SaneStation, TG2012 edition

Steinar H. Gunderson (Sesse/Excess)

March 23, 2012

Hello, and welcome to SaneStation!

This document outlines things that are particular to the TG2012 SaneStation compo. In particular, will tell you how to prepare your entry for submission.

About SaneStation

SaneStation is a 64k synth, originally envisioned around 2002 and first making its debut in “Nemesis” by Excess vs. Kvasigen at TG04. Since then, it’s been vastly expanded, optimized and polished, but remains a fixed-function virtual-analogue synth.

SaneStation is described in detail in the included manual (see manual.pdf). The synth + song + bank data + player is about 16 kB when compressed with kkrunchy, but the version given here is compiled with more optimization and static linking, so the player is more like 1 MB. If you want to use SaneStation in a production, and think 1 MB is a bit over the budget for your 64k, feel free to contact us for a leaner version (see “Credits/contact”, below).

Installing SaneStation

The archive includes a VSTi plugin which should be usable by most hosts supporting VST 2.6 or higher; both Windows and Mac OS X compiles are included, and should work and sound the same. Simply copy the SaneStation DLL (or equivalent on Mac) into your VST host’s plugin directory, and it should be available.

In your entry, you should only use one instance of the synth, so you are limited to a maximum of 16 channels at a time, although you can switch programs freely during your song. (This is a restriction in the reference player only, so using more instances temporary during composing is perfectly fine if you so desire.)

TG2012 specific notes

The TG12 version of SaneStation has been compiled without LPC support. Thus, please disregard all references to it in the manual (in particular, section 2.1).

Preparing your entry

In the compo, all entries will be played back using the included reference player. It has versions for Windows, Mac OS X and Linux, which should all work and sound the same.

Before handing in your entry, you will need to convert it into a standard bundle, which contains everything you need to play your tune in a single file. This is done as follows:

1. Set your preferred song title and author in the “globals” tab. This will be showed by the standalone player.

2. From your VST sequencer, export the song to a standard MIDI file (SMF), e.g. song.mid.

3. From the SaneStation UI, export the settings bank using the “Save bank” button, e.g. to song.sss.

4. Use the included command-line tool “compile” to compile the two into a single standardized bundle, e.g. “compile song.mid song.sss song.emp”.

5. Listen to the EMP file using the included reference player to make sure it sounds right, e.g. “player song.emp”. Optionally you can render it to a .wav file using “player song.emp song.wav”, but we do not want the .wav file for the compo (we will make our own renders).

6. Hand in the EMP file as your compo entry.

Please make a test run well ahead of the deadline! Testing your song in the reference player will avoid any issues with the player being different somehow from the player in your sequencer, and ensure that your entry is played back correctly.

Special procedures for Ableton Live

Ableton Live has limited support for MIDI export—it can only export MIDI files containing of one track. Thus, the procedure for preparing your entry is slightly more involved:

- Instead of making one SMF (.mid) file, you will need to make one per channel, exporting each track separately.

- When compiling, give all of your MIDI files in order on the command line, e.g. “compile track1.mid track2.mid track3.mid song.sss song.emp”. The first MIDI file will be mapped as usual, the second MIDI file will be mapped channel 0 → channel 1, the third will be mapped channel 0 → channel 2 and so on. This should properly merge the files back into a complete song.

If you experience problems with the procedure, feel free to contact us for more help (see below).

Credits/contact

SaneStation was programmed by Rune Holm (titanstar/Excess) and Steinar H. Gunderson (Sesse/Excess). Erik Faye-Lund (kusma/Excess) and Tor Øines (OptimizE/Kvasigen) contributed graphics, and the latter as well as Ola Gundelsby (lug00ber/Kvasigen) worked as musicians and testers. Many others have contributed with testing and various forms of feedback—in particular, Sverre Holm provided invaluable DSP consultancy in the earliest days.

If you find bugs or problems with SaneStation itself, contact me (Sesse); I can probably fix particularly bad bugs, but there will be no new features before deadline. You can reach me by email to steinar+demoscene@gunderson.no, or as Sesse on IRC (EFNet/IRCNet/Freenode/OFTC will do). For compo-specific questions, please email creativia@gathering.org.

Changes

v1.02, March 23rd

- Fix a bug where the reference player could sometimes ignore the last note-off.
- Fix a bug where exported banks could be truncated, leading to either compile errors or wrong settings on instrument 127. This could also affect exported instruments.
- Make the compile tool support merging multiple MIDI files, for the benefit of Ableton Live users (Live can only export one track at a time). Added a “Special procedure for Ableton Live” subsection to this document, documenting the merging procedure.
- Fix a problem where MIDI files exported from Ableton Live with tempo at exactly 120 bpm would be interpreted as 0 bpm.
- Change the default “max voices per channel” from 1 to 96, to confuse musicians less.
- Better VST automation filtering of illegal values. Fixes, among others, problems with the “Random” function in Renoise, and with startup in Buze when not using pVST.

v1.01, March 22nd

- Fix a bug where wrong automation information would be sent back to the VST host. Most notably, this would cause the assertion failure “pb_msb.size() == pb_lsb.size()” when changing the osc 1 delay in Ableton Live.

v1.00, March 21st

- Initial release.