

KeyKit

Tim Thompson
tjt@nosuch.com
<http://nosuch.com>

Outline

- History
- Language and GUI
- Algorithmic Tools
- Interactive Tools
- Realtime Tools
- Hardware Interfacing
- Combinations thereof

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 current version is complete 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

Language Features

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

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
$$\text{ph} = \text{ph1} + \text{ph2}$$
- Parallel merging
$$\text{ph} = \text{ph1} \mid \text{ph2}$$
- Removing notes
$$\text{ph} = \text{ph1} - \text{ph2}$$
- Matching notes
$$\text{ph} = \text{ph1} \& \text{ph2}$$
- Nth note
$$\text{ph} = \text{ph1} \% n$$

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 }
```

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, Espresso, 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 Bargaen, and others

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 U D S L

#0 S L

#0 U D S L

#0 U D S L

On More

On

tim	trav	vol	nti
1.0	0	90	1

Tempo

Techn 16 More Loop

On More

Techno

Boomix! On Resync More

0 On Mor

1

2

3

4

5

Boomix

Mouse Matr 1 note

Mouse Matrix

Bounce On

Spe O M L

O M L O 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)

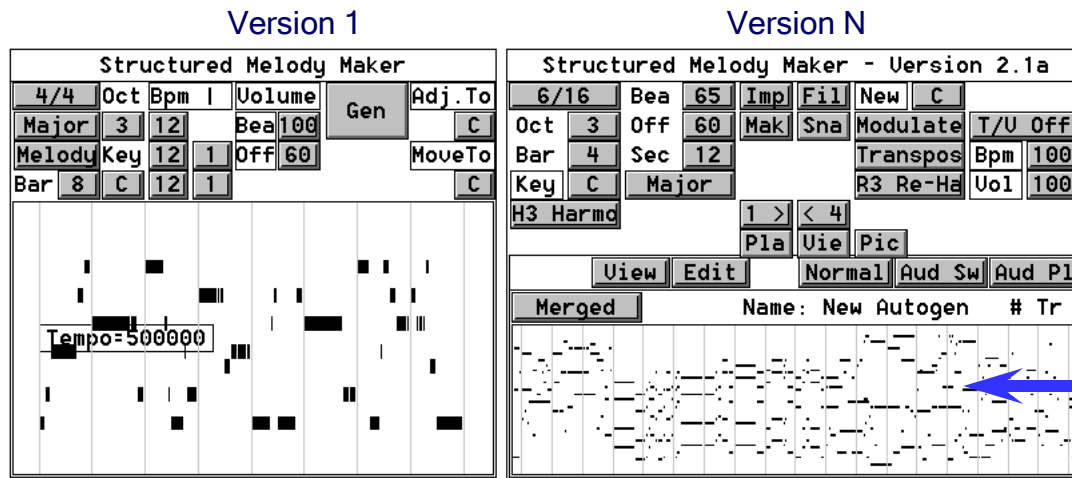
- Textual language allows concise expression of:
 - Reusable parameterized utility functions
 - Time-ordered layout of composition
 - Data-driven algorithms
 - Independent algorithms running in parallel
- GUI framework encourages “tool-oriented” approach
- Same language used to implement GUI and all 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, little visible activity, but evidence of lots of experimentation
- Burton Beerman – composition with BodySynth
- David Wooll’s “Bargen” tools

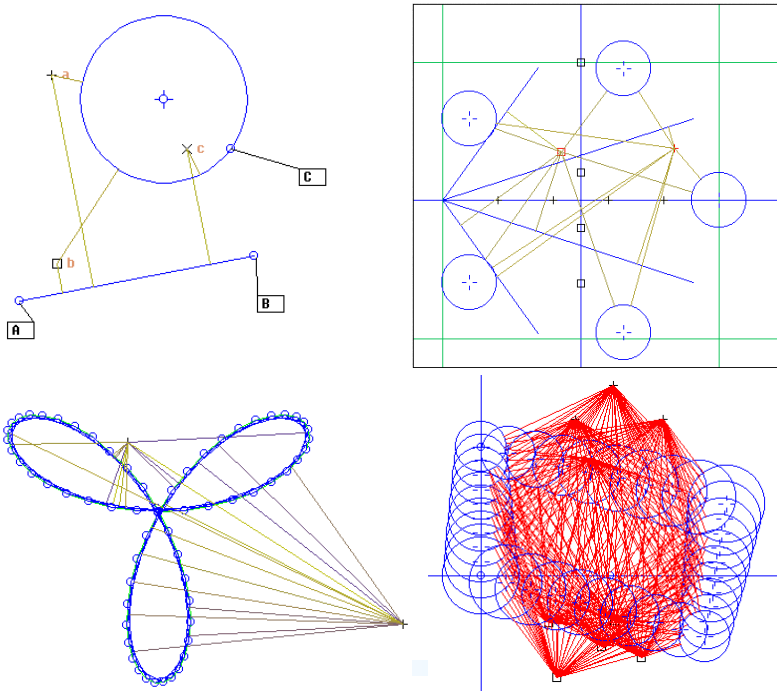


Geomaestro – WOW!

- 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>



toggle save load o:Mout s:500 E:2 aud:1 fmix:0 we:1 t:1 (composer)

Norm	?	REN	SCA	REF	
> 14: *controller(1, 0x0a					
EVAL	{	}	REP	ph	# SAVE
HardC	0	}	ph		InfoM
Score		}	aud	tag:	a
Check	A	cm	m	tc	>\$ \$>

REDRAW KILL \$
STO UNDO NEWSAVE
zoom in auto A << OUT
zoom out T custom

What is Tim doing with it?

- Algorithmic - Espresso
- Interactive - Gene Pool, Picture This
- Realtime – Typo
- Network - Konnect
- Hardware Controllers
 - Playstation (dance pads, wireless joysticks)
 - Video

Algorithmic Tools - Espresso

- 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 performance
- Basis of several Tune Toys on nosuch.com

Espresso – default transformations

```
# This is a set of transformations for espresso
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

The screenshot shows the Expresso GUI interface. The top panel contains several controls: a 'More' button, a 'Ran' field with the value '1050818', a 'g' field with '15', an 'o' field with '0', a 'c' field with '1', a 'r' dropdown menu set to 'All', a 'u' dropdown menu set to 'No d', and a '#' field with '6'. To the right of these are 'Gen' and 'Grab' buttons. The main area of the GUI is a large window displaying a complex, abstract pattern of black and white pixels, resembling a fractal or a complex signal waveform. Red arrows point from text labels to specific parts of the interface: 'Generates new result' points to the 'Gen' button; 'Phrase to use for value of X' points to the 'u' dropdown menu; '# of generations' points to the 'g' field; and '# of tracks to generate' points to the '#' field.

Generates new result

Phrase to use for value of X

of generations

of tracks to generate

8 instances (★) of Espresso tool used interactively in “21st Century Caffeine-based Life Form” at Woodstockhausen 2000

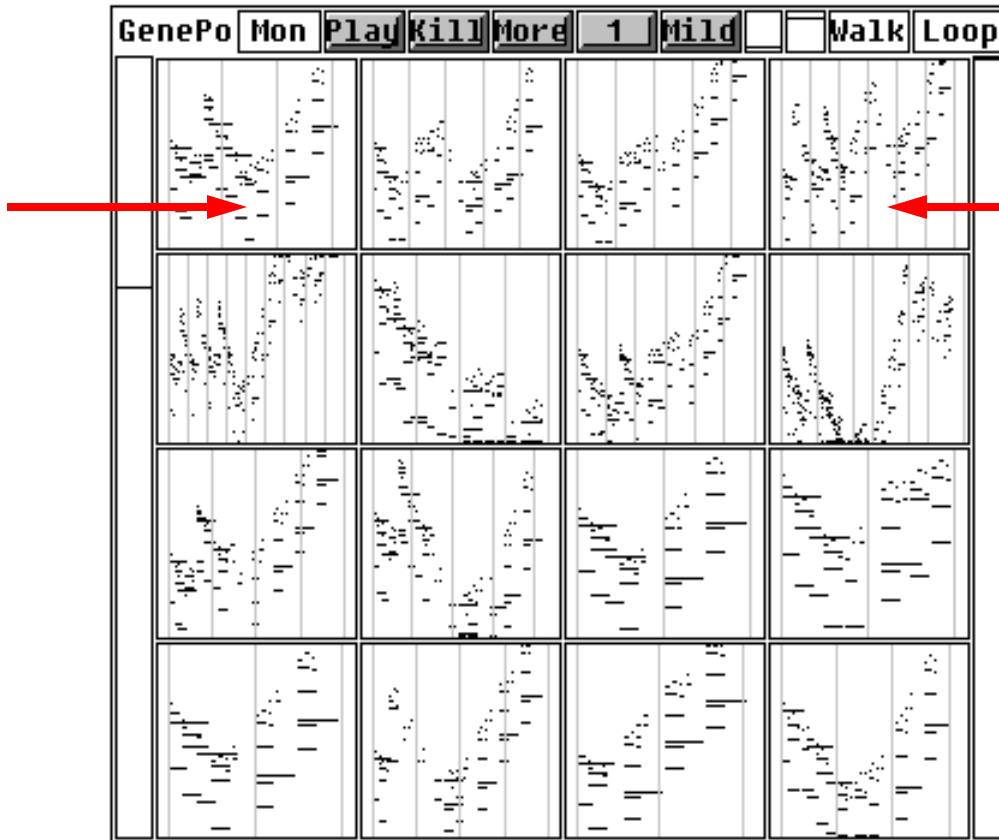
The screenshot displays the KeyKit software interface, titled "21st Century Caffeine-Based Life Form by Tim Thompson". The interface is densely packed with various control panels and data displays. At the top left, there is a "Timer" panel with a value of 10 and a "Sta" button. Below it, a large panel contains an "Espresso" button and a "More Ra 66781" display, with a red star (★) marking a specific area. To the right, there are several "Gram" panels, each with a grid of small icons. In the center, there are two "Exp default" panels, each with a "More Ra" display (values 8556 and 8879) and a red star. Below these, there are more "Exp default" panels, one with a "More Ra 120" display and a red star. On the right side, there is a "Volume Control" panel with a "More" button and a display showing the numbers 1234567891111111. At the bottom right, there is a "More Rand" panel with a "Ch Program" display. The bottom left corner shows a command prompt with "key>" prompts. The interface is highly interactive and complex, reflecting the "21st Century Caffeine-based Life Form" project.

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

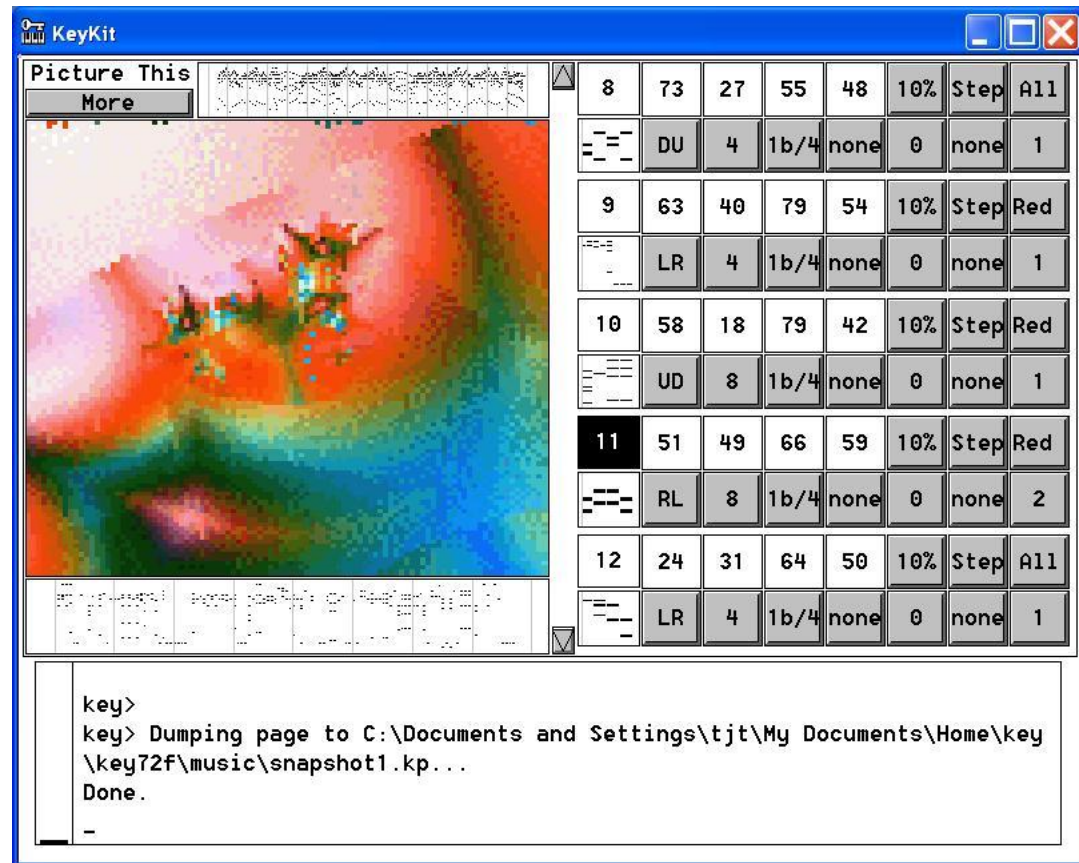
Left-click
plays



Right-click
kills and fills
by mating
survivors






Interactive Tool - Picture This

- Use RGB values of an image in various ways



The screenshot shows the KeyKit software interface. The main window is titled "KeyKit" and contains a central image of a bird. To the right of the image is a table with 11 rows and 8 columns. The table contains key signatures and their corresponding symbols. Below the table is a text area with the following text:

```
key>  
key> Dumping page to C:\Documents and Settings\tjt\My Documents\Home\key  
\key72f\music\snapshot1.kp...  
Done.  
-
```

8	73	27	55	48	10%	Step	All
	DU	4	1b/4	none	0	none	1
9	63	40	79	54	10%	Step	Red
	LR	4	1b/4	none	0	none	1
10	58	18	79	42	10%	Step	Red
	UD	8	1b/4	none	0	none	1
11	51	49	66	59	10%	Step	Red
	RL	8	1b/4	none	0	none	2
12	24	31	64	50	10%	Step	All
	LR	4	1b/4	none	0	none	1

Realtime Tool - Typo

- Typing keyboard used as musical controller
- Based on ability to receive console up/down events (Windows-specific addition to keykit)
- 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
- Used for Woodstockhausen 2001 performance

Network Tool - Konnect

- Uses Linux and Windows-specific hooks
- Broadcast of MIDI data in realtime
- Simple text-chat
- Two-way synchronization 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 joystick/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

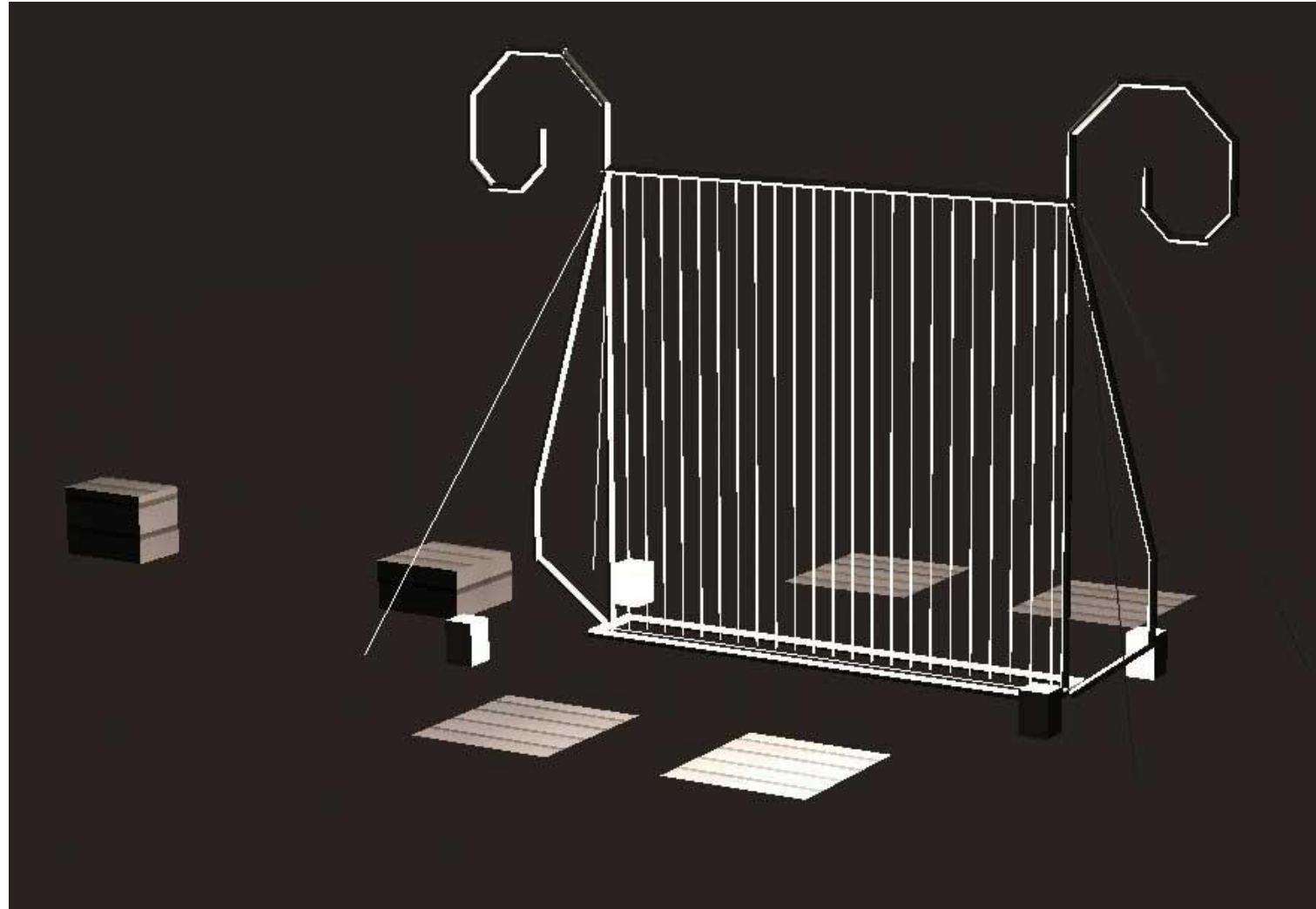
Dance Pads



Dance Pads at Burning Man 2002



Dance Pads at Burning Man 2003



Happy Feet – a composition for the 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	

Video input

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

Combining things in performance

- Ball Maze
 - Mouse
 - Wireless Joystick
 - Video

Availability and Resources

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