Visual Programming for Multimedia Artists

Jaymie Strecker
Steve Mokris
Melissa Egan
Jean Marie Cackowski-Campbell
Karl Henkel
Bradley Mellen
Matthew Radcliffe

vuo.org
vuo.org/contact
kosada.com

Vuo

Many multimedia artists want
to create original works
of interactive art and music.
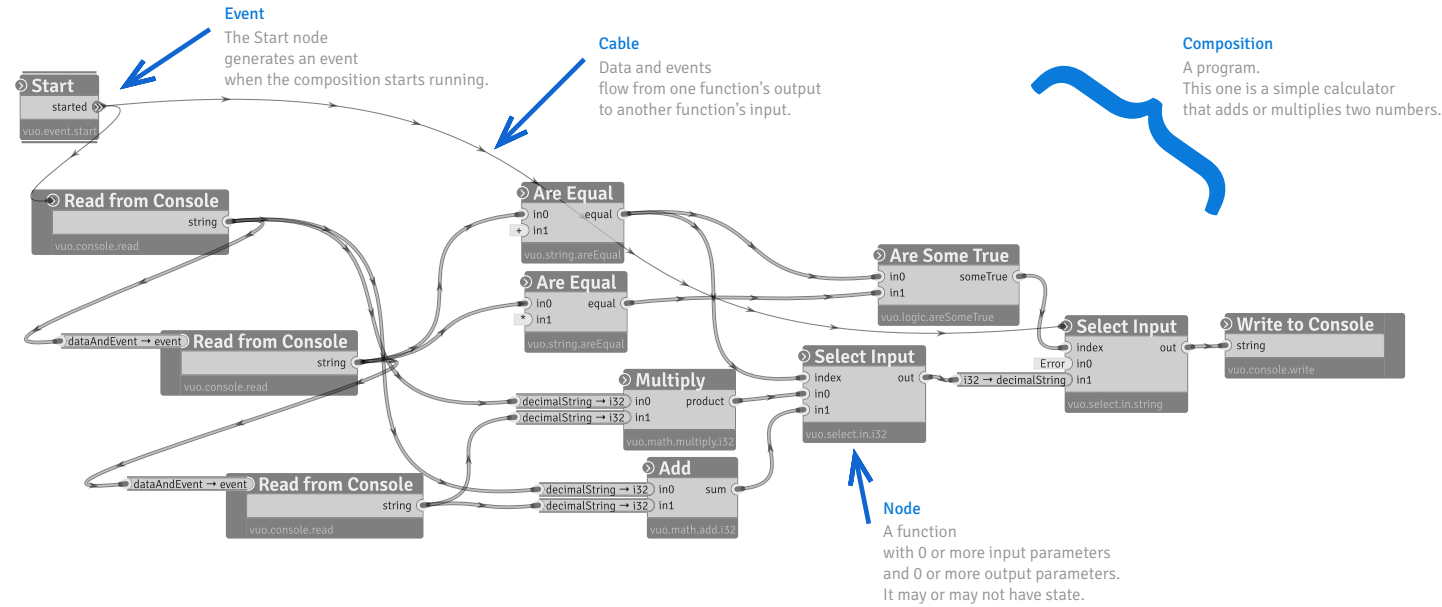Yet their background
is in art or music,
not programming.

Vuo,
being a visual language,
is a solution to this conundrum.

The multimedia artist
creates a program (or "composition")
by dragging nodes onto a canvas
and drawing cables to connect them.

Instead of worrying about syntax,
the multimedia artist
can focus on logic and data flow.

With Vuo,
multimedia artists and developers can:

- Improvise: Change a composition while it's running.
- Debug: Inspect data within a running composition.
- Create graphics with modern OpenGL and scenegraphs.
- Build multi-threaded, native executables.
- Package executables into distributable apps.
- Invoke and control compositions from other applications.

Event
The Start node
generates an event
when the composition starts running.

Cable
Data and events
flow from one function's output
to another function's input.

Composition
A program.
This one is a simple calculator
that adds or multiplies two numbers.

Node
A function
with 0 or more input parameters
and 0 or more output parameters.
It may or may not have state.

Some multimedia art

1024 Architecture
turned a building into
a dancing, talking,
interactive character.
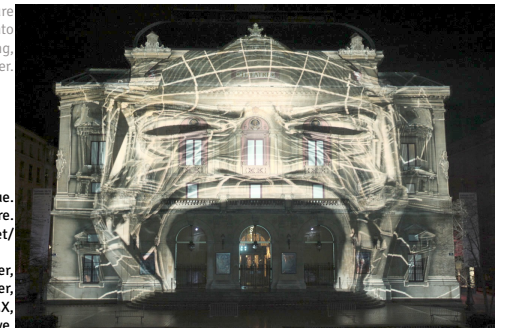
Perspective Lyrique.
Designed by 1024 Architecture.
http://1024architecture.net/

Created with MadMapper,
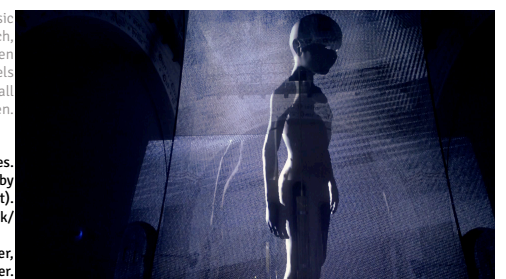**Quartz Composer,**
VDMX,
Ableton Live.

In this traveling exhibit,
visitors step inside a box
and control
color relationships
with their bodies.

Colorbox.
Designed by Gabriel Mott.
http://colorbox.me/

Created with Kinect,
Quartz Composer,
Synapse.

To accompany music
in a church,
Vibeke Bertelsen
animated 3D models
on a 13-meter-tall
transparent screen.

Uncanny Creatures.
Designed by
Vibeke Bertelsen (Udart).
http://udart.dk/

Created with Poser,
Quartz Composer.

Many musicians
at the electro-music festival
team up with visualists
or create their own visuals
to complement
their music performances.

Performance at
electro-music 2012.
Music composed
and performed by
Shannon Lee Hayden.
http://shannonleehayden.com/

Visuals by Steve Mokris.
Created with Quartz Composer.

Photo by Hong Waltzer.
http://picasaweb.google.com/hongwaltzer

Supporting text languages

Vuo users will be able to extend the language
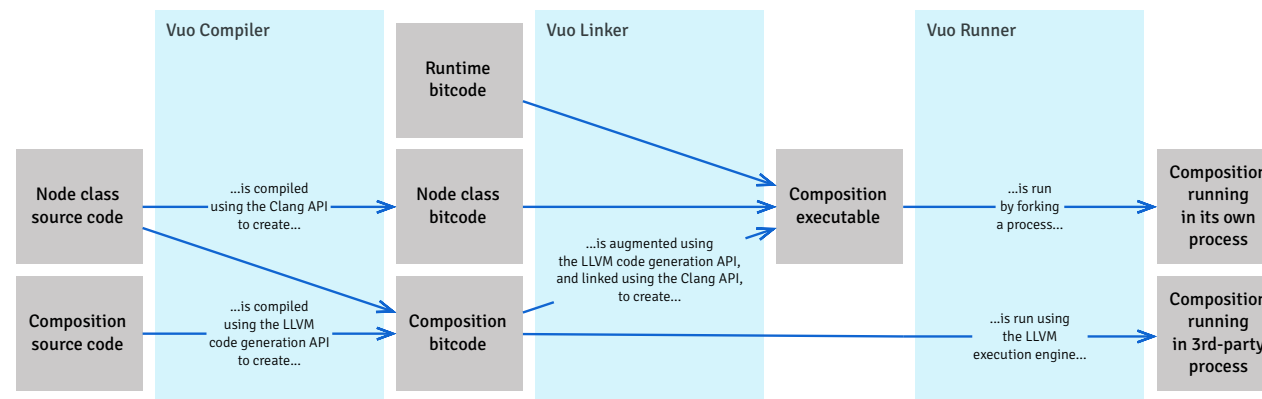by making their own node classes.

Multimedia artists with programming experience,
and developers wanting to market to multimedia artists,
will be able to implement node classes using Vuo's API.

- The initial release will support writing node classes in C.
- Support for implementing node classes in other text languages,
  and in Vuo's graph language, is planned.
- Node classes can wrap existing libraries.
  Vuo users will be able to easily combine libraries
  into imaginative, original compositions.

LLVM, and projects built on LLVM,
solve the problem of parsing many languages into a common format
(the LLVM intermediate representation).

- C, C++, and Objective-C can be parsed by Clang.
- Ada, C, C++, and Fortran are fully supported,
  and other languages partially supported, by DragonEgg.
- C# and other .NET languages can be parsed by Mono.
- Python, Lua, Haskell, and other front-ends
  are available from various external projects.

How Vuo uses LLVM

Vuo Compiler

Vuo Linker

Vuo Runner

Node class
source code

...is compiled
using the Clang API
to create...

Node class
bitcode

Runtime
bitcode

...is augmented using
the LLVM code generation API,
and linked using the Clang API,
to create...

Composition
executable

...is run
by forking
a process...

Composition
running
in its own
process

Composition
source code

...is compiled
using the LLVM
code generation API
to create...

Composition
bitcode

...is run using
the LLVM
execution engine...

Composition
running
in 3rd-party
process

Supporting operating systems

LLVM frees Vuo's developers
from the difficulty of supporting multiple platforms,
letting us focus on making Vuo powerful and fun for its users.

Vuo frees its users
from the technical constraints of being locked into one platform,
letting them focus on creating amazing works of multimedia art.

For editing compositions:

- The initial release will support Mac.
- Support for Linux and Windows is planned.
- LLVM, Clang, and other dependencies
  are open-source and cross-platform.
  This makes it easier to port from Mac to Windows and Linux.

For running compositions:

- The initial release will support Mac.
- Support for Windows, Linux, iOS, and Android is planned.
- LLVM and Clang can generate code for these (and more) platforms.
- LLVM and Clang support cross-compiling.
  A user could develop a composition on Mac and deploy it to Windows.

Visual programming environments

	Max	Pure Data	vvvv	Quartz Composer	Vuo
First release	1990	1996	1998	2005	2013
Supported platforms for editor	Windows, Mac	Windows, Mac, Linux	Windows	Mac	Mac (Windows, Linux planned)
Supported platforms for compositions	Windows, Mac	Windows, Mac, Linux, iOS, Android	Windows	Mac	Mac (Windows, Linux, iOS, Android planned)
Supported text languages for nodes	C, GLSL	C, GLSL	C#, HLSL	Objective-C, JavaScript, GLSL	C (more planned)
Are compositions compiled?	◐	○	○	○	●
Is it open source?	○	●	○	○	◐