

A brief guide on reamber_base

Eve-ning

23/2/2019

Part I

Common Concepts

In this document, I will be referencing a few terms/abbreviations to save time. I will assume that you understand most of the following terms.

Table 1: Input Abbreviations

Abbr	Expanded	Eg.
TP	Timing Point	150,200,4,2,0,20,1,0
TP_{SV}	Timing Point (Slider Velocity)	150,-200,4,2,0,20,0,0
TP_{BPM}	Timing Point (Beats Per Minute)	150,200,4,2,0,20,1,0
HO	Hit Object	448,192,150,1,0,0:0:0:0:
HO_{NN}	Hit Object (Normal Note)	448,192,150,1,0,0:0:0:0:
HO_{LN}	Hit Object (Long Note)	448,192,150,128,0,200:0:0:0:0:
EHO	Editor Hit Object	00:00:150 (150 3,200 3) -

Table 2: Timing Point Deconstructed

Example		
TP_{SV}	100,-200,4,2,1,20,0,1	
TP_{BPM}	100,200,4,2,1,20,1,1	
Label	Code	Remarks
100	Offset	Unit in ms
-200	Code (TP_{SV})	Actual Value: $-100/Code$
200	Code (TP_{BPM})	Actual Value: $60000/Code$
4	Metronome	
2	Sample Set	<i>Reference: Sample Sets</i>
1	Sample Set Index	
20	Volume	Set local volume for note
0	Type	0 1 := $SV BPM$
1	Kiai	0 1 := $Off On$

Table 3: Hit Object Deconstructed

Example		
HO_{NN}	448,192,150,1,0,1:2:3:40:	
HO_{LN}	448,192,150,128,0,200:1:2:3:40:	
Label	Code	Remarks
448	Column Code/X-Axis	Column: $round((code * keys - 256)/512)$
192	Y-Axis	No Effect
150	Offset	Unit in ms
1	Type	$1 5 128 := NN NN LN$
0	Hitsound	<i>Reference: Sample Sets</i>
200	Long Note End	<i>(OPTIONAL)Only for HO_{LN}</i>
1	Sample Set	<i>Reference: Sample Sets</i>
2	Addition Set	<i>Reference: Sample Sets</i>
3	Custom Set Index	Set hitsound file index to reference
40	Volume	Set local volume for note

Table 4: Sample Sets

Value	Sample Set
0	AUTO
1	NORMAL
2	SOFT
3	DRUM

Part II

Coding Guidelines

1 Aim

The main purpose of this guideline is to make sure that any contributor will follow the same focus

2 Guidelines

1. Keep everything stupid simple
 - (a) File names should be as short as possible
 - (b) Function names should be too
 - (c) Function inputs should always be constant, functions should always return
 - (d) More heavy weight functions, less fragmented ones. i.e. overload.
2. Reuse functions as much as possible
3. Function names should make as much sense as possible, if not possible, attach a visual
4. Use folders to group similar functions if needed, however, consider collapsing them into a large function first.

Part III

Objects

3 Singular

The simplest block is the Hit Object and Timing Point

3.1 Hit Object

For the Hit Object, it has a few primitives to take note of unsigned int m_column unsigned int m_y_axis unsigned int m_note_type sample_set m_hitsound.set double m_ln_end sample_set m_sample_set sample_set m_addition_set sample_set m_custom_set unsigned int m_volume std::string m_hitsound_file unsigned int m_keys