SCORES

How to make a score

Scores

{ sound-event
  sound-event
  sound-event
  sound-event
  … }

{ time duration sound }

{ instrument attribute: value
           attribute: value
           attrbute: value
           … }

Score Example

```
{{0.0 1.0 {note pitch: 60 vel: 100}}
 {1.0 1.0 {note pitch: 62 vel: 110}}
 {2.0 1.0 {note pitch: 64 vel: 120}}}
```

 3

Score with `score-begin-end` Pseudo-Event

- Can a score be a sound event?
- If so, when does it start? How long is it?

```
{{0 0 {score-begin-end 0 5}}
 {0.0 1.0 {note pitch: 60 vel: 100}}
 {1.0 1.0 {note pitch: 62 vel: 110}}
 {2.0 1.0 {note pitch: 64 vel: 120}}}
```

 4

Instruments

- An "instrument" is a SAL (or XLISP) function
- How do we get from

   `{note pitch: 60 vel: 100}`

  to a function call?
- STEP 1: List representation of function calls
- STEP 2: Keyword parameters

5

List Representation of Function Calls (Lisp Syntax)

- A function call in Lisp is represented by:
  - Function *symbol* followed by …
  - … parameter expressions

           `(pluck ef4 3.0)`

- Expression can be
  - Number: evaluates to self
  - Symbol: evaluated as a variable
  - List: nested function call

6

Keyword Parameters

```
function note(pitch: 60, vel: 100)
  begin
     return pluck(pitch) * vel * 0.01
  end
```

- Now, we can call it:

```
play note(pitch: 72)
play note(vel: 50, pitch: g3) ~ 2
```

Putting It Together: Lisp Syntax + Keyword Parameters

- Example of an expression from a score:

```
{note pitch: 48 vel: 95}
```

- Equivalent to this SAL function call:

```
note(pitch: 48, vel: 95)
```

- Whole sound event might look like:

```
{3.0 1.5 {note pitch: 48 vel: 95}}
```

- Equivalent to this SAL expression:

```
(note(pitch: 48, vel: 95) ~ 1.5) @ 3.0
```