

JOT – A Musical Notation Language

Brad Helstrom

June 4, 2010

The compiler I plan to implement will represent the beginnings of a musical notation language, in which notes, chords and melodies can be created by manipulating notes. Given the time constraints of this project, I hope to get a basic notational syntax and semantics defined with which further development can occur (including frequency conversion, scale libraries, etc.). Due to time constraints the notation will also only contain quarter notes.

Basic units

The fundamental unit in this compiler is the note (with an optional accidental). It will be defined by letter, in keeping with musical notation.

accidental -> # | b | ε
octave -> 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ε
note -> A | B | C | D | E | F | G | *noteaccidental* | *octavenote*
 {defaults to octave 4 and no accidental}

i.e. 4Ab, F#, 7C, etc.

The note can be then combined to define chords. The initial release of this compiler will explicitly have to spell out the formation of the chord from lowest note to highest note.

chord -> *note* | *note* + *note*

i.e.

/* C major chord triad and D minor third*/

chord CM = C + E + G;

chord Dm = D + F;

Scoring

Scoring the music consists of taking the note and chord progression and printing them to a staff. The notation will follow standard musical notation. Due to the time constraints of this project, rhythm elements will be reduced to quarter notes only.

