

CHORDS

A short-hand notation for scores

Why Keyword Parameters? Why Lisp?

- Scores are *data*
 - Score manipulation: transpose, stretch, select, ...
 - Score generation: algorithmic composition, ...
- Scores are *programs*
 - Well-defined semantics
 - Extensible through attributes and function definition

Special Case: Chords!

- Example of an event from a score:


```
{3.0 0.7 {note pitch: {48 55 64} vel: 95}}
```
- Lists of pitches are “expanded” to individual events, i.e. chords
- Equivalent to these events:


```
{3.0 0.7 {note pitch: 48 vel: 95}}
{3.0 0.7 {note pitch: 55 vel: 95}}
{3.0 0.7 {note pitch: 64 vel: 95}}
```
- Note that timing and all non-pitch parameters are duplicated for each note in the chord. (This only works for pitch:)

Scores Rendering

- `play timed-seq(my-score)`
 - Use `timed-seq` to turn a score into a `SOUND`
 - Further processing, e.g. reverb, is possible
- `exec score-play(my-score)`
 - Simple function to play a score
 - Does not return a `SOUND` value