OpenMusic - Linux-port

Anders Vinjar - Jean Bresson

May 3, 2014
1. OpenMusic Intro

2. History and Motivation

3. The OM-Linux Project

4. Conclusions and Perspectives
OpenMusic Intro

OpenMusic is:

- A visual programming language
- A computer-aided composition environment

Based on Common Lisp
Off-line / demand-driven
Local state, partial evaluations...
Extended with editors, time structures

An environment for...

Music composition

Music composition

Contemporary composition = ill-defined

⇒ good CAC tool adapts well to human artistic processes, whatever those may be

Miller Puckette: “CAC is in effect making the computer carry out thought processes previously carried out in human brains”
Music research and analysis
Control of DSP / synthesis tools


OM Development vs. Linux - Quick History

- 1997: OM release on Macintosh Common Lisp (MCL)
- 2003: 1\textsuperscript{st} Linux attempt = OM 4 on CMUCL
- 2005: OM on Mac/Windows. OM-API (AllegroCL / MCL)
- 2006 2\textsuperscript{nd} Linux attempt = OM 5 on SBCL
OM Development vs. Linux - Quick History

- 1997: OM release on Macintosh Common Lisp (MCL)
- 2003: 1st Linux attempt = OM 4 on CMUCL
- 2005: OM on Mac/Windows. OM-API (AllegroCL / MCL)
- 2006 2nd Linux attempt = OM 5 on SBCL
- 2008: OM 6 on LispWorks
- 2013: Linux-support - OM 6 on LispWorks
• Started 2013

• Supported by BEK
  Bergen Center for Electronic Arts
  http://www.bek.no

• In line with recent OM developments
  (modular code + APIs, optional external dependencies)
OM, why Linux? (Motivations)

- Useful addition to Linux sound/composition tools
- Good people - institutions, communities, users, devs
- Integrate with Linux apps (through eg. libraries), R&D
- Improve OpenMusic - general, independent (OS-es, hardware, libs...)

Anders Vinjar - Jean Bresson
1. The OM general architecture is portable across platforms:
   - kernel is 100% ANSI Common Lisp
     = easy

2. Other dependencies:

   • windowing & GUI toolkits to build the visual language and editors
   • I/O and rendering (MIDI, Audio, OSC...) 
     = not so easy
OM is a visual programming environment, and highly dependent on graphical user interfaces:

- patching, programming
- editors
- visual feedback
- Lisp interaction, control
- cross-referencing, interactive help-system...
OM 6 uses LispWorks’ **CAPI** interface:

= “cross-toolkit” (Gtk, Cocoa, Windows...)

⇒ only moderate tweaks needed to adapt OM 6 to LW-Linux:

- text-fonts (text-panes, comments)
- dimensions, screen-resolution
I/O AND RENDERING

- Attempt to port old solutions = no good
- LibAudioStream? (already in use on other platforms...)
- Alsa? Linux-native
- ⇒ More general solutions: Lisp, JACK, Port...
I/O AND RENDERING - MIDI:

MidiShare replacement:

- **Midifile (SMF) R/W ⇒ CL-MIDI** (message formatting, file I/O)
- **Scheduling ⇒ OMs new internal scheduler, JACK scheduler, other schedulers (SC3)**
- **Communication ⇒ Jack, PortMidi**
LibAudioStream replacement:

- File I/O ⇒ libsndfile Lisp wrapper
- scheduling and rendering ⇒ Jack: client, ports, hw, I/O buffering, callback-engine
CL-Jack client - MIDI & Audio I/O & scheduling

- self-contained (CL + CFFI)
- AUDIO + MIDI I/O (hw, external apps)
- Scheduling, queues, buffers (CL and libjack)
- Callbacks programmable in CL

Example clients, hooks, callbacks, external processes:

- **mplayer** - subprocess-control
- **SuperCollider** client (OSC)
  - control (start, connection, operation)
  - I/O
  - callbacks on scheduler
- **Fluidsynth** client
  - internal softsynth
Offsprings

Plans...

- CCRMA-ware – CM, CLM, Sndlib
PACKAGING, DISTRIBUTION

- OM is LGPL (LW is not)
- `.rpm` - (sources + image, dependencies)
- `.tar.bz2` - (sources & image)
- recipes for preparing `dpkg`-compliant packages at the forum:
  http://forumnet.ircam.fr/user-groups/openmusic/
- dists?
STATUS MAY 2014

- Working Linux-port
- No (known) big obstacles
- common source-tree for all 3 platforms
  - modular code + APIs $\Rightarrow$ reusability
  - dependencies less strict $\Rightarrow$ user/programmer choices
Further development depends on involvement:

- Users - feedback, bug-fixing, ideas
- Linux-developers - OM + integration w. other tools
- Libraries, (eg. OM-SoX, LilyPond)
- IRCAM developer(s)
Typical Linux-based projects:

- open source projects (JACK, ALSA, Ardour, SuperCollider, Lilypond...)
- user-groups, sourceforge, github
- active participation on many levels:
  - pro
  - semi-pro
  - end-user involvement
Conclusions

- a functional Linux-port is provided
- x-platform OM is better OM
- new solutions useful on osx/win
- get good projects going - to further development
Conclusions

- a functional Linux-port is provided
- x-platform OM is better OM
- new solutions useful on osx/win
- get good projects going - to further development
Big thanks!

- BEK - http://www.bek.no - Trond Lossius
- ANR EFFICACe - ANR-13-JS02-0004-01
- Lisp-developers, CCRMA (Bill Schottstaedt, Rick Taube, others)
- Linux music community
  - Jack developers
  - beta-testers
- OM users, mailing-lists, forums
OpenMusic - Linux-port

Thanks for your attention!

http://repmus.ircam.fr/openmusic/linux

http://forge.ircam.fr/p/OM6/downloads