WHAT'S NEW IN THE FAUST ECOSYSTEM AND COMMUNITY?

Stéphane Letz - Romain Michon - Yann Orlarey

GRAME - INRIA LYON

IFC, Saint-Etienne, June 7th 2022

AGENDA

Compiler: new backends, new options, internal API access

New architectures

Debugging, optimizing, deploying

New libraries: added in standard and external ones

Documentation, workshops, tutorials

Animating the community

Perspectives ?

COMPILER: BACKEND

Three new backends in the compiler:

- **C# backend** contributed by Mike Oliphant (possibly allowing dynamic compilation using the .NET compiler system)

- **Dlang backend** contributed by Ethan Recker, to be used in Dplug (audio plug-in framework)

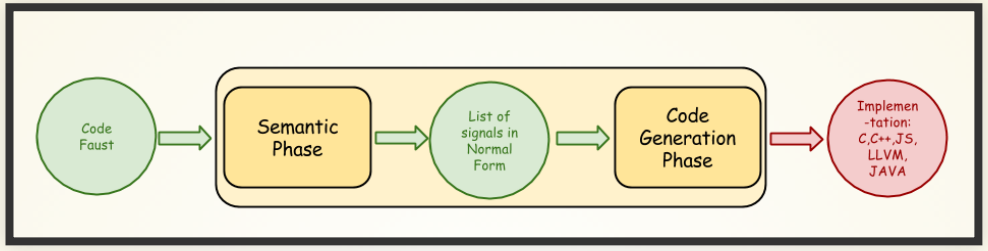
- **Julia backend:** close interaction with the Julia environment (Machine Learning at signal level)

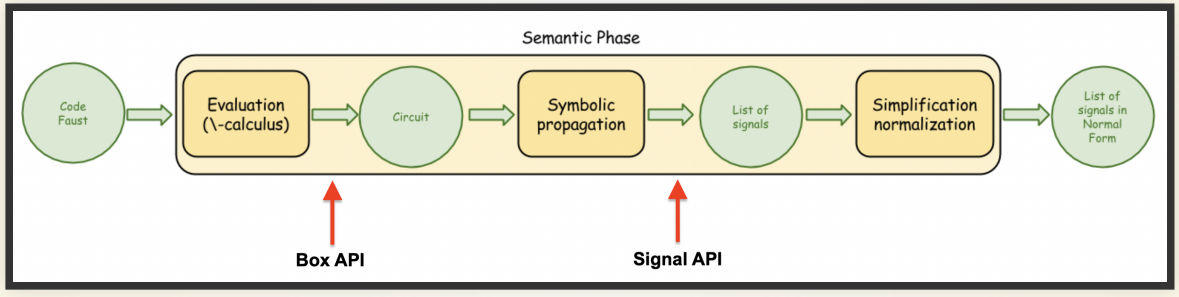
COMPILER: OPTIONS

Reworked **-mem** option to better control the DSP memory layout (embedded platforms)

New **-os** option (one-sample) and separated **control** and **compute** functions to generate C/C++ code adapted to FPGA export (FAST project)

ACCESSING "INTERNAL" APIS IN THE COMPILER





Using the signal simplication (normal form) and code generation phases

THE BOX API

Exposed in libfaust-box.h C++ header with tutorial here

Testing something different than Faust Block Diagram Algebra ?

Building box expressions algorithmically, Machine Learning at symbolic level ?

THE SIGNAL API

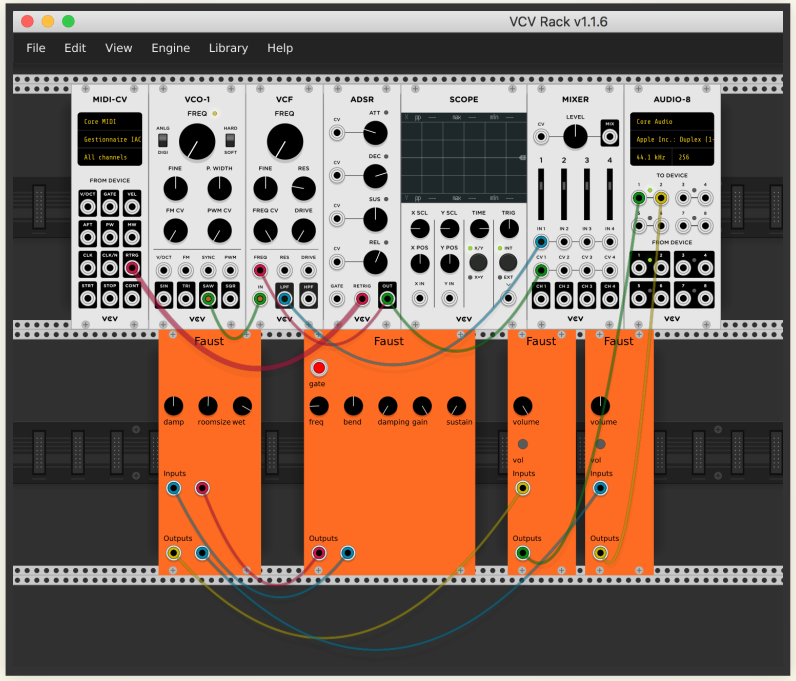
Exposed in libfaust-signal.h C++ header with tutorial here

Building signal based new languages ? Gaphical ones?

Elementary audio language as an example

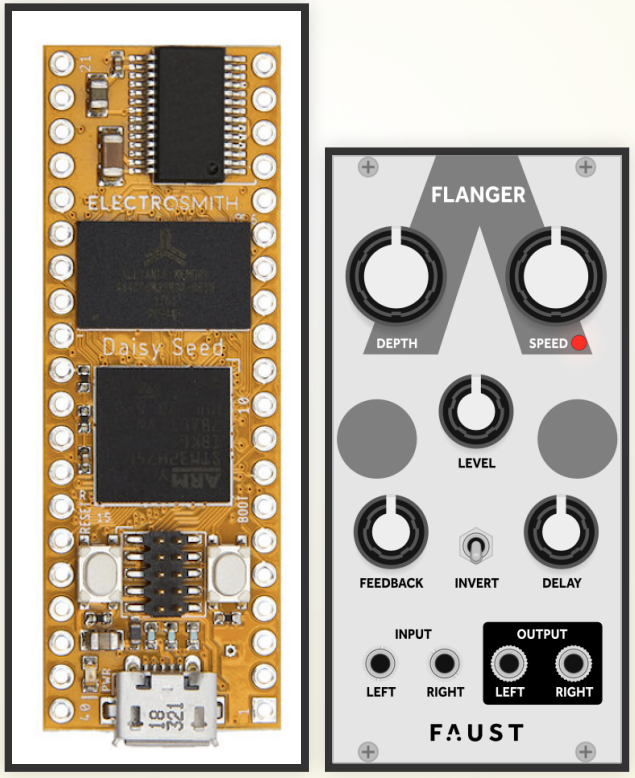
NEW ARCHITECTURES (1)

Modular Synthesis with VCV Rack: **faust2vcvrack**



NEW ARCHITECTURES (2)

Electro-Smith Daisy seed, patch... programming with **faust2daisy** or **Eurorack-blocks**



DEBUGGING

New **-ct** and **-cat** options to check tables

New **-me** option to check math function domains

Improved **interp-tracer** tool (using the Interp backend)

Debugging documentation

OPTIMIZING

Improved **faustbench** and **faustbench-llvm** tools to discover the best compilation options for a given DSP

The **faust2object** tool for multi-CPU compilation

A real use-case with Expressive E Noisy synthesizer

Optimizing documentation

ADDED STANDARD LIBRARIES

aanl.lib library developed by Dario Sanfilippo (Aliasing-suppressed nonlinearities through first-order and second-order approximations of continuous-time signals)

fds.lib library developed by Riccardo Russo (Finite Difference Schemes physical models)

wdmodels.lib library developed by Dirk Roosenburg (Wave Digital Filter)

COMMUNITY CONTRIBUTIONS

abclib library developed by CICM/MUSIDANSE (research, teaching, and creation in mixed music)

Edge of Chaos library developed by Dario Sanfilippo (musical complex adaptive systems)

realfaust library developed by Dario Sanfilippo (domain-limited versions on math functions)

bitDSP-faust library developed by Till Bovermann and Dario Sanfilippo (bit-based algorithms)

SEAM library library developed by Alvise Vidolin and Nicola Bernardini (Sustained Electro-Acoustic Music)

DOCUMENTATION, WORKSHOPS, TUTORIALS

Documentation: architectures, debugging, optimizing, librairies...

Workshops given during COVID time

Several tutorials added

COMMUNITY

Rebirth of the Faust Slack

#faust channel on The Audio Programmer community (Joshua Hodge)

Two Google Summer of Code projects accepted this year: **Integration in HISE (Roman Sommer)** and **Integration in Bespoke (Drew James)**

Powered by Faust page

PERSPECTIVES ?

Analyzing the Faust survey

Faust ideas page

Your ideas ?