

# Reference of HMM-based Speech Synthesis Engine

## “hts\_engine API” version 1.07

HTS Working Group  
December 25, 2012

## 1 Engine structures

### 1.1 Audio

**HTS\_Audio** Audio output wrapper.

size_t	<b>sampling_frequency</b>	-	sampling frequency
size_t	<b>max_buff_size</b>	-	buffer size of audio output device
short	<b>*buff</b>	-	current buffer
size_t	<b>buff_size</b>	-	current buffer size
void	<b>*audio_interface</b>	-	audio interface specified in compile step

### 1.2 Model

**HTS\_Window** Window coefficients to calculate dynamic features.

size_t	<b>size</b>	-	# of windows (static + deltas)
int	<b>*l_width</b>	-	left width of windows
int	<b>*r_width</b>	-	right width of windows
double	<b>**coefficient</b>	-	window coefficients
size_t	<b>max_width</b>	-	maximum width of windows

**HTS\_Pattern** List of patterns in a question and a tree.

char	<b>*string</b>	-	pattern string
HTS_Pattern	<b>*next</b>	-	pointer to the next pattern

**HTS\_Question** List of questions in a tree.

char	<b>*string</b>	-	name of this question
HTS_Pattern	<b>*head</b>	-	pointer to the head of pattern list
HTS_Question	<b>*next</b>	-	pointer to the next question

**HTS\_Node** List of tree nodes in a tree.

int	<b>index</b>	-	index of this node
size_t	<b>pdf</b>	-	index of PDF for this node (leaf node only)
HTS_Node	<b>*yes</b>	-	pointer to its child node (yes)
HTS_Node	<b>*no</b>	-	pointer to its child node (no)
HTS_Node	<b>*next</b>	-	pointer to the next node
HTS_Question	<b>*quest</b>	-	question applied at this node

**HTS\_Tree** List of decision trees in a model.

HTS_Pattern	<b>*head</b>	-	pointer to the head of pattern list for this tree
HTS_Tree	<b>*next</b>	-	pointer to the next tree
HTS_Node	<b>*root</b>	-	root node of this tree
size_t	<b>state</b>	-	state index of this tree

**HTS\_Model** Set of PDFs, decision trees and questions.

size_t	<b>vector_length</b>	-	vector length (static features only)
size_t	<b>num_windows</b>	-	# of windows for delta
HTS_Boolean	<b>is_msd</b>	-	flag for MSD
size_t	<b>ntree</b>	-	# of trees
size_t	<b>*npdf</b>	-	# of PDFs at each tree
float	<b>***pdf</b>	-	PDFs
HTS_Tree	<b>*tree</b>	-	pointer to the list of trees
HTS_Question	<b>*question</b>	-	pointer to the list of questions

**HTS\_ModelSet** Set of duration models, HMMs and GV models.

char	<b>*hts_voice_version</b>	-	version of HTS voice format
size_t	<b>sampling_frequency</b>	-	sampling frequency
size_t	<b>frame_period</b>	-	frame period
size_t	<b>num_voices</b>	-	# of HTS voices
size_t	<b>num_states</b>	-	# of HMM states
size_t	<b>num_streams</b>	-	# of streams
char	<b>*stream_type</b>	-	stream type
char	<b>*fullcontext_format</b>	-	fullcontext label format
char	<b>*fullcontext_version</b>	-	version of fullcontext label
HTS_Question	<b>*gv_off_context</b>	-	GV switch
char	<b>**option</b>	-	options for each stream
HTS_Model	<b>*duration</b>	-	duration PDFs and trees
HTS_Window	<b>*window</b>	-	window coefficients for delta
HTS_Model	<b>**stream</b>	-	parameter PDFs and trees
HTS_Model	<b>**gv</b>	-	GV PDFs and trees

### 1.3 Label

**HTS\_LabelString** Individual label string with time information.

HTS_LabelString	<b>*next</b>	-	pointer to the next label string
char	<b>*name</b>	-	label string
double	<b>start</b>	-	start frame specified in the given label
double	<b>end</b>	-	end frame specified in the given label

**HTS\_Label** List of label strings.

HTS_LabelString	<b>*head</b>	-	pointer to the head of label string
size_t	<b>size</b>	-	# of label strings

### 1.4 State stream

**HTS\_SStream** Individual state stream.

size_t	<b>vector_length</b>	-	vector length (static features only)
double	<b>**mean</b>	-	mean vector sequence
double	<b>**vari</b>	-	variance vector sequence
double	<b>*msd</b>	-	MSD parameter sequence
size_t	<b>win_size</b>	-	# of windows (static + deltas)
int	<b>*win_l_width</b>	-	left width of windows
int	<b>*win_r_width</b>	-	right width of windows
double	<b>**win_coefficient</b>	-	window coefficients
size_t	<b>win_max_width</b>	-	maximum width of windows
double	<b>*gv_mean</b>	-	mean vector of GV
double	<b>*gv_vari</b>	-	variance vector of GV
HTS_Boolean	<b>*gv_switch</b>	-	GV flag sequence

**HTS\_SStreamSet** Set of state stream.

HTS_SStream	<b>*sstream</b>	-	state streams
size_t	<b>nstream</b>	-	# of streams
size_t	<b>nstate</b>	-	# of states
size_t	<b>*duration</b>	-	duration sequence
size_t	<b>total_state</b>	-	total state
size_t	<b>total_frame</b>	-	total frame

### 1.5 PDF stream

**HTS\_SMatrices** Matrices/Vectors used in the speech parameter generation algorithm.

double	<b>**mean</b>	-	mean vector sequence
double	<b>**ivar</b>	-	inverse diagonal variance sequence
double	<b>*g</b>	-	vector used in the forward substitution
double	<b>**wuw</b>	-	$W' U^{-1} W$
double	<b>*wum</b>	-	$W' U^{-1} m$

**HTS\_PStream** Individual PDF stream.

size_t	<b>vector_length</b>	-	vector length (static features only)
size_t	<b>length</b>	-	stream length
size_t	<b>width</b>	-	width of dynamic window
double	<b>**par</b>	-	output parameter vector
HTS_SMatrices	<b>sm</b>	-	matrices for parameter generation
size_t	<b>win_size</b>	-	# of windows (static + deltas)
int	<b>*win_l_width</b>	-	left width of windows
int	<b>*win_r_width</b>	-	right width of windows
double	<b>**win_coefficient</b>	-	window coefficients
HTS_Boolean	<b>*msd_flag</b>	-	Boolean sequence for MSD
double	<b>*gv_mean</b>	-	mean vector of GV
double	<b>*gv_vari</b>	-	variance vector of GV
HTS_Boolean	<b>*gv_switch</b>	-	GV flag sequence
size_t	<b>gv_length</b>	-	frame length for GV calculation

**HTS\_PStreamSet** Set of PDF streams.

HTS_PStream	<b>*pstream</b>	-	PDF streams
size_t	<b>nstream</b>	-	# of PDF streams
size_t	<b>total_frame</b>	-	total frame

## 1.6 Generated parameter stream

**HTS\_GStream** Generated parameter stream.

size_t	<b>vector_length</b>	-	vector length (static features only)
double	<b>**par</b>	-	generated parameter

**HTS\_GStreamSet** Set of generated parameter stream.

size_t	<b>total_nsample</b>	-	total sample
size_t	<b>total_frame</b>	-	total frame
size_t	<b>nstream</b>	-	# of streams
HTS_GStream	<b>*gstream</b>	-	generated parameter streams
double	<b>*gspeech</b>	-	generated speech

## 1.7 Engine

**HTS\_Condition** Synthesis condition.

size_t	<b>sampling_frequency</b>	- sampling frequency
size_t	<b>fperiod</b>	- frame period
size_t	<b>audio_buff_size</b>	- audio buffer size (for audio device)
HTS_Boolean	<b>stop</b>	- stop flag
double	<b>volume</b>	- volume
double	<b>*msd_threshold</b>	- MSD thresholds
double	<b>*gv_weight</b>	- GV weights
HTS_Boolean	<b>phoneme_alignment_flag</b>	- flag for using phoneme alignment in label
double	<b>speed</b>	- speech speed
size_t	<b>stage</b>	- if $stage = 0$ then $gamma = 0$ else $gamma = -1/stage$
HTS_Boolean	<b>use_log_gain</b>	- log gain flag (for LSP)
double	<b>alpha</b>	- all-pass constant
double	<b>beta</b>	- postfiltering coefficient
double	<b>additional_half_tone</b>	- additional half tone
double	<b>*duration_iw</b>	- weights for duration interpolation
double	<b>**parameter_iw</b>	- weights for parameter interpolation
double	<b>**gv_iw</b>	- weights for GV interpolation

**HTS\_Engine** Engine itself.

HTS_Condition	<b>condition</b>	- synthesis condition
HTS_Audio	<b>audio</b>	- audio output
HTS_ModelSet	<b>ms</b>	- set of duration models, HMMs and GV models
HTS_Label	<b>label</b>	- label
HTS_SStreamSet	<b>sss</b>	- set of state streams
HTS_PStreamSet	<b>pss</b>	- set of PDF streams
HTS_GStreamSet	<b>gss</b>	- set of generated parameter streams

## 2 Engine functions

### 2.1 Initialize engine

#### 2.1.1 HTS\_Engine\_initialize

Type            void  
Use            Initialize engine.  
Arguments    HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!   To start engine, first you must call this function.

### 2.2 Load models

#### 2.2.1 HTS\_Engine\_load

Type            HTS\_Boolean  
Use            Load duration PDFs and trees from files using given file names.  
Arguments    HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
                 char    \*\*voices   -   HTS voice file names  
                 size\_t   num\_voices -   # of HTS voices  
Attention!!   You must initialize engine using HTS\_Engine\_initialize before calling this function.

### 2.3 Synthesize speech and set/get synthesis parameters

#### 2.3.1 HTS\_Engine\_set\_sampling\_frequency

Type            void  
Use            set sampling frequency.  
Arguments    HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
                 size\_t    i           -   sampling frequency (Hz),  $1 \leq i$

#### 2.3.2 HTS\_Engine\_get\_sampling\_frequency

Type            size\_t  
Use            get sampling frequency.  
Arguments    HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

#### 2.3.3 HTS\_Engine\_set\_fperiod

Type            void  
Use            set frame shift.  
Arguments    HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
                 size\_t    i           -   frame shift (point),  $1 \leq i$

### 2.3.4 HTS\_Engine\_get\_fperiod

Type           size\_t  
Use            get frame shift.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.5 HTS\_Engine\_set\_audio\_buff\_size

Type           void  
Use            set buffer size for direct audio output.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             size\_t    i            -   buffer size (sample)  
Attention!!   Default value is 0. If  $i = 0$ , direct audio play is turned off.

### 2.3.6 HTS\_Engine\_get\_audio\_buff\_size

Type           size\_t  
Use            get buffer size for direct audio output.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!   Default value is 0. If  $i = 0$ , direct audio play is turned off.

### 2.3.7 HTS\_Engine\_set\_stop\_flag

Type           void  
Use            set stop flag.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             HTS\_Boolean   b        -   flag  
Attention!!   Default value is FALSE.

### 2.3.8 HTS\_Engine\_get\_stop\_flag

Type           HTS\_Boolean  
Use            get stop flag.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!   Default value is FALSE.

### 2.3.9 HTS\_Engine\_set\_volume

Type           void  
Use            set volume in db.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             double    f            -   volume in db  
Attention!!   Default value is 0.0.

### 2.3.10 HTS\_Engine\_get\_volume

Type           double  
Use            get volume in db.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.11 HTS\_Engine\_set\_msd\_threshold

Type           void  
Use            set MSD threshold.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              size\_t    stream\_index -   index of streams  
              double    f           -   threshold

### 2.3.12 HTS\_Engine\_get\_msd\_threshold

Type           double  
Use            get MSD threshold.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              size\_t    stream\_index -   index of streams

### 2.3.13 HTS\_Engine\_set\_gv\_weight

Type           void  
Use            set GV weight.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              size\_t    stream\_index -   index of streams  
              double    f           -   GV weight  
Attention!!   Default value is 1.0.

### 2.3.14 HTS\_Engine\_get\_gv\_weight

Type           double  
Use            get GV weight.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              size\_t    stream\_index -   index of streams



### 2.3.15 HTS\_Engine.set\_speed

Type           void  
Use            set speech speed.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              double    f           -   speed  
Attention!!   Default value is 1.0.

### 2.3.16 HTS\_Engine.set\_phoneme\_alignment\_flag

Type           void  
Use            set flag to use phoneme alignment in label.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              HTS\_Boolean   b       -   flag  
Attention!!   Default value is FALSE.

### 2.3.17 HTS\_Engine.set\_alpha

Type           void  
Use            set frequency warping parameter alpha.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              double    f           -   alpha,  $0.0 \leq f \leq 1.0$

### 2.3.18 HTS\_Engine.get\_alpha

Type           double  
Use            get frequency warping parameter alpha.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.19 HTS\_Engine.set\_beta

Type           void  
Use            set postfiltering coefficient parameter beta.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
              double    f           -   beta,  $0.0 \leq f \leq 1.0$   
Attention!!   Default value is 0.0.

### 2.3.20 HTS\_Engine.get\_beta

Type           double  
Use            get postfiltering coefficient parameter beta.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!   Default value is 0.0.

### 2.3.21 HTS\_Engine\_add\_half\_tone

Type	void		
Use	add half tone.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	double	f	- half tone

### 2.3.22 HTS\_Engine\_set\_duration\_interpolation\_weight

Type	void		
Use	set weight for duration interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of duration models
	double	f	- interpolation weight

### 2.3.23 HTS\_Engine\_get\_duration\_interpolation\_weight

Type	double		
Use	get weight for duration interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of duration models

### 2.3.24 HTS\_Engine\_set\_parameter\_interpolation\_weight

Type	void		
Use	set weight for parameter interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of parameter models
	size_t	stream_index	- index of streams
	double	f	- interpolation weight

### 2.3.25 HTS\_Engine\_get\_parameter\_interpolation\_weight

Type	double		
Use	get weight for parameter interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of parameter models
	size_t	stream_index	- index of streams

### 2.3.26 HTS\_Engine\_set\_gv\_interpolation\_weight

Type	void		
Use	set weight for GV interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of GV models
	size_t	stream_index	- index of streams
	double	f	- interpolation weight

### 2.3.27 HTS\_Engine\_get\_gv\_interpolation\_weight

Type	double		
Use	get weight for GV interpolation.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	voice_index	- index of GV models
	size_t	stream_index	- index of streams

### 2.3.28 HTS\_Engine\_get\_total\_state

Type	size_t		
Use	get total # of state.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure

### 2.3.29 HTS\_Engine\_set\_state\_mean

Type	void		
Use	set mean value of state.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	stream_index	- index of streams
	size_t	state_index	- index of states
	size_t	vector_index	- index of vector
	double	f	- mean value

### 2.3.30 HTS\_Engine\_get\_state\_mean

Type	double		
Use	get mean value of state.		
Arguments	HTS_Engine	*engine	- pointer to HTS_Engine structure
	size_t	stream_index	- index of streams
	size_t	state_index	- index of states
	size_t	vector_index	- index of vector

### 2.3.31 HTS\_Engine\_get\_state\_duration

Type           size\_t  
Use            get state duration.  
Arguments   HTS\_Engine   \*engine    -   pointer to HTS\_Engine structure  
             size\_t    state\_index -   index of states

### 2.3.32 HTS\_Engine\_get\_nvoices

Type           size\_t  
Use            get # of HTS voices.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.33 HTS\_Engine\_get\_nstream

Type           size\_t  
Use            get # of stream.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.34 HTS\_Engine\_get\_nstate

Type           size\_t  
Use            get # of state.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.35 HTS\_Engine\_get\_nsamples

Type           size\_t  
Use            get # of samples.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure

### 2.3.36 HTS\_Engine\_get\_speech

Type           size\_t  
Use            get generated speech.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             size\_t    index    -   index of samples

### 2.3.37 HTS\_Engine\_synthesize\_from\_fn

Type        HTS\_Boolean  
Use         synthesize speech from file name.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             char        \*fn        -   label file name

### 2.3.38 HTS\_Engine\_synthesize\_from\_strings

Type        HTS\_Boolean  
Use         synthesize speech from string list.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             char        \*\*lines   -   label string list  
             size\_t     num\_lines -   # of lines

### 2.3.39 HTS\_Engine\_save\_information

Type        void  
Use         output trace information.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             FILE        \*fp        -   output file pointer  
  
Attention!!

### 2.3.40 HTS\_Engine\_save\_label

Type        void  
Use         output label with time.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             FILE        \*fp        -   output file pointer  
  
Attention!!

### 2.3.41 HTS\_Engine\_save\_generated\_parameter

Type        void  
Use         output generated parameter.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
             FILE        \*fp        -   output file pointer  
  
Attention!!

### 2.3.42 HTS\_Engine\_save\_generated\_speech

Type           void  
Use            output generated speech.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
                          FILE       \*fp       -   output file pointer  
Attention!!

### 2.3.43 HTS\_Engine\_save\_riff

Type           void  
Use            output riff format file.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
                          FILE       \*fp       -   output file pointer  
Attention!!

### 2.3.44 HTS\_Engine\_refresh

Type           void  
Use            free label, state streams, PDF streams and generated parameter streams per one time synthesis  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!

## 2.4 Free engine

### 2.4.1 HTS\_Engine\_clear

Type           void  
Use            free engine.  
Arguments   HTS\_Engine   \*engine   -   pointer to HTS\_Engine structure  
Attention!!