KeyKit

A Scripting Language and GUI Interface for MIDI

Slides from a talk originally given at ElectroMusic 2005

Tim Thompson
me@timthompson.com
https://timthompson.com

Outline

- History
- Language and GUI
- Algorithmic, Interactive, and Realtime Tools
- Hardware Interfacing
- Recent projects

What is it?

- Textual programming language – procedural, interpreted, multi-tasking, graphics, object-oriented
- Specialized for MIDI algorithmic and realtime manipulation, first-class “phrase” data type, realtime scheduling
- Multi-window graphical user interface, pull-off menus and buttons, tools include multi-track editor, drum pattern editor
- Entire user interface and all tools written in the language and provided as source code in library, easily customized and extended

Reason for Being

- Hacking - fun, programming, normal music composition, algorithmic music
- Learning - OS’s, device drivers, graphics, user interface design, OO
- Personal and programmer-centric, not commercial, although it is usable enough for non-programmers

Development History

- 1.0 - BASIC-like
- 2.0 - realtime
- 3.0 - grammer, rewrite, fast enough to avoid built-ins
- 4.0 - graphics
- 5.0 - first multi-window attempts, multi-tasking, fifos, tools using tasks/fifos
- 6.0 - object-oriented, multi-window interface completely rewritten, pull-off menus/buttons
- 7.0 – support for multiple MIDI ports

Development machines

- Atari ST, PC (286), UNIX (386, X11), Windows NT, Win95/98/XP
- Ported at various times to: Atari ST, DOS, UNIX PC, Mac, Amiga, SVR3, SVR4, SunOS, NeXT, X Windows, SGI, Amdahl, VAX, 5620, Plan 9, Windows 3.1/NT/95/98/XP

MIDI Phrase is a first-class data type

- Time-ordered list of MIDI “notes” - can be system-exclusives, isolated note-on, isolated note-off, or full note with duration
- Constant value syntax

```
ph = 'c e g'          # ph is a c major triad
ph = 'dc2,e,f'        # ph is an arpeggio, channel 2
```

- Structure-like manipulation of attributes :

```
ph.dur = 1b           # all note durations = 1 beat
ph.pitch += 12        # transposed up an octave
```

Phrase/Note Attributes

- pitch (0-127)
- vol(0-127)
- chan (1-16)
- dur (in clicks)
- time (in clicks, relative to beginning of phrase)
- type (NOTE, NOTEON, NOTEOFF, MIDIBYTES, PROGRAM, PRESSURE, etc.)
- length (of phrase, independent of notes in it)
- attrib (string, arbitrary meaning)
- flags (integer, arbitrary meaning, bit 1 == picked)

Phrase Operations

- Serial concatenation
  ```
  ph = ph1 + ph2
  ```
- Parallel merging
  ```
  ph = ph1 | ph2
  ```
- Removing notes
  ```
  ph = ph1 - ph2
  ```
- Matching notes
  ```
  ph = ph1 & ph2
  ```
- Nth note
  ```
  ph = ph1 % n
  ```

Phrase Operations - the “select”

```
ph = ph1 { ??.pitch > 60 }
ph = ph1 { ??.dur > 1b }
ph = ph1 { isonbeat(??,4b) }
ph = ph1 { ??.number < 4 }
ph = ph1 { rand(3)==0 }
ph = ph1 { isinscale(??,scale) }
ph = ph - ph { ??.type == MIDIBYTES }
```

Language Features

- Inspired by awk (a Unix scripting language)
- Variables need not be declared
- Semicolons not required
- `#define`, `#include`
- The usual control structures and expressions (although no `switch`)

Phrase Operations - Looping

```
# randomize volume of each note and
# construct a new phrase with the result
r = ``
for ( nt in ph ) {
     nt.vol += rand(10)
     r = r | nt          # or    r |= nt
}

# randomize volume of each note, in-place
for ( n=0; n<sizeof(ph); n++ )
     ph%n.vol += rand(10)
```

Function values

```
function major(k) {
        return(k|transpose(k,4)|transpose(k,7));
}
function minor(k) {
        return(k|transpose(k,3)|transpose(k,7));
}
function randchordtype() {
        if ( rand(2) == 0 )
                return(major)
        else
                return(minor)
}

f = randchordtype()    # value of f is a function
f('c')
randchordtype()('c')
```

Other Language Features

- Variable arguments - … , nargs(), argv(), varg()
- Fifos and locking
- Objects
- Graphics - primitive elements are:
  lines, rectangles, text, windows,
  phrase windows, menus
- Machine-dependent hook – mdep() – used to add/expose non-portable features
- TCP/IP hooks available for Windows and Linux, network interaction

Variable Arguments

```
function calleither(f1,f2,...){
  if (rand(2) == 0) {
     f1(...)
  } else {
     f2(...)
}

P = calleither(flip,reverse,p)

P = calleither(scadjust,scafilt,p,scale1)
```

Tasks and I/O

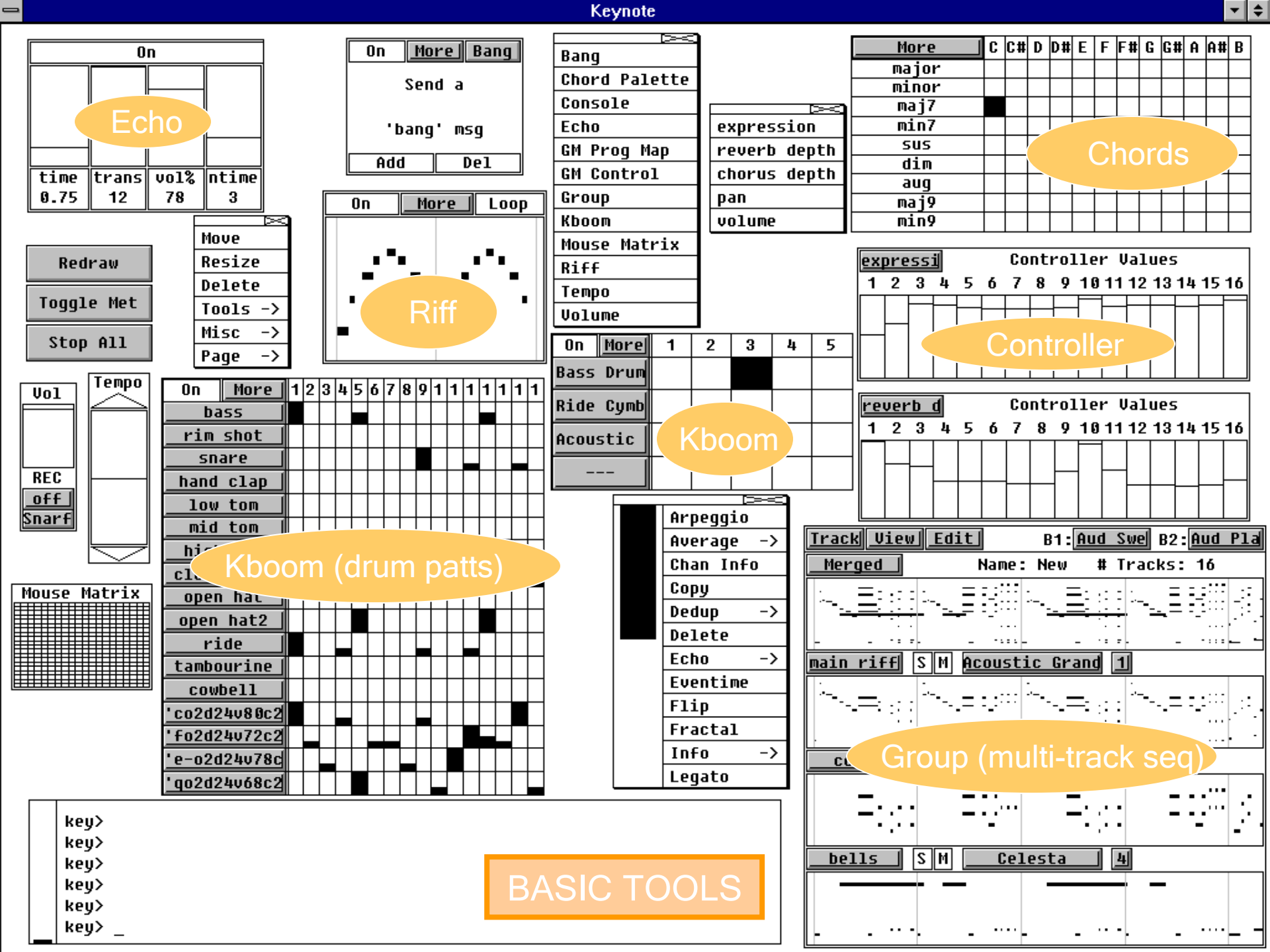
- All tasks are time-shared evenly, interleaved at the interpreted instruction level
- Scheduled MIDI output events are tasks as well, but performance can’t be degraded by other tasks
- MIDI input is always being recorded, available in a global variable for easy and immediate processing
- MIDI, mouse, and console input events can be read from special fifos
- Reading a fifo (with no data waiting) blocks a task
- lock() and unlock() used for exclusion and synchronization

KeyKit - the GUI

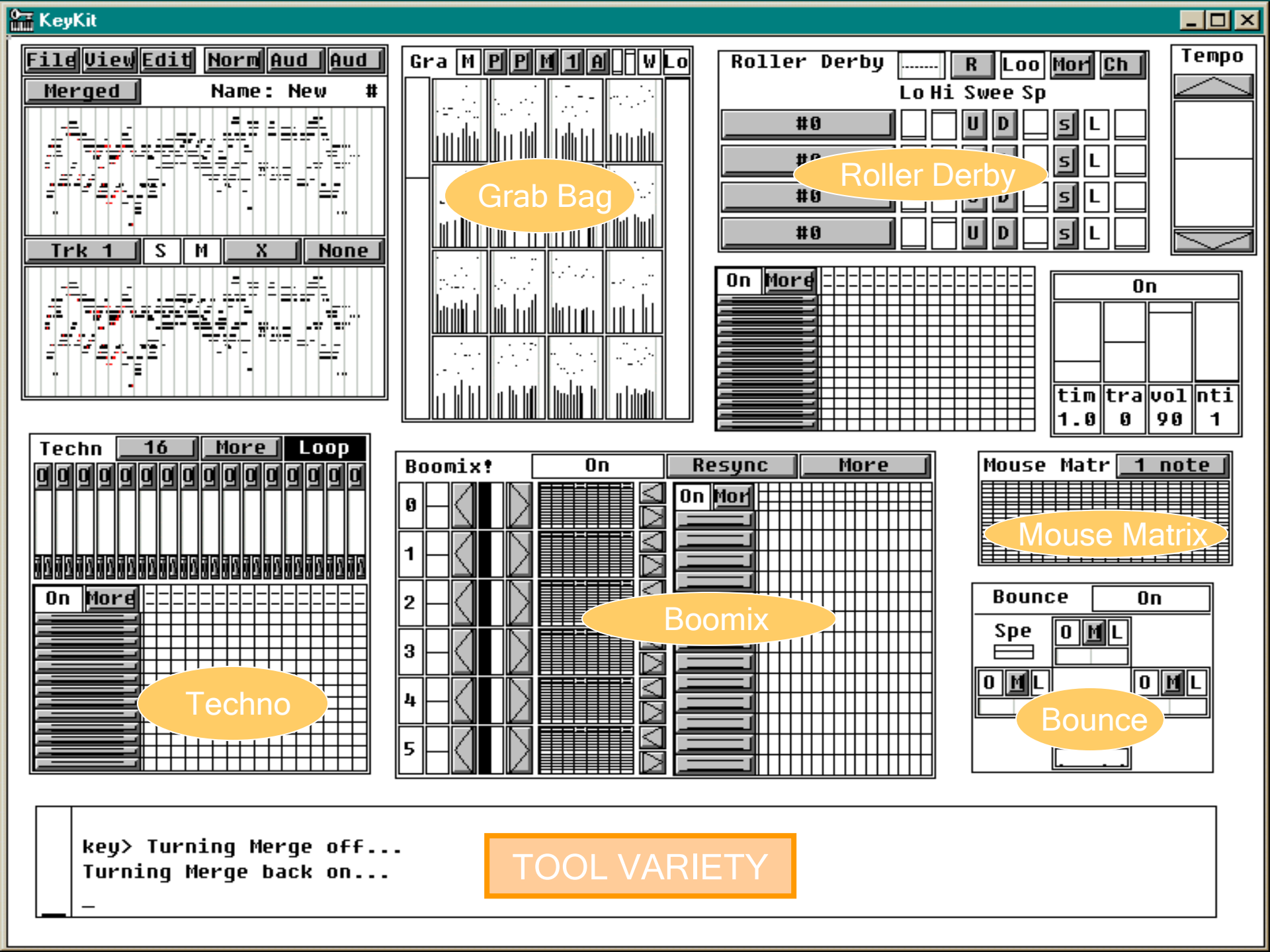
- Completely implemented with Keykit code, even pull-off menus, dragging of windows, window-manager-like operations, etc.
- Each tool is independent, with consistent methods for resizing and inter-tool communication
- Consistent saving/restoring mechanism of individual tools is highly leveraged, used for:
  - Copying between like tools
  - Copy/paste of entire tools
  - Moving tools between “pages”
  - Manipulating of tools within tools
  - Broadcasting of a tool and its contents across a network

Variety of Tools

- Why so many?
  - **Improvisational interactive programming**
- Ball Maze, Bang, Blocks, Boomix, Bounce, Chords, Console, Controller, Echo, Expresso, FourPlay, Fractal, Gene Pool, Ginsu, Grab Bag, Grind, Group, Kboom, Konnect, Loopy, Markov Maker, Monitor, Mouse Matrix, Mousey, Parameters, Party, Peer, Picture This, Prog Change, Quix, Remapper, Riff, RiffRaff, Roller, Sectionalize, Techno, Tempo, Video Decay, Volume, Woolls Bargen, and others


Keynote
On
Echo
time 0.75
trans 12
vol% 78
ntime 3
On More Bang
Send a
'bang' msg
Add Del
Bang
Chord Palette
Console
Echo
GM Prog Map
GM Control
Group
Kboom
Mouse Matrix
Riff
Tempo
Volume
expression
reverb depth
chorus depth
pan
volume
More C C# D D# E F F# G G# A A# B
major
minor
maj7
min7
sus
dim
aug
maj9
min9
Chords
Redraw
Toggle Met
Stop All
Move
Resize
Delete
Tools ->
Misc ->
Page ->
On More Loop
Riff
expressi
Controller Values
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Controller
On More 1 2 3 4 5
Bass Drum
Ride Cymb
Acoustic

Kboom
reverb d
Controller Values
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Vol
REC
off
Snarf
Tempo
On More 1 2 3 4 5 6 7 8 9 1 1 1 1 1 1 1
bass
rim shot
snare
hand clap
low tom
mid tom
open hat2
ride
tambourine
cowbell
'co2d24v80c2
'fo2d24v72c2
'e-o2d24v78c
'qo2d24v68c2
Kboom (drum patts)
Mouse Matrix
Arpeggio
Average ->
Chan Info
Copy
Dedup ->
Delete
Echo ->
Eventime
Flip
Fractal
Info ->
Legato
Track View Edit B1: Aud Swe B2: Aud Pla
Merged
Name: New # Tracks: 16
main riff S M Acoustic Grand 1
bells S M Celesta 4
Group (multi-track seq)
key>
key>
key>
key>
key>
key> _
BASIC TOOLS


KeyKit
File View Edit Norm Aud Aud
Merged
Name: New #
Trk 1 S M X None
Gra M P P M 1 A W Lo
Grab Bag
Roller Derby R Loo Mor Ch
Lo Hi Swee Sp
#0
#0
#0
#0
U D s L
Roller Derby
Tempo
On More
On
tim 1.0
tra 0
vol 90
nti 1
Techn 16 More Loop
On More
Techno
Boomix! On Resync More
0
1
2
3
4
5
On Mor
Boomix
Mouse Matr 1 note
Mouse Matrix
Bounce On
Spe
0 M L
Bounce
key> Turning Merge off...
Turning Merge back on...
TOOL VARIETY

Summary of Unique Strengths

- Phrase (as opposed to note) manipulation supported directly by the language syntax
- Interpreted language makes iterative development a breeze - immediate feedback
- Robust - syntax and execution errors do not bring the system (or even other tasks) down
- Associative arrays - simple but powerful
- Finely-grained multi-tasking gives graceful sharing of CPU, no degradation in realtime scheduling

Summary of Unique Strengths (continued)

- Textual language allows concise expression of:
  - Reusable parameterized utility functions
  - Time-ordered layout of composition
  - Data-driven algorithms
  - Independent algorithms running in parallel
- GUI framework encourages “tool-oriented” approach
- Same language used to implement GUI and all tools, no need to escape to (or learn) C

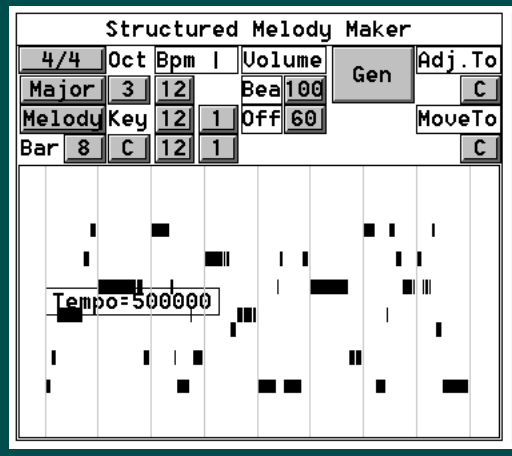
Availability and Resources

- Freely available, with complete source code
- Win95/98/NT/XP and Linux executables
- Mac port exists, but needs lots of polishing
- Download site:
  - **`http://nosuch.com/keykit`**
- Documentation
  - Tutorial, tools reference, language reference, hacking guide
- Mailing list

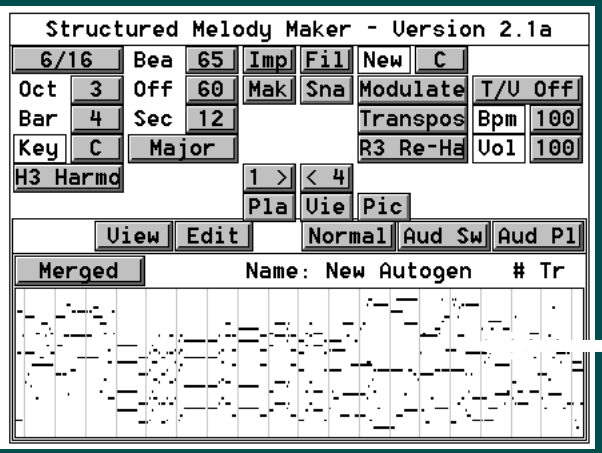
What are other people doing with it?

- Mailing list has 3000 people, not much visible activity, but evidence of lots of experimentation
- Burton Beerman – composition with BodySynth
- Tim Perkis' performance instrument
- David Wooll's "Bargen" tools

Version 1



Version N



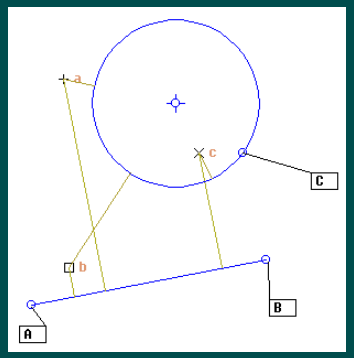
Re-used Group tool

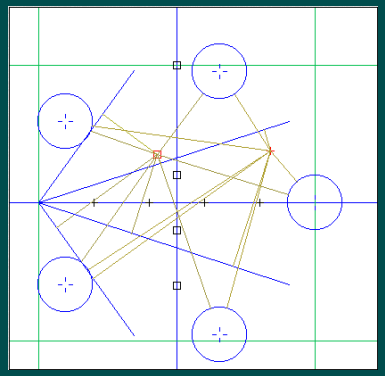
Geomaestro – WOW!

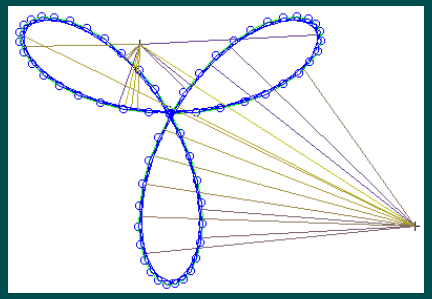
- By Stephane Rollandin, well documented:

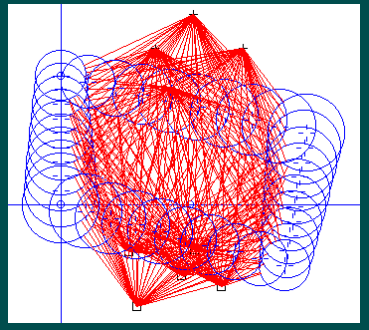
  `http://www.zogotounga.net/GM/paper1.html`

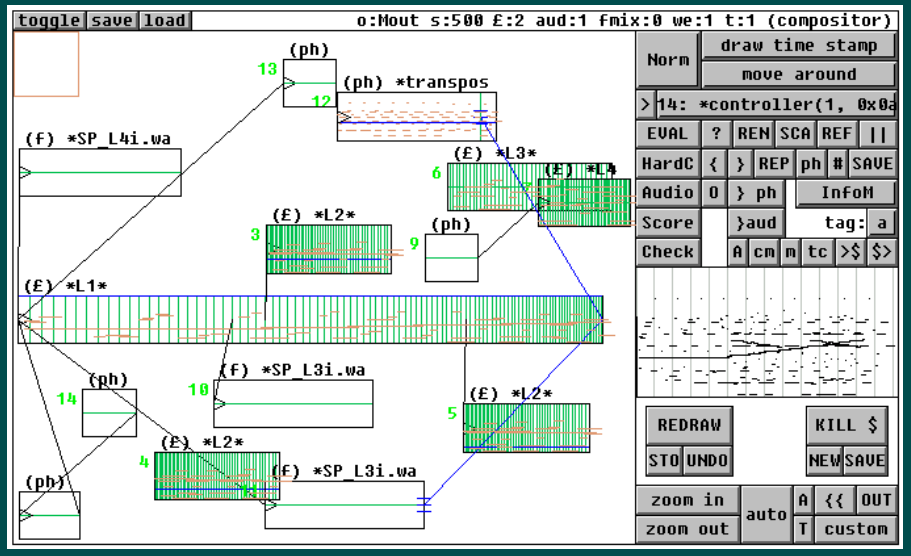
- Chosen (along with KeyKit) in recent Art.Bit collection in Japan: `http://www.art-bit.jp`











What is Tim doing with it?

- Algorithmic - Expresso
- Interactive - Gene Pool, Picture This
- Realtime – Typo, Hoops
- Network - Konnect
- Hardware interfaces
  - Playstation controllers (dance pads, wireless joysticks)
  - QWERTY
  - Relays
  - Webcam
  - iGesture

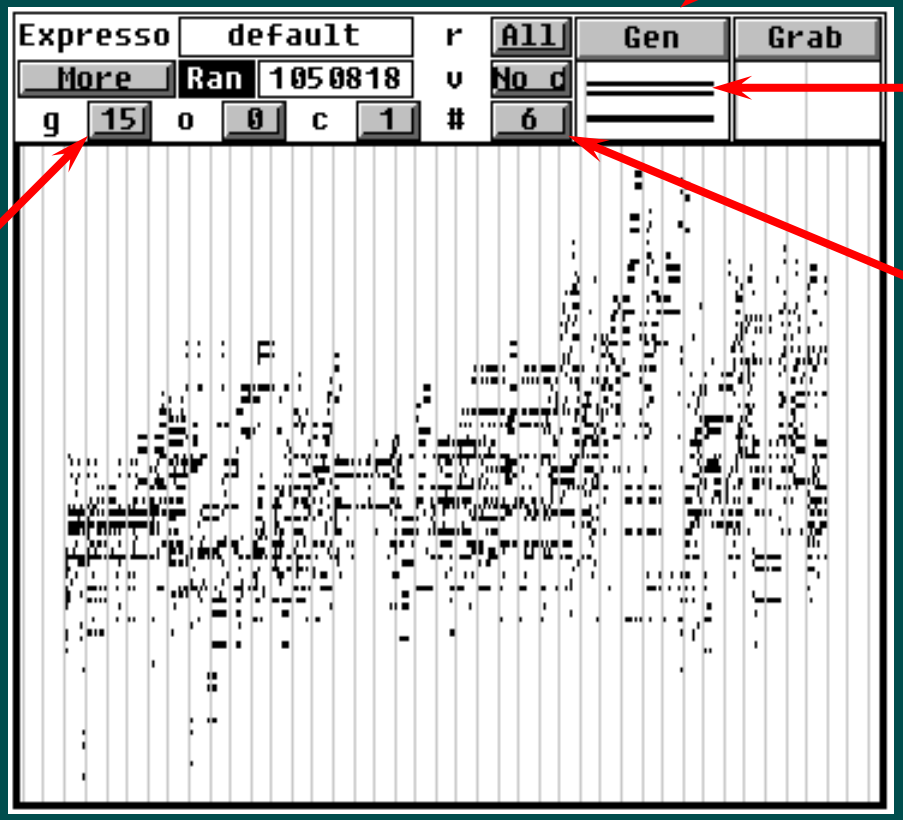
Algorithmic Tools - Expresso

- L-systems fractal generation
- Driven from file of expression transformations
- Starting expression is “X”
- After 10-20 generations, expression is huge
- Substituting note or small phrase for X produces wide variety of results
- Used interactively for Woodstockhausen 2000
- Basis of several Tune Toys on nosuch.com
- Algorithmic Shorts 2001 – “23 Shots of Expresso”

Expresso – default transformations

```
# This is a set of transformations for expresso
A = A+A
A = A|A
A = transpose(A,4)
A = transpose(A,-5)
A = transpose(A,-7)
A = A+transpose(A,12)
A = A+transpose(A,7)
A = A+transpose(A,4)+transpose(A,7)
A = echo(A,4,6)
A = step(A,12)
A = arpeggio(A)
A = shuffle(A)
```

Expresso – GUI interface



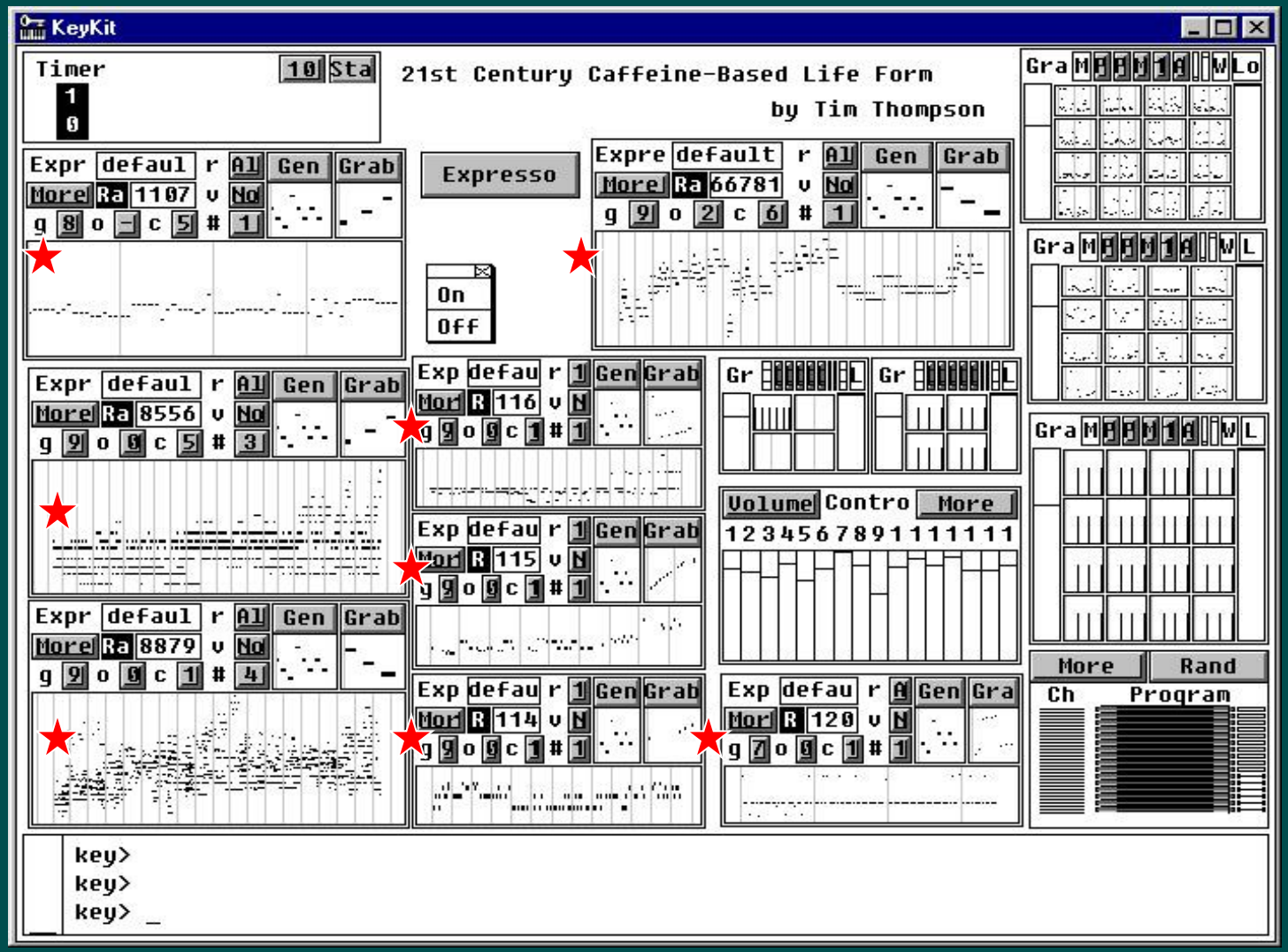
Generates new result

Phrase to use for value of X

# of tracks to generate

# of generations

8 Expresso ★ tools used interactively in "21st Century Caffeine-based Life Form" at Woodstockhausen 2000



Audio examples using Expresso

21st Century Caffeine-based Life Form (Woodstockhausen 2000)

23 Shots of Expresso (Algorithmic Shorts 2001)

Algorithmic music based on digits of PI

- Project of microsound mailing list
- Completely algorithm-based, using digits of PI (3.14…)
- Algorithm and data choices modified as part of composition process
- Two compositions:

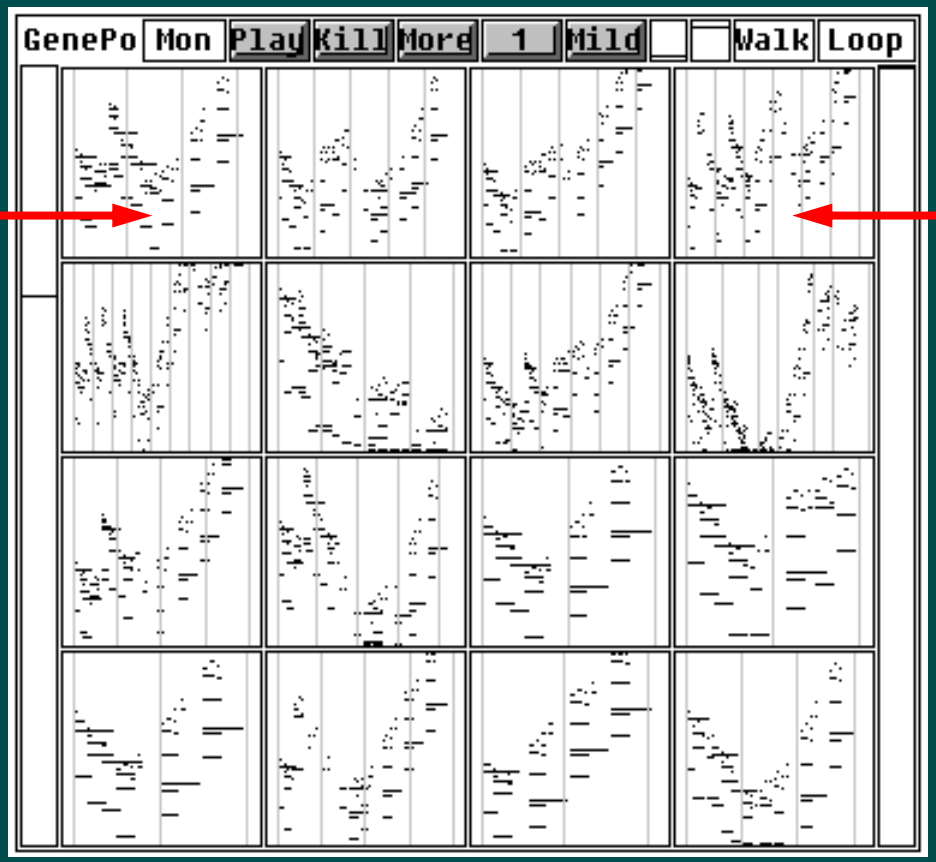
Irrational	tonal version mixed to emphasize tonality
Irrational Too	atonal version, algorithm-chosen Access Virus patches

Interactive Tool - Gene Pool

- Uses small instruction set of musical opcodes
- Each opcode does one thing:
  - Adjust pitch (or time, velocity, duration) of current note
  - Set pitch (or time, velocity, duration) of current note
  - Trigger note (i.e. add current note to the generated result)
- Any phrase can be disassembled into a sequence of musical opcodes that generates it
- Mating of these sequences produces new generations
  - Take half of one, half of the other
  - Shuffle them
  - Etc.

Gene Pool – GUI

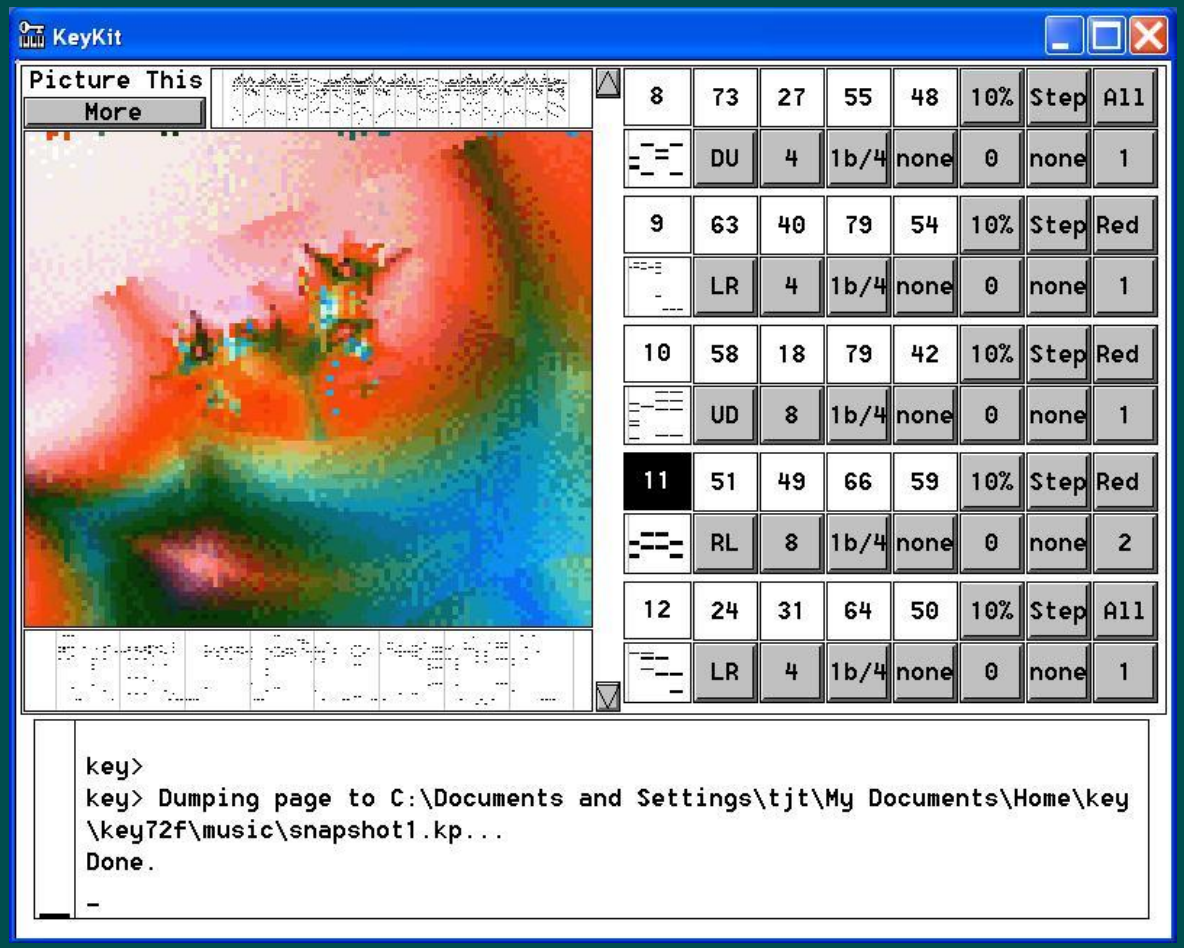
Left-click plays



Right-click kills and fills by mating survivors

Interactive Tool - Picture This

- Use RGB values of an image in various ways



Realtime Tool - Typo

- QWERTY keyboard used as controller
- Based on ability to receive QWERTY up/down events (Windows-specific)
- Most keys used to play notes; holding down control key used to access other functions
- Holding down shift key causes notes to be recorded and looped
- Number keys 0-9 control “sections” – each section retains sound choices and looped notes
- “Oops, I Made a Typo” - Woodstockhausen 2001

Oops, I made a typo (Woodstockhausen 2001)

Realtime Tool - Hoops

- Collaboration with Herb Heinz
- Extension and simplification of Typo
- No quantized loop length, tempo tapped after first loop
- All control through MIDI (no QWERTY)
- Control buttons: REC, UNDO, CHAN, TAP, QUANT, TRANS, NUDGE, MUTE, BEAT, COPY
- Some controls are chorded with notes to provide values:
  - CHAN+’D’ sets channel to 3
  - TRANS+’E’ transposes current channel by 4
- Holding down ‘C’ applies it to all channels
  - TRANS+’C’+’E’ transposes all channels by 4

Network Tool - Konnect

- Uses Linux and Windows-specific hooks
- Broadcast of MIDI data in realtime
- Simple text-chat
- Two-way resynchronization with 4-beat delay
  - Each side continuously transmits and receives
  - Received data is resynchronized to local timing
  - What you hear during a given 4 beats is what the other side played in response to the 4 beats you just finished several beats ago.
- Linux server runs KeyKit process that serves as proxy/broadcaster, >2 clients can connect and jam simultaneously

Playstation Controller Interfacing

- PS2-to-USB interfaces, not all created equal
- EMS USB 2-port interface works well (available at www.levelsix.com or www.gocybershop.net)

- Windows driver makes dance pad look like buttons on a joystick, works with standard multimedia API
- Able to connect 4 interfaces (8 pads) simultaneously
- Pads and interfaces have been surprisingly reliable

KeyKit hooks for Playstation devices

- It’s a generic joystick interface – anything with a Windows driver that looks like a joystick will work
- Windows events (and/or polling) generate keykit events
- Looks like a fifo in the KeyKit language, just like mouse/console/midi/network inputs
- Good responsiveness
- Order of devices is non-deterministic, need to establish order interactively, if order is important

Wireless joysticks

- Anything that looks like a joystick becomes a music controller
- Logitech wireless joysticks for the Playstation work well (with EMS USB2 interface), and have natural layout of buttons for performance
- 10 buttons + 4-button joypad + 4 axis of analog joystick control
- Both button-down and button-up events can be used

Video input

- Windows-specific feature, uses DirectShow API
- Grabs samples of video, provides averaged low-res (adjustable) grid of RGB values

Dance Pads

Dance Pads at Burning Man 2002

Happy Feet – a composition for Dance Pads

- Performance at Woodstockhausen 2002
- Bach's "Jesu, Joy of Man's Desiring" provides notes
- Music broken into snippets by time or attacks
- Snippets assigned across all 4 dance pads, in sets
- Advancing through sets is controlled by select button
- 4 sections in performance

Dance Pad UI

- 8 main buttons play notes or snippets
- Select and Start buttons, followed by a main button, perform control functions
- Pressing Select or Start multiple times (2 or 4) is used to perform less-common functions
- Each of 4 pads is independent and usually identical, some functions affect one pad, some affect all pads
- People try the Select and Start buttons without knowing what they do – need to “hide” functions more

Dance Pad Controls

SELECT

NEXT RHYTHM	MORE NOTES	PATCH CHANGE
ARPEGGIO		PATCH RANDOM
ADVANCE	LESS NOTES	PATCH TYPE CHANGE

START

CLEAR LOOP	OCTAVE UP	RECORD ON/OFF
SHORTER DURATION		LONGER DURATION
SOFTER	OCTAVE DOWN	LOUDER

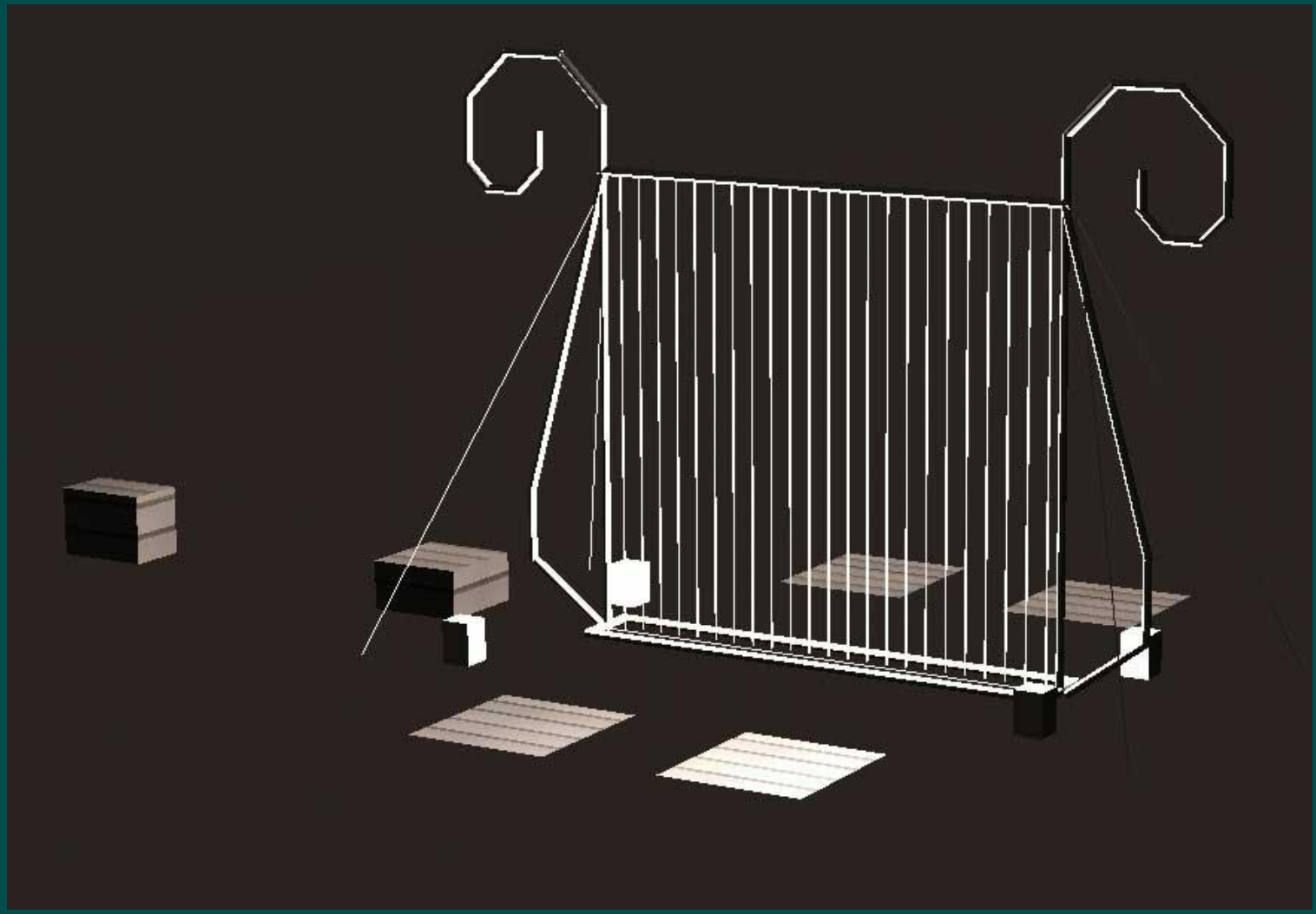
SELECT 2

RHYTHM ON/OFF		PATCH RESET
DRUMS ON/OFF		

START 2

RESET ALL	FASTER	FADE ON/OFF
SHORTER LOOP		LONGER LOOP
	SLOWER	

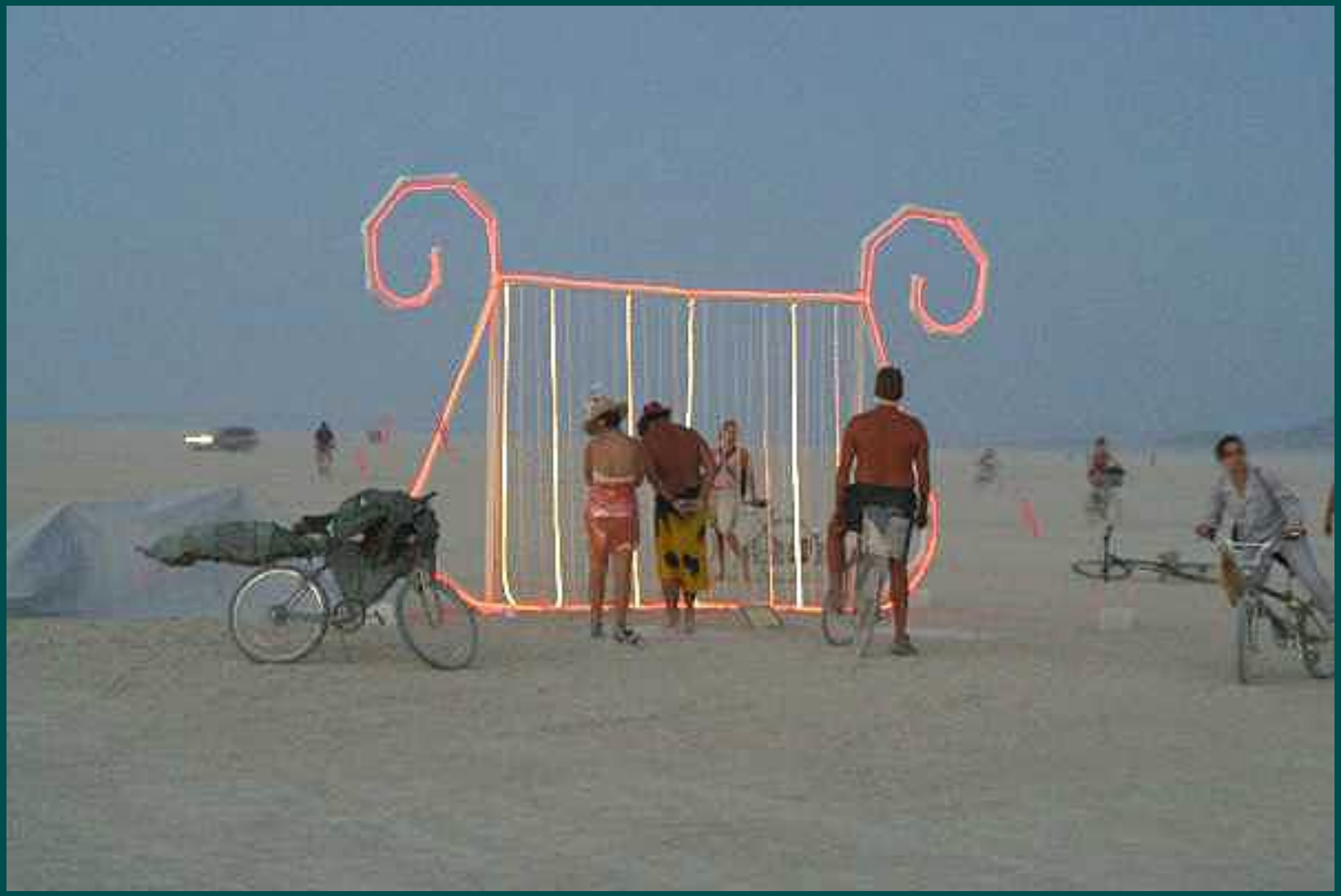
Dance Pads at Burning Man 2003



Dance Pads at Burning Man 2003

Click on image
to see video
of lyre in action

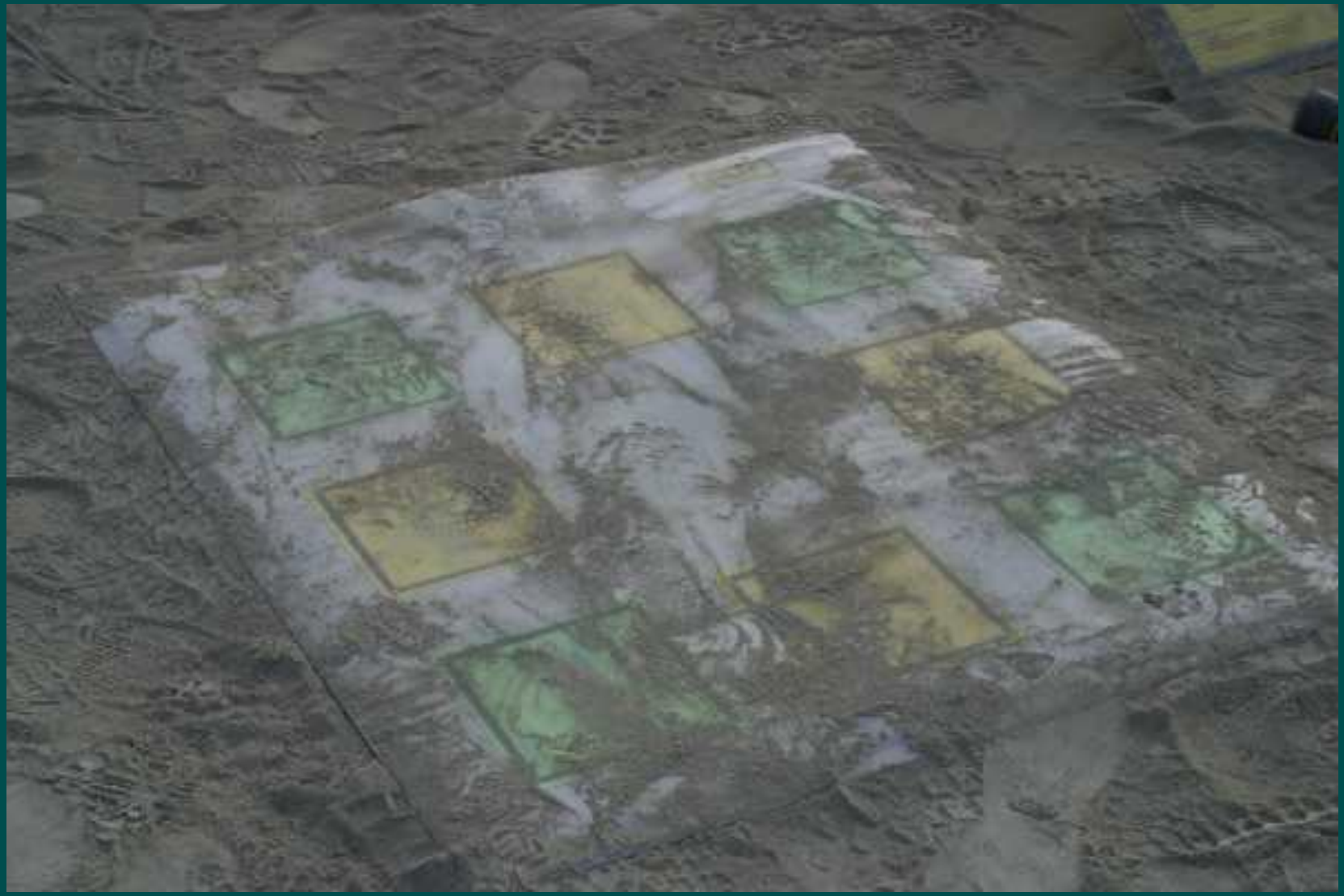
Dance Pads at Burning Man 2003



Dance Pads at Burning Man 2003

Dance Pads at Burning Man 2003

Dance Pads at Burning Man 2003



Wireless QWERTY + Dance Pad

- NoSuch Music at 26Mix, San Francisco dorkbot, and San Jose Works Gallery

dork @ dorkbot

Radio Free Quasar

- Burning Man 2004
- Antique radio with computer inside
- Python-based VST host
- Randomized VST parameters
- Controlled by Big Silver Knob (Griffin Technologies)
- Audio-controlled laser display

iGesture pad

- **`www.fingerworks.com`**
- Inexpensive multi-touch pad
- Excellent responsiveness
- KeyKit interface to event stream
- Event data: x, y, proximity, eccentricity, orientation, contacts, device, finger, hand, xvelocity, yvelocity
- Multiple pads can be used simultaneously

Visual Music experiments

- Python used for OpenGL support
- KeyKit is interface to input devices (MIDI, iGesture)
- Messages sent over TCP/IP to Python process
- MIDI sliders and buttons control graphics parameters
- MIDI from drummer triggers graphics
- Text typed interactively is used as graphics
- Words typed interactively can immediately search clip-art database whose images are then used as graphics
- Used in dud (improvised art ensemble) - **www.dudland.com**

Etcetera

- Simple OSC (Open Sound Control) support

Availability and Resources

- Freely available, with complete source code
- Win95/98/NT/XP and Linux executables
- Mac port exists, but needs polishing
- Documentation
  - Tutorial, tools reference, language reference, hacking guide
- Mailing list
- Source Code: **`https://github.com/nosuchtim/keykit`**
- Questions: **`me@timthompson.com`**

KeyKit

A Scripting Language and GUI Interface for MIDI

Slides can be found at
https://timthompson.com/talks

Tim Thompson
me@timthompson.com
https://timthompson.com