**Instruments**

I want to support most types of Harmonicas ; Diatonic, Chromatic and Tremolo with a few odd exceptions, e.g. stacked tremolo harmonicas.

As such , a harmonica is defined as something having a fixed number of holes, which produce a tone on blow or on draw, both optionally, and which can also be bent, and may optionally have a button which raises each tone a single semitone.

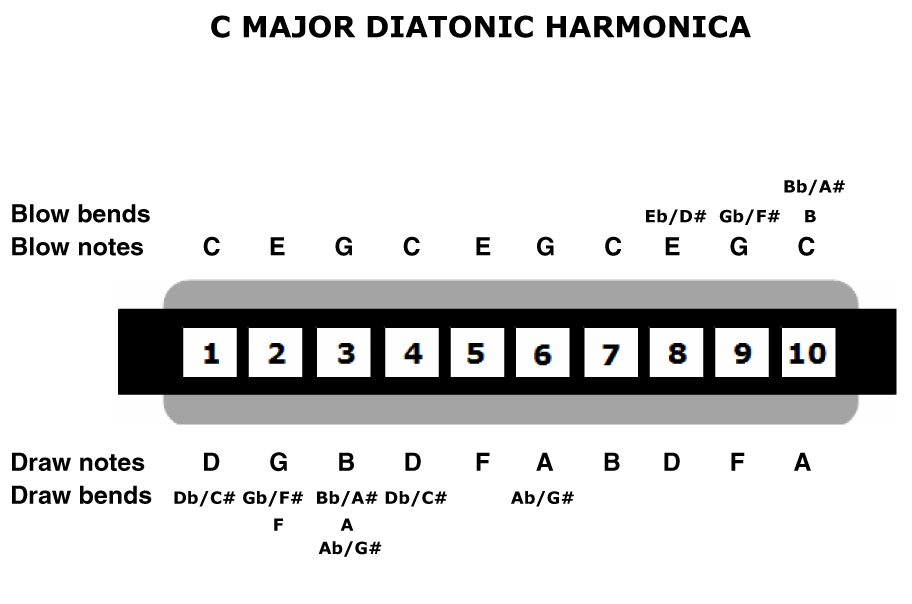
Initially these are designed using a Python class, named after the Harmonica. (e.g. class DiatonicHarmonica). It has methods getHoleCount() getBlowNote() getDrawNote(),getHighestBend() hasChromaticButton(). Note values are None, if no note is produced (e.g. Tremolo Harmonica) or an integer which is an offset from C4 (e.g. C4 == 1). hasChromaticBoolean() returns a Boolean. getHighestBend() returns the furthest bend possible in +/- semitones (or 0 for no bend). Finally method getName() returns the name of the Harmonica as a string, getShortName() returns a simple key.

Each constructor has an optional parameter which shifts the tuning, so the default DiatonicHarmonica would be a ‘C’, and you could produce a ‘D’ tuned Diatonic by setting this parameter to 2.

The initial DiatonicHarmonica class should be viewed as the current documentation status.

Each of these are subsequently analysed to extract the data, the purpose of this is to say whether this note can be played and if so how.

(to be completed)



Instrument data is stored in text format, and can be analysed by Python or Javascript.

The main part is separated by semicolons as follows

*<short key name> ; <long name> ; <number of holes> ;*

*<chromatic Button> (Y or N) ; <playable notes>*

The <playable notes> section is separated by commas and consists of pairs of <noteid> = <production>

So to produce F# (draw bend hole 2 once) it would be

7=-2b

* **7** is the note number
* - is for drawing (nothing for blowing)
* **2** is the hole number
* **b** is for bend one step

This gives you a single long string. All of these are reformatted into a JSON/Python format

{

“key” : “text”,

}