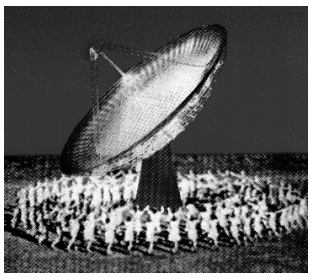
Siren: Sound and Music Tools for Squeak and VisualWorks



Stephen T. Pope
Center for Research in Electronic Art Technology (CREATE)
U. of California, Santa Barbara (UCSB)
`stp@create.ucsb.edu`

Outline

- **Siren Background**
- **Siren Models**
- **Siren Sound Synthesis & Control**
- **Siren Databases & Interfaces**
- **Siren Applications**
- **Siren Platform Mobility**
- **Demonstration**
- ***All source on the net, for more info, see: http://www.create.ucsb.edu/Siren***

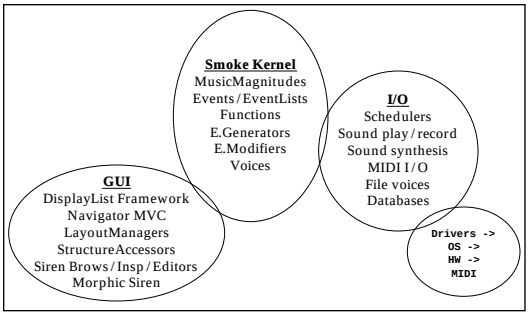
Siren/MODE Background

- **(I reimplement and rename it periodically.)**
- **SmallSong (1984), DoubleTalk (1986), HyperScore ToolKit (1989), MODE (1992), *Siren* (1998)**
- **The Smoke Representation Language**
- **Siren I/O: Voices & Drivers**
- **Siren Databases & GUIs**
- **Siren Applications**
- **Publications: *Musical Signal Processing*, SqueakBook2**

Why Siren?

- **Provide a Flexible and Extensible Environment for Musicians**
- **Address Tasks of Composition, Realization, and Production**
- **Support Working with *Sound***
- **Provide Extensible GUIs**
  - **No Need to Support "Historical" Music**

Siren Architecture



Squeak Smalltalk (Tangent)

- **VI based on PARC Smalltalk-80 (v1)**
- **Developed at Apple, WDI, etc.**
- **Two books + CD-ROMs from Prentice-Hall**
- **New VM Written in Primarily Smalltalk(!) and translated to C**
- **Garbage Collector in Smalltalk**
- **Morphic GUI Framework**
- **Network, data streaming, plug-ins, multimedia**
- **Ported to Mac, MS-Win, UN*X, WinCE, hand-held, stand-alone, ...**

Smoke Music Representation

- **Smoke Consists of Classes for:**
  - **MusicMagnitudes**
  - **Events and EventLists**
  - **Functions of 1-n variables**
  - **Voices, Ports, and Drivers**
  - **EventGenerators**
  - **EventModifiers**

Music Magnitude Models

- **Abstractions**
  Chronos, Chroma, Ergon, Positus
- **Representation Classes**
  Pitch, Duration, Loudness
- **Implementation Classes**
  HertzPitch, SymbolicP, RatioP, MIDIP
  (Pitch value: 'c3') == ('c3' pitch)
  (Amplitude value: #mf) == (#mf ampl)
- **Mixed-mode Arithmetic**
  (#f4 pitch + 80 Hz)
- **Extended MusicMagnitudes**
  - **ConditionalDuration, Sharpness**

Events

- **Events are just Property Lists**
  (with [optional] durations but *no start times*)
- **There are *Verbose* and *Terse* Formats**

```
(DurationEvent dur: 250 msec voice: #flute)

((880 Hz, 250 msec, (#voice -> #flute),
                0.7071 ampl) accent: #sfz)

anEvent color: #green; shape: #round
```

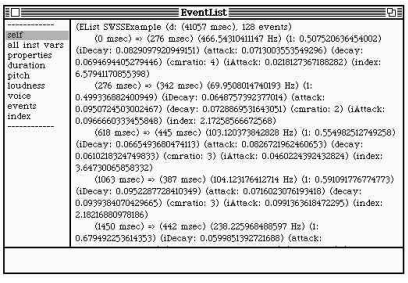

Event Lists

- **List of (Delay -> Event) Associations**
  The delay is the event's start time relative to the start of the list (i.e., it's a duration)
- **Methods to Add, Remove, Filter Events**
- **Methods to "perform" Events on their Voices**
- **Verbose and Terse Formats**
- **Editors, Browsers, Databases**

Event List Examples

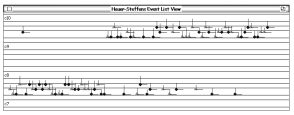
```
[(EventList newNamed: #Chord1)
      add: ((1/2  beat), "d3"  pitch,  "mf" ampl)  at: 0;
      add: ((1/2  beat), "fs3"  pitch, "mf" ampl)  at: 0 … ]

(0     => 440 Hz, 250 msec, -3 dB) ,
((1/4) => (471 Hz, 0.37 beat, #ff))

EventList named: 'phrase1'
   fromSelectors: #(duration: loudness: phoneme:)
   values: (Array
     with: #(595 545 545 540 570 800 540)
     with: #(0.8 0.4 0.5 0.3 0.2 0.7 0.1)
     with: #(dun kel kam mer ge sprae che))
```


Event List Inspector



EventGenerators

- **Models of "Middle-Level" Structures**
- **Support "Composition by Refinement"**
- **Provide "Constant Performability"**
- **An Extensible Framework for Composition**
- **Chords, clouds, clusters, ostinati**



EventModifiers

- **Apply a function to an event list's event properties**
- **Do so eagerly (at declaration time) or lazily (at performance time)**
- **EMods can be composed**
- **Rely on Smoke Function Models**
- **N-dimensional functions of tempo**
- **Time-scaled application**

Siren Performance: Voices

- **Events or EventLists have "abstract" Properties, and Voices**
- **A Voice is a *Property-to-Parameter Mapper* (e.g., HzPitch -> MIDIPitch to play a Hz-oriented score on MIDI, or SymbolicLoudness -> MIDIVelocity)**
- **Voices can have Ports and Devices, or formatted I/O Files/streams**

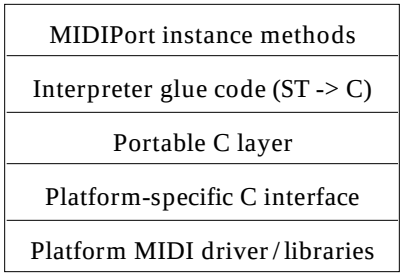
The Siren Scheduler

- **Class Scheduler and sole instance *Schedule* can have clients registered to receive the *scheduleAt:* message**
- **They may do something in response to it, and may answer a time when they wish to be scheduled again.**
- **Smalltalk-only scheduler is pretty fast!**

MIDI I/O

- **Instance of MIDIPort calls primitives**
- **Their glue code is written in ST80 and translated into C; it calls driver fcns.**
- **The portable driver layer implements the module defined by the primitives**
- **Several Back-end Driver Interfaces**
- **Relation to PortAudio & PortMIDI**

MIDI I/O Layers



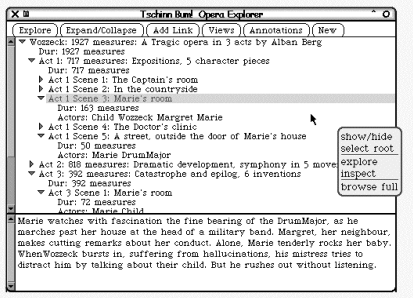
Persistency and Siren Objects

- **Paleo Project: Storage, feature extraction, and queries on MIDI data, scores, instrumental performance, sound effects, spoken voice, etc.**
- **Feature extraction using NOLib (MATLab), Backtalk constraints, FASTLab, ReBa, LPC, & pvoc**
- **Back-ends**
  - **SMS = Siren MinneStore ObjectSets**
  - **Gemstone models for Smoke objects**
  - **MySQL storage of Smoke feature vectors**
- **Front-ends**
  - **Browsers and collectors**

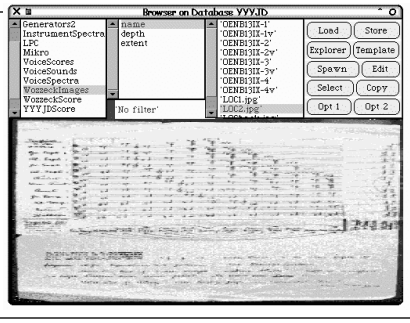
Siren Database Applications

- **Wozeck image database**
- **Paleo harmonic queries**
- **NOLib performance extraction**
- **Composition database**
  - ***YYYJD***
  - ***FourMagicSentences***
  - ***Sword***

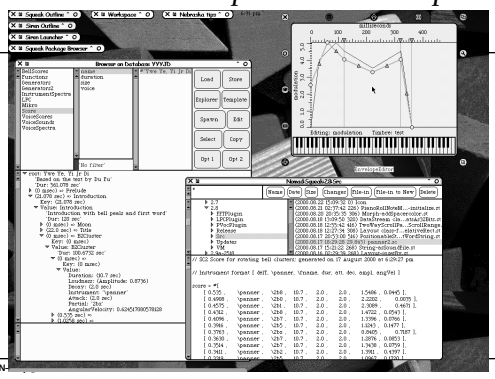
Wozzeck Opera Structure Editor



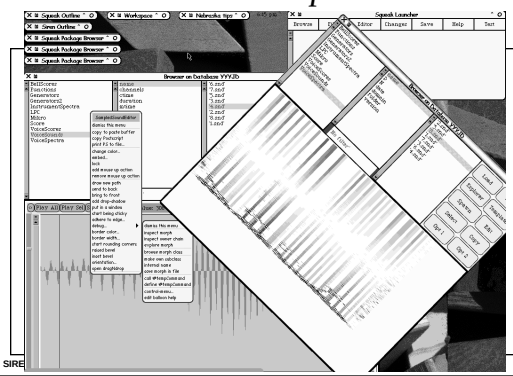
Wozeck Score Image Database



Siren Score and SuperCollider Output



Siren Editors in Morphic



Siren Performance

- **Supports scores (i.e., 20s) of real-time synthesis voices on lap-tops**
- **Full-bandwidth MIDI I/O**
- **Complex structure-editing GUIs (under development)**
- **Smalltalk-level scheduler can flood MIDI (msec-level timing)**
- **Siren 3.0 is ~ 350 Classes, 4000 Methods**

Portable Smalltalk

- **Siren core (Smoke, drivers, scheduler, etc.) are "platform independent" between Squeak and VisualWorks**
- **Use of *FlavorCompatibility* and *ExceptionCompatibility* classes as in MinneStore (mildly extended)**
- ***DBPortability* = WIP**
- **GUIs: no hope…**

Siren Interfaces

- **Squeak (Plug-ins)**
  - **R-T synthesis classes**
  - **OpenSoundControl/SuperCollider**
  - **MIDI/Sound I/O**
  - **LPC, pvoc plug-ins**
- **VisualWorks (CORBA/DLLCC)**
  - **ReBa analysis/synthesis**
  - **CREATE Auralizer**
  - **NOLib feature extraction**

Future (1999, as seen from 1998)

- **Squeak clients on many platforms talk to DB "Stones" at CREATE**
- **Squeak translated synthesis, DSP, mixing (should rival C-based SWSS)**
- **Scalable tool/instrument—central resources and distributed access**
- **DRIVE, Creatophone, Paleo, Time-Machine, and HPDM Projects**

Future (2000+ as seen from 1998)

- **ATM-based wide-area sound/music computing with end-user nodes at many bandwidths (GIOP/ATM to 10T)**
- **DB queries to and operations on very large score/sound databases**
- **Poly-channel I/O and pluriphonic projection from synthesis SW or disks**
- **New tool paradigm**

Siren