters' allows you to send data as a continuous control stream as a MIDI Controller or to VST instruments and effects within Maestro Frankenstein.
 2. "Frequency Domain" allows you to plot a power spectrum of the data and execute a spectral bandpass filter. If your data is not sampled at a constant sampling rate, it will need to be interpolated before doing any frequency domain processes – Maestro Frankenstein will ask to do this for you usually. Then you have all the typical FFT controls. This FFT power spectrum is based on an FFT of the whole data at once, so it can be very accurate. The best way to do a bandpass filter is to drag along the power spectrum to one of the bounds of your desired filter, then press the corresponding "Set Limit of Bandpass Filter" button. Do this for the other bound and press 'Execute Bandpass Filter'. The result can be seen, tweaked, shown as a spreadsheet, Exported and copied to the clipboard to be pasted into another track (in the Import Data window).
 3. 'Create Audio Sample' is fun – it turns your data directly into a sound wave which can be exported as an AIFF or WAV file! Usually these are noisy, but cool – and you can hear the frequencies of any periodic variation in your data. In the future this will be one of the available audio samples for a built-in FrankenSampler.
 4. The 'score' widget at the bottom right of the track shows the current state of score computation. Clicking it will create/update your score with the current settings. Its not as useful anymore since score creation was sped up hugely in 0.3b.
13. Add another track! You can have as many as you want. Infinite. However ONLY the first 256 will be saved correctly to a project file, and loading that project would be quite slow with that many tracks :) So there is a limit, but most pro audio applications don't go above 256 audio tracks anyway, so I think I am justified in choosing that as a limit!

Other Neat Stuff

Clicking and dragging in any track or on the fft display will report data values from the data point closest to the horizontal cursor position.

Try double-clicking any gray visual data display – it will split to reveal a resizable history display that can be dragged over to examine previous states of the display – useful for comparing plots with different fft settings, etc.

Zoom the 'View Score' window by dragging the timeline at its top (which is in ms).

Clicking Maestro Frankenstein's main timeline will set the play position.

There is a menu command to update all scores.

The DSP Status window (which will very be familiar to any Max/MSP user) is where you set which audio interface to use, buffers (ie audio latencies) (the trade-off is always more responsive and less delay = more processor intense, more delay = less processor intense) and audio channel configuration.

Overdrive in the edit menu and DSP status window should always be on for accurate playback.

Saving and loading of projects generally works well. Format consists of a folder with the project name. Inside is a .mfproj file with the project name (this is the master project file that you should choose to open when you want to re-open the project – don't be fooled by its diminutive 8 Kb size. It's just a listing of tracks to restore on load) and a folder called 'Tracks' containing all the essential track data. You can open and take a look at any of the files using a plain text editor if you are curious – they are all human-readable.

About VST plugins

Frankenstein supports the use of VST effect (on the master bus) and VSTi instrument plugins (one per track). These are small software plugins which either process (VST) or generate (VSTi) sound. Some are expensive, but there are many free or cheap instruments and effects available online for both Mac and Windows. Some good places to look kvr-vst.com, hitsquad.com, or <http://macmusic.org/> for Mac. However, not all of these are stable – so do some testing.

Maestro Frankenstein can map data to any published parameter of a VST or VSTi plugin. Just go to that channel's 'Parameter Mapping' page, refresh the parameter list if necessary (first select a track if you are mapping to a VSTi on another track), then choose the parameter from the list, and it will be inserted into the text field below, and will be controlled by that data. If you have the plugin open during playback, you will see the knobs and sliders move! Yeah!

System Requirements

Mac: Mac OS X 10.4 or later, 512 MB minimum of system memory recommended. Universal binary should run on both PPC and Intel macs.

Tested primarily on a 1.25GHz G4 Powerbook. A version supporting OS X 10.3 may also be forthcoming at some point, but at the moment you need 10.4 or greater.

PC: Windows XP (Home or Pro), with 500 Mhz P3 or better processor and 256 MB of system memory recommended. Java Runtime Environment (JRE) 6 (<http://sun.com/java>)

Both: At least 800x600 display (1024x768 or greater recommended), MIDI interface and/or [VST plugins](#) recommended to get some good sounds.

Version History

0.4 beta

- Finally, a new Windows XP build! Its not that well tested, but it seems to work so far. Please let me know of any problems!
- Added global commands window that allows you to make changes to all tracks at once. Accessed from the 'Display' menu.
- Added "Midi File Export Sync Assurance" option in the global commands that adds a tiny midi note at the beginning of each score so that exported MIDI files can be re-synced with confidence in sequencers such as Logic which don't import the beginning of a MIDI file if there is no data in it. Remember to remove this note when you put your piece back together in your sequencer!
- GREATLY improved the power spectrum plotting of maestro frankenstein. Plots are now done by default with one fft of a zero-padded set of the data. Average power spectra with several frames/lags using a windowing function can still be done.
- Improved peaks finding feature in the power spectrum – now you can sort by intensity before viewing/exporting peak data. Also it usually works now, and lets you set how many peaks to find with live feedback in the plot.
- Power spectrum plot window enlarged and vertical view scaling bar improved, with a number field for textual vertical view ceiling entry.
- Controls in the 'Create Score' window now automatically update the score
- New brighter UI look in main window
- Time Scaling changed again – I realized my previous way of doing it was stupid. This method scales the original space between the original 0 and the original first data point, but does not scale any time offset you have added manually in Maestro Frankenstein. Thus temporal relationships between datasets can be maintained when they are all scaled by the same amount.
- Track colors are now saved
- [Change] since score update is now faster and happens automatically, the colored "update

score button” of each track in the main window is now called “Export Score” and now exports the score as a MIDI file in one click, instead of opening the “Create Score” window for each track.

■ [Fix] More bug fixes than I can (or want to) remember. Here are a few:

- Fixed an annoying bug that could skip the first note when generating a score under certain conditions.
- Getting velocities of notes from a column of data from the same track will now work the first time after a previously saved project is reloaded
- Maestro Frankenstein no longer takes FOREVER to quit. However, it still takes a long time to erase tracks or to start a new project if you have one open with lots of tracks. This is due to Max's patrstorage object being really slow at reorganizing itself and there being no way I can find of stopping it from doing aside from deleting the object before close. This is what I do when you quit, so it makes it faster. But I can't do that otherwise, or Maestro Frankenstein will cease functioning altogether. So, I suggest quitting a restarting before creating or a new project or opening another project if you have more than three tracks currently open.
- Bugs with sample rates in power spectrum window
- Getting note velocities from other tracks now uses interpolation instead of the value of the last data point

● 0.3 beta

- Added playhead indication on timebar and page-scroll during playback. A green line on the timebar shows your progress during playback, placing a tick mark where each note is played. When you reach the end of your screen, Maestro Frankenstein will now scroll to the next page.
- Score generation is now exponentially faster.
- [Change] Scaling of data now takes into account only the time spanned by the data, eliminating some display and computation errors.
- [Fix] Indication of % length of data is maintained between saves for all new documents, so pressing 100% will always return you to the original length of the data, even after closing and re-opening a project.
- [Fix] Several small unit display and project loading bugs fixed.

● 0.26 beta

- [Fix] Now works on Intel macs – 0.25 crashed on creating a new track in Intel based macs. Sorry about that!

● 0.25 beta

- Built as a Universal Binary with FTM 1.7.11 – no longer are there two separate versions for PPC and Intel Macs!
- A few minor cosmetic changes and console error messages eradicated

● **0.24 beta**

- Redesigned the windowed interface:
 - Maestro Frankenstein is now all contained in one easy to manage window, saving you screen space and headache
 - The timeline bar no longer scrolls vertically with the tracks, so it will not disappear when working on tracks at the bottom of a large project :)
 - The window height automatically adapts to your screen size
 - The horizontal scrollbar is in the same window as the tracks
- Fixed problems with loss of some settings on project load in version 0.22 beta
- Fixed anomalous behavior of audition controls in “Create Audio Sample” window
- Added Master Playback Speed Control in the toolbar: allows for adjustment of playback speed from 0.001x to 1000x in realtime during playback. This is an easy way to speed up or slow down playback of all tracks at once. However it does not affect the speed or tempo of exported MIDI files.
- Slight file format change (unfortunately this breaks compatibility with projects saved using earlier versions on Maestro Frankenstein and there is no easy way to convert them at this time).
- More accurate rendering of graphs thanks to updates to the FTM library.

● **0.22 beta – Mac only (PPC and Intel)**

- Fixed graphical bug with windowing type display in Average Power Spectrum window
- Updated About box to be less stupid

● **0.21 beta – Mac only (PPC and Intel)**

- A few little user interface improvements
- FFT settings and parameters are now actually stored or stored more intelligently than before
- Compiled with FTM 1.7.8
- Bug Fixes:
 - Track settings are now properly restored when a 2nd project is loaded without restarting Maestro Frankenstein
 - nasty fft interpolation graphical bug gone (hopefully)

- fft display now displays correct information from cursor if sample rate is less than 1
- scaling system for various information displays
- redesigned to ALWAYS give correct values
- Files saved with beta 0.2 should still be compatible
- Documentation Wiki up and running at <http://studios.brown.edu/~arvid/mfwiki/pmwiki.php>
- **0.2 beta – Mac only (PPC and Intel)**
 - First public release
- **0.1 beta**
 - The first somewhat usable version of Maestro Frankenstein

The Future

In the future I would like to add new rhythm and pitch mapping algorithms to Maestro Frankenstein. Also I have begun work on a running average FFT that will be rendered as a 2D colored image – i.e. A spectrogram. I would also like to streamline and further modularize the engine to eliminate some of the limitations of the current track system.

I am also planning a Windows build as soon as I can secure a windows machine to debug and build Frankie on. This would at first be for XP, and it is unknown whether it would work on Windows Vista until I get a machine running Vista to test on.

If you find a bug (that is not listed below)

If you find a bug that is not listed below, please email me at arvidtp@gmail.com and tell me how to reproduce your problem. Unfortunately, due to me not having a PC, any Windows-specific problems may take longer to get fixed. Thanks!

Known issues (**PLEASE READ IF YOU VALUE YOUR SANITY!**)

This is beta quality software, and there not only "will probably be bugs", but there are KNOWN to be some. So save/export often.

- Importing stereo audio files creates a mess, as Frankie thinks that one channel is time and the other data values. But you can see stereo correlation this way :)
- Frankensamp - the sample player - does not yet exist in working form. Right now you can loop audio samples for audition and export them as wavs or aifs or the other random formats FTM supports. But you can not play back scores.
- Sliders for parameters in VST plugins without custom GUIs may be invisible. Complain to Cycling '74, the makers of Max/MSP about this. They still work, trust me, you just can't see the sliders.

- On Mac, do not close java-based windows (ie score or data view windows) with the keyboard command Cmd-W (⌘-W). Due to a bug in FTM, this may cause the window to spastically open and close later on during iterative operations, once for every operation. This can take a long time to finish, and keep Maestro Frankenstein unresponsive all the while. It usually will not crash Maestro Frankenstein, but it is very annoying until it finishes. If this happens I would suggest saving and restarting Maestro Frankenstein.
- Undo and Redo in the edit menu do not do anything, and may never. Use the Undo in the 'Edit Data' window on each individual track.
- FFT data does not yet contain frequencies, periods or original data frequencies or periods – just the raw intensities at each fft bin and its phase. These will be added in the future. However this info is present in the peaks data.
- In the data view window (spreadsheet) please do not use the 'Import...' item from the 'fmat' menu to import data. Some routines will not get triggered and it will probably be a mess. Always use the big yellow "Import" button instead.
- Sometimes if Maestro Frankenstein is left running for long periods of time, such as overnight, opening the java based editors (ie view data, view score, view fft data) may cause frankie to lock up. This I believe is either a bug with FTM or Max/MSP's java handling. Its not under my control at the moment.
- If you try to apply one of the 'edit data' functions while another function is processing, they will probably both apply at once and you may end up with something quite screwy. So just wait. Some operations, such as fft, audio sample creation and score generation, do not modify the original data and can occur simultaneously without need for concern.
- Max/MSP's vst handling is not the tightest, so some 3rd party VST plugins with small bugs may cause Maestro Frankenstein to crash - especially if opened while playback is occurring with some data mapped to one of their parameters.
- Track colors are not restored on reloading a project.
- If you grab your note velocities for track A from a second track (lets say track B), the score for track A will not update when the track B changes. The velocities will update only when track A changes, or when track A's score is manually updated.
- Frankensynth sounds crappy. I know.
- Generally don't press cancel on any save dialogs – there are still some cases where this is not detected and your file is closed anyway without being saved. I'm working on this. If in doubt, save. A Maestro Frankenstein Project File is not very big on disk unless you have tons of big tracks.
- There are a few errors that will come up in the Maestro Frankenstein console (Status Window). They are just from somewhat sloppy programming and should not hurt. I can't find where they are coming from or I would eliminate them.

- Maestro Frankenstein will not ask you to save when he quits or exits – so remember that and save first!
- On Mac Maestro Frankenstein may try to launch web locations and other such things in the same folder as it. So keep it in its own folder. Hopefully Cycling '74 has fixed this.
- Double clicking a .mfproj file does not work as a way to load projects. You must choose 'Open' from within Maestro Frankenstein.
- Deleting tracks or closing large projects sometimes takes a long time. I don't know why.

License

This software is released as free, open source software under the GNU GPL license. Anyone who has Max/MSP can view and edit the source, which is available from http://arvidtp.net/sw/maestro_frankenstein.html. A copy of the license can be viewed online at <http://www.gnu.org/licenses/gpl.txt>.