Tuning Interchange Documents

The diffusion of software synthesizers and software samplers over the past few years generated a growing interest for standardized microtuning control systems, the most interesting of which are based on specially formatted human-readable text files containing the device microtuning data (later on I will refer to these files as Tuning Interchange DOCuments, or TIDOCs). Besides the formats based on MTS sysex messages (MIDI Tuning Standard), that can also be embedded in a standard MIDI file (.mid) but that cannot be considered "human-readable" documents, the most relevant attempts to use TIDOCs in order to add microtonal support to commercial software synthesizers have involved the Scala format (.scl, used for example by the rgc:audio z3ta+ for Windows) and the Native Instruments Pro-52 format (.p5m, also adopted for their Pro-53 model). This article briefly describes the syntax of the Microtuner TIDOC format (.mtx), the native format of Max Magic Microtuner, a microtonal software originally developed as a companion application for Max/MSP.

The Microtuner TIDOC format (.mtx)

Three main concepts were followed in obtaining the Microtuner format, and they are:

I) human readability;
II) easy programming of the synth-side conversion routines;
III) no need for a different keyboard mapping format: scales and keyboard mappings are actually the same format.

There is a trade-off between these three factors that made all decisions converge toward a TIDOC format based on pure frequencies in Hertz rather than measurements in cents (this was also possible thanks to the fact that some hardware constraints pertaining to electronic design limitations - and even design traditions - are not an issue with software synthesizers). The basic syntax rules are the following (in these notes a *"receiving device"* is any software capable of reading Microtuner .mtx TIDOCs):

1. in a Microtuner TIDOC all lines end with a carriage return (ASCII 13) or with a carriage return followed by a line feed character (ASCII 13 + ASCII 10, typical of Windows applications): the receiving device must correctly interpret both the cases; empty lines are allowed anywhere in the file and they must be disregarded by the receiving device;

2. comment lines start with a double slash ("//"); all comment lines should be placed in the header part of the TIDOC (this is not mandatory);

3. the first numeric line (integer, 0-127) starting with "@" is interpreted as the lowest MIDI note number to which all the frequencies listed below will be referred (60 = middle C);

4. the scale expansion mode depends on a special line statement that follows the MIDI note number line; this statement can be ":absolute" or ":intervals";

4.1 in order to define an n-tone octave-wise scale, where frequencies are doubled octave after octave, the n tones (frequencies) of one octave (ANY octave of the scale) must be listed below a ":absolute" statement; as a particular case, a whole keyboard mapping can be defined with the ":absolute" mode (the MIDI note number line in this case would be "@0");

4.2 in order to define a non-octave scale based on intervals between frequencies then, since the receiving device has to determine n different frequency ratios between the adjacent tones of a scale, n+1 frequencies must be listed below a ":intervals" statement. The receiving device will then compute and use the ratios repeatedly to extend the scale over the full MIDI note range; as a particular case, the ":intervals" mode is useful for octave-wise scales based on an equal division of the octave, where the frequency ratio between two adjacent tones is constant (like in the 12-tone Equal Tempered scale);

5. zero frequency values are allowed in the frequency list in order for the corresponding MIDI notes to be left unmapped, but their presence in the list is inconsistent in the case of the ":intervals" mode;

6. the scale expansion algorithm of the receiving device will have to properly manage out-of-range values; the allowed frequency range in Max Magic Microtuner is 0-22050 Hz.

Examples from the Max Magic Microtuner documentation

Example 1: 12-tone Equal Tempered scale (expansion mode = absolute)

```
// 12 tone Equal Tempered scale, absolute mode

@60

:absolute

261.62558
277.182617
293.664764
311.126984
329.627563
349.228241
369.994415
391.995422
415.304688
440
466.163757
493.883301
```

Example 2: 12-tone Equal Tempered scale (expansion mode = intervals)

```
// 12 tone Equal Tempered scale, intervals mode
//
// A4 (MIDI note #69) = ref. frequency = f1
// f2 = f1 * pow(2, 1/12)

@69

:intervals

440.
466.16376151809
```

Additional information

Max Magic Microtuner: http://groups.yahoo.com/group/16tone
Max/MSP: http://www.cycling74.com
Scala home page: http://www.xs4all.nl/~huygensf/scala
Scala file format details: http://www.xs4all.nl/~huygensf/scala/scl_format.html
Native Instruments: http://www.nativeinstruments.com
rgc:audio: http://www.rgcaudio.com