

s.c.r.e.a.m.
**System for the Creation of Random
Electronic Adaptive Music**

THE ORANGE LUNCHBOX BRIGADE
Ryan Curtin and Chad D. Kersey

Georgia Tech School of Electrical and Computer Engineering
ECE4884-L04: Dr. David V. Anderson

Project Overview

Project Overview

» Project Overview

» Technical Abstractions

» Project Objectives

» The Brain

» The Instruments

» The Mixer

» Interprocess Communication

Project Timeline

Current Status

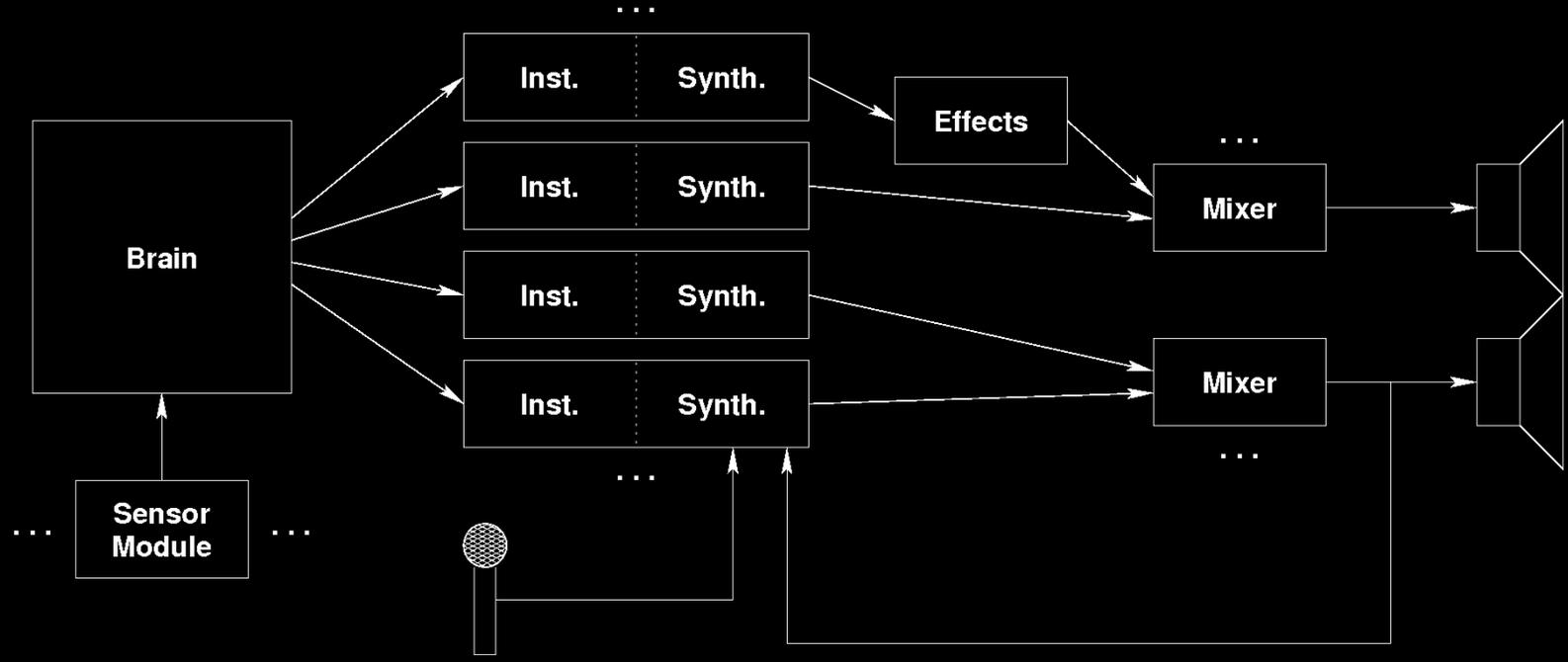
The Future

Questions and Comments

- **What** : A fully automatic music synthesis system influenced by its environment
- **Why** : To advance the frontiers of electronic creative expression
- **Who** : Electronic musicians; potential listeners
- **How** : C/C++ software applications
- **When** : Now.

Technical Abstractions

- Project Overview
 - » Project Overview
 - » **Technical Abstractions**
 - » Project Objectives
 - » The Brain
 - » The Instruments
 - » The Mixer
 - » Interprocess Communication
-
- Project Timeline
-
- Current Status
-
- The Future
-
- Questions and Comments



Top-level abstraction of the s.c.r.e.a.m.

Project Objectives

- Project Overview
- » Project Overview
- » Technical Abstractions
- » Project Objectives
- » The Brain
- » The Instruments
- » The Mixer
- » Interprocess Communication

- Project Timeline

- Current Status

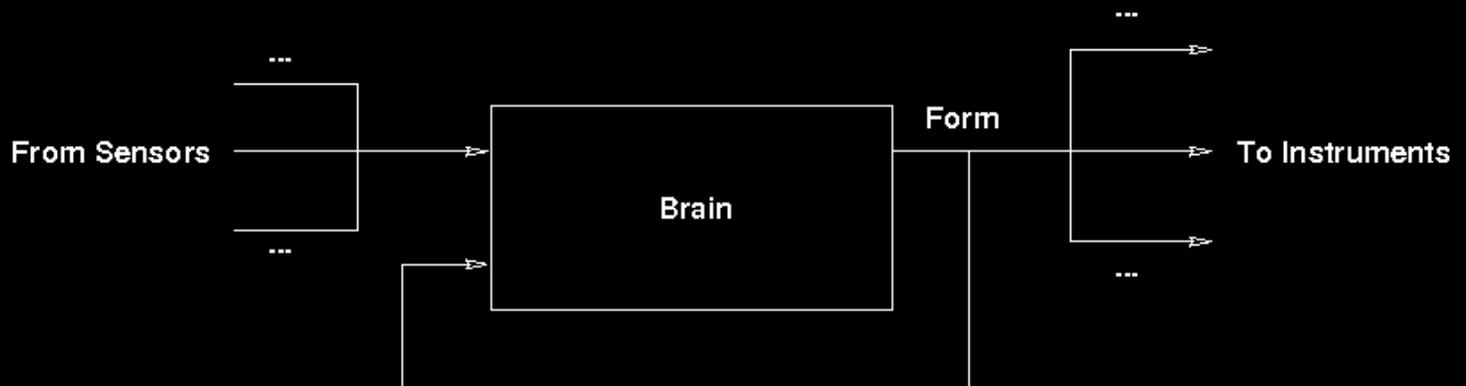
- The Future

- Questions and Comments

- Conceptual objectives
 - Modular framework
 - Freedom to generate any style of music
- Technical objectives
 - Portable C/C++ codebase
 - Conformity to C99 and ISO C++ standards
 - Each component is its own process

The Brain

- Generates and passes a `form` structure to each instrument
- Controlled by text files defining event probabilities
- Environmental input modifies probabilities
- Feedback means new `form` objects can evolve from old ones



Project Overview

» Project Overview

» Technical Abstractions

» Project Objectives

» The Brain

» The Instruments

» The Mixer

» Interprocess Communication

Project Timeline

Current Status

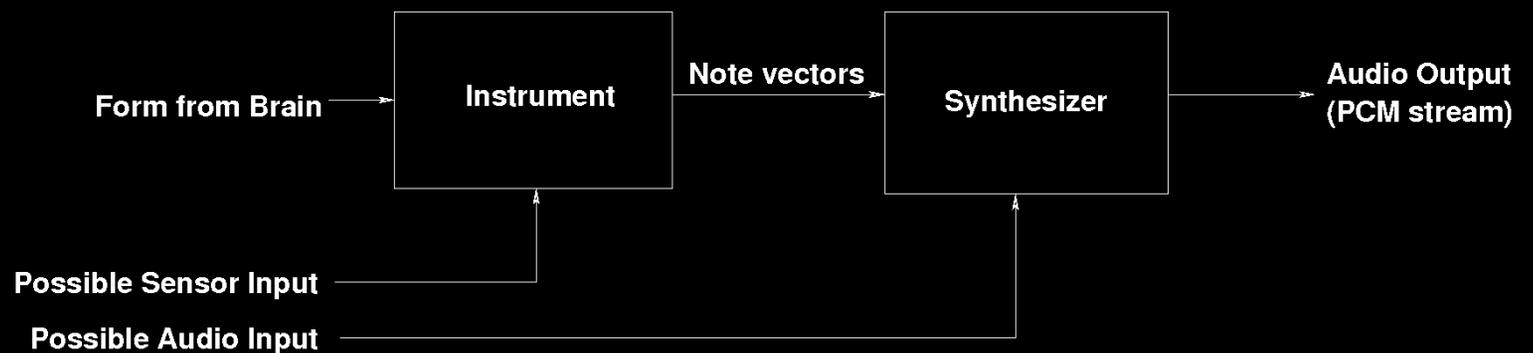
The Future

Questions and Comments

The Instruments

- Project Overview
 - » Project Overview
 - » Technical Abstractions
 - » Project Objectives
 - » The Brain
 - » **The Instruments**
 - » The Mixer
 - » Interprocess Communication
- Project Timeline
- Current Status
- The Future
- Questions and Comments

- Using the given `form` as input, each instrument decides what to play
- Control the synthesizer with pitches, timings, intensities, etc.
- Synthesizer uses collections of PCM samples
- Output to the mixer is a raw PCM stream delivered through JACK



The Mixer

- Project Overview
 - » Project Overview
 - » Technical Abstractions
 - » Project Objectives
 - » The Brain
 - » The Instruments
 - » **The Mixer**
 - » Interprocess Communication
- Project Timeline
- Current Status
- The Future
- Questions and Comments

- Mixes each PCM signal from the instruments into one output PCM signal
- Current implementation uses the JACK audio library



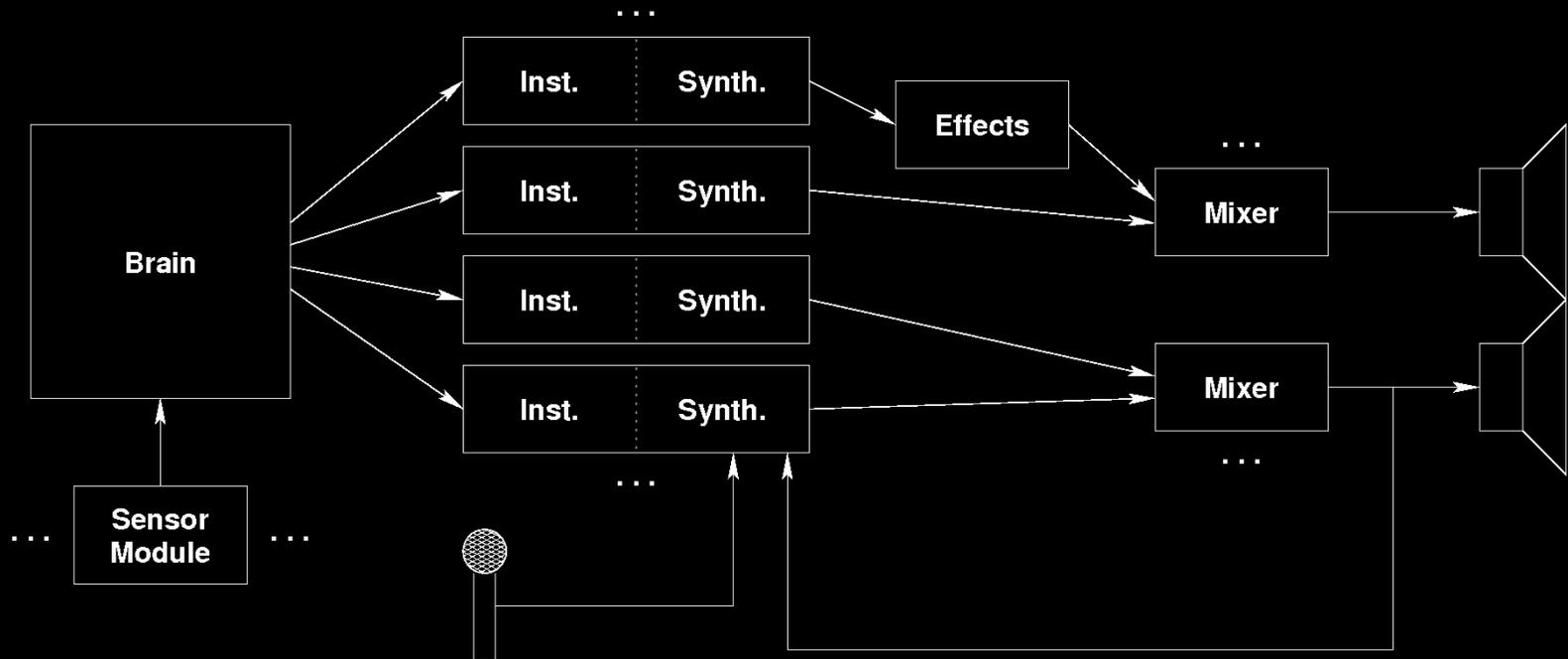
Interprocess Communication

- Project Overview
- » Project Overview
- » Technical Abstractions
- » Project Objectives
- » The Brain
- » The Instruments
- » The Mixer
- » Interprocess Communication

- Project Timeline
- Current Status
- The Future
- Questions and Comments

Regular UNIX named pipes were used in the IPC library.

Any other communication medium (TCP/IP) could be used.



Project Timeline

Project Overview

Project Timeline

» Project Timeline

» Difficulties and Problems

Current Status

The Future

Questions and Comments

- 2008.01.31 (done 2008.01.31) : All interfaces are defined and documented
- 2008.02.17 (done 2008.04.20) : Sample-based synthesizer and simple instrument finished
- 2008.03.01 (done 2008.04.12) : An autonomous brain is complete
- 2008.03.16 (done 2008.04.21) : The s.c.r.e.a.m. framework is complete
- 2008.04.30 : A polished s.c.r.e.a.m. implementation will be demonstrated

Difficulties and Problems

Project Overview

Project Timeline

» Project Timeline

» Difficulties and Problems

Current Status

The Future

Questions and Comments

- Main problem: shortage of time
- Clean implementation vs. efficiency
- System synchronization

Accomplishments

Project Overview

Project Timeline

Current Status

» Accomplishments

» The SimpleModularBrain

» SimpleModularBrain

Limitations

» Simple Instruments

» Synthesizer

» Audio

The Future

Questions and Comments

- Working brain implementation (SimpleModularBrain)
- Several simple instruments
- Simple synthesizer and audio output library
- Fully functional PMR library
- Fully functional IPC library
- Interfaces fully defined and documented (framework is **complete**)

The SimpleModularBrain

Project Overview

Project Timeline

Current Status

» Accomplishments

» The SimpleModularBrain

» SimpleModularBrain

Limitations

» Simple Instruments

» Synthesizer

» Audio

The Future

Questions and Comments

- Generates very simple chord sequences, then beat sequences, then melodies
- Uses PMR files to define probabilities:

```
CspsFm3 | =0.4 # F7
```

```
CspsBbm7 | =0.3 # Bbm7
```

```
CspsCm3 | =0.2 # C7
```

```
CspsBm12 | =0.08 # Bdim7
```

SimpleModularBrain Limitations

Project Overview

Project Timeline

Current Status

» Accomplishments

» The SimpleModularBrain

» SimpleModularBrain

Limitations

» Simple Instruments

» Synthesizer

» Audio

The Future

Questions and Comments

- Precision only to 16th notes (250000 microbeats)
- Cannot read every possible PMR condition
- Each component of the form is separate
- Tempos, time signature, and root key of form are hardcoded

Simple Instruments

Project Overview

Project Timeline

Current Status

- » Accomplishments
- » The SimpleModularBrain
- » SimpleModularBrain
- Limitations
- » Simple Instruments
- » Synthesizer
- » Audio

The Future

Questions and Comments

- Simple Horribly Ignorant Tambourine
- Techno High Hat
- Thump Drum
- Simple Walking Bass
- Simple Trumpet

Synthesizer

Project Overview

Project Timeline

Current Status

» Accomplishments

» The SimpleModularBrain

» SimpleModularBrain

Limitations

» Simple Instruments

» Synthesizer

» Audio

The Future

Questions and Comments

- Simple polyphonic synthesizer
- One synthesizer per instrument
- Sample parameters on command line

Audio

Project Overview

Project Timeline

Current Status

» Accomplishments

» The SimpleModularBrain

» SimpleModularBrain

Limitations

» Simple Instruments

» Synthesizer

» Audio

The Future

Questions and Comments

- Separate library
- Connects synthesizer to JACK
- Source of timing
- `scream_wave` `libsndfile` front end

Demonstration

Project Overview

Project Timeline

Current Status

The Future

» Demonstration

» Future Plans

Questions and Comments

- April 30th: live demonstration of simple implementation
- Components that will be used:
 - SimpleModularBrain
 - Trumpet and Tenor Sax
 - Acoustic Guitar
 - Bass Guitar
 - Percussion (tambourine, drum kit, possibly cowbell)
 - Anything else the Orange Lunchbox Brigade might write
 - Synthesizer for all instruments
 - Audio backend
- A clear demonstration of the robustness and flexibility of the framework

Future Plans

Project Overview

Project Timeline

Current Status

The Future

» Demonstration

» Future Plans

Questions and Comments

- The code will be released under the GPLv3
- Development will continue in an open-source development style
- You can help!

The project website is currently:

<http://www.igglybob.com/projects/scream/>

Questions and comments?

Project Overview

Project Timeline

Current Status

The Future

Questions and Comments

» Questions and comments?

These slides were created in \LaTeX .