

MMDAgent クリエイター向けリファレンス

Ver. 1.03

2016年9月27日
名古屋工業大学

目次

1. 本文書の位置付け	7
2. MMDAgent とは	7
3. MMDAgent のディレクトリ構成	8
システムディレクトリ <SysDir>	8
コンテンツディレクトリ <ContentsDir>	8
4. MMDAgent の操作法	10
マウス操作	10
キーボード操作	10
ドラッグ&ドロップ	11
実行時の引数	11
5. fst ファイルについて	12
概要	12
構文	12
変数	13
正規表現	14
メッセージ一覧	15
MODEL_ADD	15
MODEL_CHANGE	17
MODEL_DELETE	18
MODEL_EVENT_ADD	19
MODEL_EVENT_CHANGE	20
MODEL_EVENT_DELETE	21
MOTION_ADD	22
MOTION_ACCELERATE	24
MOTION_CHANGE	26
MOTION_DELETE	28
MOTION_EVENT_ADD	29
MOTION_EVENT_ACCELERATE	30
MOTION_EVENT_CHANGE	31
MOTION_EVENT_DELETE	32
MOVE_START	33
MOVE_STOP	35
MOVE_EVENT_START	36

MOVE_EVENT_STOP.....	37
TURN_START	38
TURN_STOP	40
TURN_EVENT_START	41
TURN_EVENT_STOP.....	42
ROTATE_START	43
ROTATE_STOP	45
ROTATE_EVENT_START	46
ROTATE_EVENT_STOP.....	47
SOUND_START	48
SOUND_STOP.....	49
SOUND_EVENT_START	50
SOUND_EVENT_STOP	51
STAGE	52
LIGHTCOLOR.....	54
LIGHTDIRECTION	55
CAMERA.....	56
RECOG_EVENT_START	58
RECOG_EVENT_STOP	59
RECOG_MODIFY GAIN.....	60
RECOG_EVENT_MODIFY GAIN	61
RECOG_MODIFY USERDICT_SET	62
RECOG_EVENT_MODIFY USERDICT_SET.....	63
RECOG_MODIFY USERDICT_UNSET	64
RECOG_EVENT_MODIFY USERDICT_UNSET.....	65
SYNTH_START	66
SYNTH_STOP	68
SYNTH_EVENT_START	69
SYNTH_EVENT_STOP	70
LIPSYNC_START	71
LIPSYNC_STOP	72
LIPSYNC_EVENT_START	73
LIPSYNC_EVENT_STOP.....	74
VALUE_SET	75
VALUE_UNSET	77
VALUE_EVAL	78

VALUE_EVENT_SET	80
VALUE_EVENT_UNSET	81
VALUE_EVENT_EVAL	82
TIMER_START	84
TIMER_STOP	85
TIMER_EVENT_START	86
TIMER_EVENT_STOP	87
PLUGIN_ENABLE	88
PLUGIN_DISABLE	89
PLUGIN_EVENT_ENABLE	90
PLUGIN_EVENT_DISABLE	91
DRAGANDDROP	92
KEY	93
EXECUTE	94
KEY_POST	95
6. mdf ファイルについて	96
概要	96
構文	96
設定項目一覧	97
use_cartoon_rendering	97
use_mmd_like_cartoon	97
cartoon_edge_width	97
cartoon_edge_step	98
cartoon_edge_selected_color	98
camera_rotation	98
camera_transition	100
camera_distance	100
camera_fovy	100
stage_size	101
show_fps	101
fps_position	101
window_size	102
full_screen	102
log_size	102
log_position	103
log_scale	103

light_direction.....	103
light_intensity.....	104
light_color.....	104
campus_color.....	104
max_multi_sampling.....	106
motion_adjust_time.....	106
lypsync_priority.....	106
bullet_fps.....	107
gravity_factor.....	107
rotate_step.....	107
translate_step.....	108
distance_step.....	108
fovy_step.....	108
use_shadow_mapping.....	109
shadow_mapping_texture_size.....	109
shadow_mapping_self_density.....	109
shadow_mapping_floor_density.....	110
shadow_mapping_light_first.....	110
display_comment_time.....	111
max_num_model.....	111
7. ojt ファイルについて.....	112
概要.....	112
構文.....	112
8. dic ファイルについて.....	114
概要.....	114
構文.....	114
9. lip ファイルについて.....	116
概要.....	116
構文.....	116
10. lab ファイルについて.....	118
概要.....	118
構文.....	118
11. 音素対応表.....	120
概要.....	120
音声認識用の音素対応表.....	121
音声合成用の音素対応表.....	122

1. 本文書の位置付け

本仕様書は、MMDAgent を使用するクリエイターに向けた、各種設定ファイルや対話スクリプトで使用されるメッセージに関する解説をまとめた仕様書である。

本仕様書は、以下のバージョンの MMDAgent を対象に作成されている。

▼対象とするバージョン

項目	バージョン
MMDAgent.exe 本体	1.6.1
MMDAgent_Example	1.6

▼解説対象ファイル

fst ファイル、mdf ファイル、ojt ファイル、dic ファイル、lip ファイル、lab ファイル

2. MMDAgent とは

MMDAgent は音声認識や音声合成の実行に必要なモデルや定義・設定ファイルを読み込み、3D オブジェクトを OpenGL でレンダリングしながら、音声入力の録音からのリアルタイム音声認識や合成音声のリアルタイム出力を行い、FST 形式で書かれた対話スクリプトに従って 3D オブジェクトのモーション・移動等を行うことで音声インタラクションを実行するソフトウェアである。

▼MMDAgent 公式サイト

<http://www.mmdagent.jp/>

3. MMDAgent のディレクトリ構成

システムディレクトリ <SysDir>

MMDAgent.exe および AppData ディレクトリ、Plugins ディレクトリ、およびシステム共通設定ファイルが存在するディレクトリ。

▼ディレクトリ / ファイル構成

システムディレクトリの構成を以下に記す。

<SysDir>	
└ AppData/	----- システムデータディレクトリ
├ └ Julius/	----- 音声認識 (Julius) 用ディレクトリ
├ └ └ lang_m/	----- 言語モデルディレクトリ
├ └ └ phone_m/	----- 音響モデルディレクトリ
├ └ └ └ jconf.txt	--- jconf 設定ファイル
├ └ Open_JTalk	----- 音声合成 (Open JTalk) 用ディレクトリ、発音辞書
├ └ └ lip.txt	----- リップシンク用音素、モーフ対応定義
└ Plugins/	----- プラグイン用ディレクトリ
└ MMDAgent.exe	----- MMDAgent 本体
└ └ MMDAgent.mdf	----- システム共通設定

コンテンツディレクトリ <ContentsDir>

MMDAgent.exe の引数に mdf ファイルを指定せずに起動した場合は、システムディレクトリがコンテンツディレクトリになる。システム共通設定ファイル MMDAgent.mdf が読み込まれた後、そのディレクトリに存在する MMDAgent.fst ファイル、MMDAgent.dic ファイル、MMDAgent.ojt ファイルが参照される。

MMDAgent.exe の引数に mdf ファイルを指定した場合は、mdf ファイルの存在するディレクトリがコンテンツディレクトリになる。システムディレクトリのシステム共通設定ファイル MMDAgent.mdf が読み込まれた後、指定された mdf ファイルが読み込まれ、そのディレクトリに存在する fst ファイル、dic ファイル、ojt ファイル（ファイル名は拡張子以外 mdf ファイルと同じ）が参照される。（mdf ファイルを MMDAgent.exe に関連付けして mdf ファイルから起動した場合や、mdf ファイルを MMDAgent.exe にドラッグ&ドロップした場合も同様。）

MMDAgent.exe の引数に mmda ファイルを指定した場合は、mmda ファイルがテンポラリディレクトリに解凍され、内部の mdf ファイルが読み込まれる。この場合、解凍されたディレクトリは MMDAgent が

終了する時に削除される。

いずれの場合も起動後、カレントディレクトリはコンテンツディレクトリに設定される。

4. MMDAgent の操作法

マウス操作

MMDAgent のマウス操作を以下に記す。

操作	MMDAgent の動作
ドラッグ	視点を回転する。
Shift + ドラッグ	視点を移動する。
ホイール	ズームの倍率を変更する。

キーボード操作

MMDAgent のキーボード操作を以下に記す。

操作	MMDAgent の動作
カーソルキー	視点を回転する。
Shift + カーソルキー	視点を移動する。
+/-	ズームの倍率を変更する。
Esc	プログラムを終了する。
B	ボーン表示の ON/OFF を切り替える。
D	ログ表示の ON/OFF を切り替える。(デバッグ画面)
E	エッジを太くする。
Shift + E	エッジを細くする。
F	フルスクリーン表示の ON/OFF を切り替える。
Shift + F	fst ログ表示の ON/OFF を切り替える。
H	ホールド機能の ON/OFF を切り替える。(一時停止)
Shift + J	音量バー表示の ON/OFF を切り替える。
L	マウスカーソル追尾の ON/OFF を切り替える。
Shift + O	コンテンツディレクトリを開く。
P	物理演算の ON/OFF を切り替える。
S	FPS 表示の ON/OFF を切り替える。
W	ワイヤーフレーム表示の ON/OFF を切り替える。
Shift + W	剛体表示の ON/OFF を切り替える。
X	セルフシャドウの ON/OFF を切り替える。
PageUp/PageDown	ログ表示画面をスクロールする。

ドラッグ&ドロップ

MMDAgent に特定ファイルをドラッグ&ドロップした場合の動作を以下に記す。

なお、ファイルをドロップしたタイミングで **DRAGANDDROP** メッセージが出力される。

対象ファイル + 操作	ドロップ対象	MMDAgent の動作
.vmd	モデル	ドロップされたモーションをループ再生する。
.vmd + Shift	モデル	ドロップされたモーションを 1 度だけ再生する。
.vmd + Ctrl	-	ドロップされたモーションを全てのモデルでループ再生する。
.vmd + Ctrl + Shift	-	ドロップされたモーションを全てのモデルで 1 度だけ再生する。
.pmd	モデル	ドロップされたモデルに変更する。
.pmd + Ctrl	-	ドロップされたモデルを表示する。
(.png .jpg .bmp)	-	ドロップされた画像を背景に設定する。
(.png .jpg .bmp) + Ctrl	-	ドロップされた画像をフロアに設定する。
.xpmd	-	ドロップされたモデルをステージに設定する。

実行時の引数

MMDAgent.exe の実行時に指定可能な引数を以下に記す。

対応ファイル	MMDAgent の動作
.pmd	引数で指定したモデルファイルを MMDAgent 起動時に読み込む。複数のモデルファイルを引数に指定した場合、指定したモデルファイルが全て読み込まれる。
.mdf	引数で指定した mdf ファイルの存在するディレクトリ位置をコンテンツディレクトリとして MMDAgent を起動する。
.mmda	引数で指定した mmda ファイルをテンポラリディレクトリに解凍し、解凍ディレクトリ内の mdf ファイルが存在する位置をコンテンツディレクトリとして MMDAgent を起動する。

5. fst ファイルについて

概要

MMDAgent で実行される対話スクリプトが定義されているファイル。

通常はひとつの fst ファイルを実行するが、複数の fst ファイルを並列実行することも可能。その場合は、以下の命名規則に従ったサブ fst ファイルをメイン fst ファイルと同一のディレクトリに格納することで、複数の fst ファイルが並列実行される。

▼サブ fst ファイルの命名規則

[命名規則]

```
メイン fst ファイルの名前.fst.サブ fst ファイルの名前.fst
```

[記述例]

```
MMDAgent.fst.Sub.fst
```

構文

現在の状態番号、次の状態番号、実行条件となるメッセージ、実行されるメッセージを順番に記述する。

構文中の区切り文字には [space] または [tab] を使用する。

メッセージ内で空白を利用する場合はダブルクォーテーション "" で囲む。

また、[#] で開始された行はコメントとして扱われる。

```
現在の状態番号 次の状態番号 実行条件となるメッセージ 実行されるメッセージ
```

▼記述例

1 キー押下でモデルファイルを表示する場合の記述例を以下に記す。

```
# 1 キー押下時、モデルを表示する。
```

```
0 1 KEY|1 MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
```

▼説明

- 現在の状態番号（第 1 フィールド）
現在の状態番号を記述する。
- 次の状態番号（第 2 フィールド）
メッセージ実行後に遷移先となる番号を記述する。

3. 実行条件となるメッセージ（第 3 フィールド）

実行する条件となるメッセージを記述する。

4. 実行されるメッセージ（第 4 フィールド）

条件を満たした場合に実行されるメッセージを記述する。

▼サンプルスクリプト

実際の fst ファイルは上記 4 つのフィールドから構成される行を複数行記述した形で構成されている。以下にモデルファイルを表示した後、音声認識（こんにちは）またはキー入力（1 キー）を条件に、モデルが音声合成（こんにちは。）するシナリオのサンプルスクリプトを以下に記す。

```
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 10 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP|ON|OFF
10 11 RECOG_EVENT_STOP|こんにちは <eps>
10 11 RECOG_EVENT_STOP|こんにちわ <eps>
10 11 KEY|1 <eps>
11 12 <eps> SYNTH_START|mei|mei_voice_normal|こんにちは。
12 10 SYNTH_EVENT_STOP|mei <eps>
```

▼備考

1. 状態番号の初期値

MMDAgent 起動時の状態番号は必ず 0 となる。また、fst ファイルごとに状態番号が存在し、複数の fst ファイルで状態番号は共有されない。

2. 複数行に同じ状態番号を記述した場合

同一の状態番号が存在する場合、先に発生したメッセージが優先される。

3. 発生しているメッセージの確認方法

デバッグ画面（D キー押下）を表示することで確認することができる。

変数

fst ファイルではローカル変数を利用できる。

変数への値の代入は第 5 フィールドに記述する。

値は文字列のみの扱いとなるため、数字も文字として扱われる。

変数名は半角アルファベット、半角数字、及び半角アンダーバーが利用できる。

代入した変数の値は第 3 フィールド、第 4 フィールドで利用できる。

▼記述例

キー押下で変数を利用する記述例を以下に記す。

```
# 1 キー押下時、変数 place1 に値 Nagoya を代入する。
0 1 KEY|1 <eps> ${place1}=Nagoya
# 1 キー押下時、変数 place2 に変数 place1 の値を代入する。
1 2 KEY|1 <eps> ${place2}=${place1}
# 1 キー押下時、変数 place3 と place4 に変数 place2 の値を代入する。
2 3 KEY|1 <eps> ${place3}=${place2},${place4}=${place2}
# 1 キー押下時、変数 place4 の名前のモデルファイルを表示する。
3 4 KEY|1 MODEL_ADD|map|${place4}.pmd
```

正規表現

fst ファイルでは第 3 フィールドに正規表現を利用できる。

正規表現を利用する場合、第 3 フィールド全体を半角アットマーク @ で囲い、その中にマッチングパターンを記述する。

正規表現内で半角括弧 () を指定することで、マッチした文字列断片を取得でき、文字列はローカル変数 \${1}, \${2}, … に自動的に格納される。

▼記述例

正規表現を利用する記述例を以下に記す。

```
# station の直前に出てきた単語列を変数 station に代入する。
0 1 @RECOG_EVENT_STOP¥|.*?([^,]*)|[Ss]tation.*@ <eps> ${station}=${1}
# 音声認識結果全体を変数 recog_string に代入する。
0 2 @RECOG_EVENT_STOP¥|(.*@ <eps> ${recog_string}=${1}
# モーション追加を検知して変数 model にモデル名、変数 motion にモーション名を代入する。
0 3 @MOTION_EVENT_ADD¥|(.*¥|(.*@ <eps> ${model}=${1},${motion}=${2}
```

メッセージ一覧

MODEL_ADD

▼説明

モデルを追加して 3D 空間に表示する。

モデルファイルの追加が完了したタイミングで MODEL_EVENT_ADD が出力される。

▼構文

```
MODEL_ADD|(model alias)|(model file name)|(x position),(y position),(z
position)|(x rotation),(y rotation),(z rotation)|(ON or OFF for
cartoon)|(parent model alias)|(parent bone name)
```

[引数]

•(model alias)

追加するモデルを参照するための別名。

•(model file name)

表示するモデルファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

•(x position),(y position),(z position) ※ 省略可能。省略時 : 0,0,0

モデルの表示位置。

カンマ区切りで X 座標, Y 座標, Z 座標の順番に記述する。

また、parent model alias を記述している場合、親モデルに対する相対座標として計算される。

•(x rotation),(y rotation),(z rotation) ※ 省略可能。省略時 : 0,0,0

モデルの 3D 回転。

カンマ区切りで X 軸回転, Y 軸回転, Z 軸回転の順番に記述する。

•(ON or OFF for cartoon) ※ 省略可能。省略時 : ON

トゥーンレンダリングの有効または無効。

ON=有効、OFF=無効。

- (*parent model alias*) ※ 省略可能。
基準にする親モデル名。
- (*parent bone name*) ※ 省略可能。省略時 : *CenterBone*
基準にするボーン名。
親モデルが所有しているボーン名を記述する。

▼ サンプルスクリプト

```
# メイちゃんのモデルとメイちゃんのモデルを基準に設定したメニューアイコンを表示する。
# その後、1 キーを押下することで、メイちゃんがモーションを再生しメニューアイコンが連動して動く。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0|0.0,0.0,0.0
1 2 <eps> MODEL_ADD|menu|Accessory¥menu¥menu.pmd|0.0,-7.5,0.0|0.0,0.0,0.0|ON|mei|右手首
2 2 KEY|1 MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|ONCE|ON|OFF
```

▼ 関連メッセージ

MODEL_CHANGE

MODEL_DELETE

MODEL_EVENT_ADD

MODEL_EVENT_CHANGE

MODEL_EVENT_DELETE

MODEL_CHANGE

▼説明

モデルを変更する。

モデルの変更が完了したタイミングで MODEL_EVENT_CHANGE が出力される。

▼構文

```
MODEL_CHANGE|(model alias)|(model file name)
```

[引数]

・(model alias)

変更するモデル名。

・(model file name)

変更後のモデルファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト

```
# 1 キー押下時、モデルを変更する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 MODEL_CHANGE|mei|Model¥sd_mei¥sd_mei.pmd
```

▼関連メッセージ

[MODEL_ADD](#)

[MODEL_DELETE](#)

[MODEL_EVENT_ADD](#)

[MODEL_EVENT_CHANGE](#)

[MODEL_EVENT_DELETE](#)

MODEL_DELETE

▼説明

モデルを削除する。

モデルの削除が完了したタイミングで MODEL_EVENT_DELETE が出力される。

▼構文

```
MODEL_DELETE|(model alias)
```

[引数]

・(*model alias*)

削除するモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、モデルを削除する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 MODEL_DELETE|mei
```

▼関連メッセージ

[MODEL_ADD](#)

[MODEL_CHANGE](#)

[MODEL_EVENT_ADD](#)

[MODEL_EVENT_CHANGE](#)

[MODEL_EVENT_DELETE](#)

MODEL_EVENT_ADD

▼説明

モデルの追加が完了したタイミングで出力される。

▼構文

```
MODEL_EVENT_ADD|(model alias)
```

[引数]

・(*model alias*)

追加されたモデル名。

▼サンプルスクリプト

```
# モデルの追加完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 MODEL_EVENT_ADD|mei SYNTH_START|mei|mei_voice_normal|追加完了
```

▼関連メッセージ

[MODEL_ADD](#)

[MODEL_CHANGE](#)

[MODEL_DELETE](#)

[MODEL_EVENT_CHANGE](#)

[MODEL_EVENT_DELETE](#)

MODEL_EVENT_CHANGE

▼説明

モデルの変更が完了したタイミングで出力される。

▼構文

```
MODEL_EVENT_CHANGE |(model alias)
```

[引数]

・(model alias)

変更されたモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、モデルを変更し完了後に音声合成する。  
0 1 <eps> MODEL_ADD |mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 MODEL_CHANGE |mei|Model¥sd_mei¥sd_mei.pmd  
2 3 MODEL_EVENT_CHANGE |mei SYNTH_START |mei|mei_voice_normal|変更完了
```

▼関連メッセージ

[MODEL_ADD](#)

[MODEL_CHANGE](#)

[MODEL_DELETE](#)

[MODEL_EVENT_ADD](#)

[MODEL_EVENT_DELETE](#)

MODEL_EVENT_DELETE

▼説明

モデルの削除が完了したタイミングで出力される。

▼構文

```
MODEL_EVENT_DELETE|(model alias)
```

[引数]

・(*model alias*)

削除されたモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、モデルを削除し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 MODEL_DELETE|mei
2 3 MODEL_EVENT_DELETE|mei SYNTH_START|mei|mei_voice_normal|削除完了
```

▼関連メッセージ

[MODEL_ADD](#)

[MODEL_CHANGE](#)

[MODEL_DELETE](#)

[MODEL_EVENT_ADD](#)

[MODEL_EVENT_CHANGE](#)

MOTION_ADD

▼説明

モデルに対してモーションを追加する。

モーションの追加が完了したタイミングで MOTION_EVENT_ADD、追加されたモーションの再生が終了したタイミングで MOTION_EVENT_DELETE が出力される。

▼構文

```
MOTION_ADD|(model alias)|(motion alias)|(motion file name)|(FULL or PART)|(ONCE or LOOP)|(ON or OFF for smooth)|(ON or OFF for repos)
```

[引数]

•(model alias)

モーションを追加するモデル名。

•(motion alias)

追加するモーションを参照するための別名。

•(motion file name)

追加するモーションファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

•(FULL or PART) ※ 省略可能。省略時 : FULL

対象ボーンの設定。

FULL=全てのボーン、PART=キーフレームが設定されているボーン。

•(ONCE or LOOP) ※ 省略可能。省略時 : ONCE

ループ再生の設定。

ONCE=1 度だけ再生、LOOP=モーションが削除されるまで再生。

なお、ループ再生を選択した場合、MOTION_EVENT_DELETE は発生しない。

・(ON or OFF for smooth) ※ 省略可能。省略時 : ON

スムージングの有効または無効。

ON=有効、OFF=無効。

・(ON or OFF for reposition) ※ 省略可能。省略時 : ON

位置補正の有効または無効。

ON=有効、OFF=無効。

▼サンプルスクリプト

```
# 1 キー押下時、モーションをループ再生する。
# 2 キー押下時、モーションを 1 度だけ再生する。

0 1 <eps>                                MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1                                  MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|LOOP
1 2 KEY|2                                  MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|ONCE
2 1 MOTION_EVENT_DELETE|mei|bye <eps>
```

▼関連メッセージ

MOTION_ACCELERATE

MOTION_CHANGE

MOTION_DELETE

MOTION_EVENT_ADD

MOTION_EVENT_ACCELERATE

MOTION_EVENT_CHANGE

MOTION_EVENT_DELETE

MOTION_ACCELERATE

▼説明

モーションの再生速度を変更する。

再生速度の変更が完了したタイミングで MOTION_EVENT_ACCELERATE が出力される。

▼構文

```
MOTION_ACCELERATE|(model alias)|(motion  
alias)|(speed)|(duration)|(specified time for end)
```

[引数]

•(model alias)

モーションの再生速度を変更するモデル名。

•(motion alias)

再生速度を変更するモーション名。

•(speed) ※ 省略可能。省略時：0

目標速度。

30 フレーム/秒を 1.0 と見なした倍率を 0 以上の数値で記述する。

•(duration) ※ 省略可能。省略時：0

現在の速度から目標速度に達するまでの時間。

秒単位で 0 以上の数値を記述する。

値が大きいくほど緩やかな変化となり、0 を指定すると即座に値が変化する。

•(specified time for end) ※ 省略可能。省略時：-1

目標速度に到達したタイミングで再生されるべきモーションの再生位置。

秒単位でモーションの最終時間以下の値を設定する。

なお、負の値を指定した場合、即座に再生速度が変化する。

▼サンプルスクリプト

```
# 1 キー押下時、モーションの再生時間を変更する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 <eps> MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|LOOP  
2 2 KEY|1 MOTION_ACCELERATE|mei|bye|2.0|0|-1
```

▼関連メッセージ

MOTION_ADD

MOTION_CHANGE

MOTION_DELETE

MOTION_EVENT_ADD

MOTION_EVENT_ACCELERATE

MOTION_EVENT_CHANGE

MOTION_EVENT_DELETE

MOTION_CHANGE

▼説明

再生中のモーションを変更する。

モーションの変更が完了したタイミングで MOTION_EVENT_CHANGE が出力される。

▼構文

```
MOTION_CHANGE |(model alias)|(motion alias)|(motion file name)
```

[引数]

•(model alias)

モーションを変更するモデル名。

•(motion alias)

変更するモーション名。

•(motion file name)

変更後のモーションファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト

```
# 1 キーまたは2 キー押下時、モーションを変更する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP|ON|OFF
2 2 KEY|1 MOTION_CHANGE|mei|base|Motion¥mei_idle¥mei_idle_think.vmd
2 2 KEY|2 MOTION_CHANGE|mei|base|Motion¥mei_idle¥mei_idle_sleep.vmd
```

▼関連メッセージ

MOTION_ADD

MOTION_ACCELERATE

MOTION_DELETE

MOTION_EVENT_ADD

MOTION_EVENT_ACCELERATE

MOTION_EVENT_CHANGE

MOTION_EVENT_DELETE

MOTION_DELETE

▼説明

再生中のモーションを削除する。

モーションの削除が完了したタイミングで MOTION_EVENT_DELETE が出力される。

▼構文

```
MOTION_DELETE |(mpdel alias)|(motion alias)
```

[引数]

・(*mpdel alias*)

モーションを削除するモデル名。

・(*motion alias*)

削除するモーション名。

▼サンプルスクリプト

```
# 1 キー押下時、モーションを削除する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP|ON|OFF  
2 3 <eps> MOTION_ADD|mei|think|Motion¥mei_idle¥mei_idle_think.vmd|FULL|LOOP|ON|OFF  
3 4 KEY|1 MOTION_DELETE|mei|think
```

▼関連メッセージ

[MOTION_ADD](#)

[MOTION_ACCELERATE](#)

[MOTION_CHANGE](#)

[MOTION_EVENT_ADD](#)

[MOTION_EVENT_ACCELERATE](#)

[MOTION_EVENT_CHANGE](#)

[MOTION_EVENT_DELETE](#)

MOTION_EVENT_ADD

▼説明

モーションの追加が完了したタイミングで出力される。

▼構文

```
MOTION_EVENT_ADD|(model alias)|(motion alias)
```

[引数]

•(model alias)

モーションが追加されたモデル名。

•(motion alias)

追加されたモーション名。

▼サンプルスクリプト

```
# 1 キー押下時、モーションを追加し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|LOOP
2 3 MOTION_EVENT_ADD|mei|bye SYNTH_START|mei|mei_voice_normal|追加完了
3 4 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[MOTION_ADD](#)

[MOTION_ACCELERATE](#)

[MOTION_CHANGE](#)

[MOTION_DELETE](#)

[MOTION_EVENT_ACCELERATE](#)

[MOTION_EVENT_CHANGE](#)

[MOTION_EVENT_DELETE](#)

MOTION_EVENT_ACCELERATE

▼説明

モーションの再生速度が変更されたタイミングで出力される。

▼構文

```
MOTION_EVENT_ACCELERATE|(model alias)|(motion alias)
```

[引数]

•(model alias)

モーションの再生速度が変更されたモデル名。

•(motion alias)

再生速度が変更されたモーション名。

▼サンプルスクリプト

```
# 1 キー押下時、モーションの再生速度を変更し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 <eps> MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|LOOP
2 3 KEY|1 MOTION_ACCELERATE|mei|bye|2.0|0|0
3 4 MOTION_EVENT_ACCELERATE|mei|bye SYNTH_START|mei|mei_voice_normal|再生速度変更
4 2 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[MOTION_ADD](#)

[MOTION_ACCELERATE](#)

[MOTION_CHANGE](#)

[MOTION_DELETE](#)

[MOTION_EVENT_ADD](#)

[MOTION_EVENT_CHANGE](#)

[MOTION_EVENT_DELETE](#)

MOTION_EVENT_CHANGE

▼説明

モーションの変更が完了したタイミングで出力される。

▼構文

```
MOTION_EVENT_CHANGE |(model alias)|(motion alias)
```

[引数]

•(model alias)

モーションが変更されたモデル名。

•(motion alias)

変更されたモーション名。

▼サンプルスクリプト

```
# 1 キーまたは 2 キー押下時、モーションを変更し完了後に音声合成する。
0 1 <eps>                                MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 <eps>                                MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 3 KEY|1                                MOTION_CHANGE|mei|base|Motion¥mei_idle¥mei_idle_think.vmd
2 3 KEY|2                                MOTION_CHANGE|mei|base|Motion¥mei_idle¥mei_idle_sleep.vmd
3 4 MOTION_EVENT_CHANGE|mei|base SYNTH_START|mei|mei_voice_normal|変更完了
4 2 SYNTH_EVENT_STOP|mei                 <eps>
```

▼関連メッセージ

[MOTION_ADD](#)

[MOTION_ACCELERATE](#)

[MOTION_CHANGE](#)

[MOTION_DELETE](#)

[MOTION_EVENT_ADD](#)

[MOTION_EVENT_ACCELERATE](#)

[MOTION_EVENT_DELETE](#)

MOTION_EVENT_DELETE

▼説明

モーションの再生が終了または削除が完了したタイミングで出力される。

▼構文

```
MOTION_EVENT_DELETE|(model alias)|(motion alias)
```

[引数]

•(model alias)

モーションが再生終了または削除されたモデル名。

•(motion alias)

再生終了または削除されたモーション名。

▼サンプルスクリプト

```
# 1 キー押下時、モーションを削除し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 3 <eps> MOTION_ADD|mei|bye|Motion¥mei_bye¥mei_bye.vmd|FULL|LOOP
3 4 KEY|1 MOTION_DELETE|mei|bye
4 5 MOTION_EVENT_DELETE|mei|bye SYNTH_START|mei|mei_voice_normal|削除完了
5 6 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[MOTION_ADD](#)

[MOTION_ACCELERATE](#)

[MOTION_CHANGE](#)

[MOTION_DELETE](#)

[MOTION_EVENT_ADD](#)

[MOTION_EVENT_ACCELERATE](#)

[MOTION_EVENT_CHANGE](#)

MOVE_START

▼説明

モデルを移動させる。

モデルの移動開始タイミングで MOVE_EVENT_START、移動完了タイミングで MOVE_EVENT_STOP が出力される。

▼構文

```
MOVE_START|(model alias)|(x position),(y position),(z position)|(GLOBAL  
or LOCAL position)|(move speed)
```

[引数]

•(model alias)

移動するモデル名。

•(x position),(y position),(z position)

移動先の座標。

カンマ区切りで X 座標,Y 座標,Z 座標の順番に記述する。

•(GLOBAL or LOCAL position) ※ 省略可能。省略時 : GLOBAL

絶対座標または相対座標。

GLOBAL=原点を中心とした絶対座標、LOCAL=現在のモデル位置を中心とした相対座標。

•(move speed) ※ 省略可能。省略時 : -1

移動速度。

-1 を設定した場合、速い速度かつ滑らかに移動する。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、絶対座標で左右にモデルを移動させる。  
# 4 キーまたは 6 キー押下時、相対座標で左右にモデルを移動させる。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 MOVE_START|mei|-5.0,0.0,0.0|GLOBAL|1.0  
1 1 KEY|3 MOVE_START|mei|5.0,0.0,0.0|GLOBAL|1.0  
1 1 KEY|4 MOVE_START|mei|-5.0,0.0,0.0|LOCAL|1.0  
1 1 KEY|6 MOVE_START|mei|5.0,0.0,0.0|LOCAL|1.0
```

▼関連メッセージ

MOVE_STOP

MOVE_EVENT_START

MOVE_EVENT_STOP

MOVE_STOP

▼説明

モデルの移動を停止させる。

モデルの移動停止タイミングで MOVE_EVENT_STOP が出力される。

▼構文

```
MOVE_STOP|(model alias)
```

[引数]

・(model alias)

移動を停止するモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを左右に移動する。  
# その後、移動中に 0 キーを押下すると移動を停止する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 MOVE_START|mei|-5.0,0.0,0.0|LOCAL|1.0  
1 1 KEY|3 MOVE_START|mei|5.0,0.0,0.0|LOCAL|1.0  
1 1 KEY|0 MOVE_STOP|mei
```

▼関連メッセージ

[MOVE_START](#)

[MOVE_EVENT_START](#)

[MOVE_EVENT_STOP](#)

MOVE_EVENT_START

▼説明

モデルの移動が開始したタイミングで出力される。

▼構文

```
MOVE_EVENT_START|(model alias)
```

[引数]

・(model alias)

移動が開始されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを左右に移動し、  
# モデルの移動開始を待った後、0 キー押下の移動停止を受け付ける。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 MOVE_START|mei|-5.0,0.0,0.0|LOCAL|1.0  
1 2 KEY|3 MOVE_START|mei|5.0,0.0,0.0|LOCAL|1.0  
2 3 MOVE_EVENT_START|mei <eps>  
3 1 KEY|0 MOVE_STOP|mei  
3 1 MOVE_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[MOVE_START](#)

[MOVE_STOP](#)

[MOVE_EVENT_STOP](#)

MOVE_EVENT_STOP

モデルの移動が完了または停止したタイミングで出力される。

▼構文

```
MOVE_EVENT_STOP|(model alias)
```

[引数]

・(model alias)

移動が完了または停止されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを左右に移動し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 MOVE_START|mei|-5.0,0.0,0.0|LOCAL|1.0
1 2 KEY|3 MOVE_START|mei|5.0,0.0,0.0|LOCAL|1.0
2 3 MOVE_EVENT_STOP|mei SYNTH_START|mei|mei_voice_normal|移動完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[MOVE_START](#)

[MOVE_STOP](#)

[MOVE_EVENT_START](#)

TURN_START

▼説明

モデルを回転させる。

モデルの回転開始タイミングで TURN_EVENT_START、回転完了タイミングで TURN_EVENT_STOP が出力される。

▼構文

```
TURN_START|(model alias)|(x position),(y position),(z position)|(GLOBAL  
or LOCAL position)|(rotation speed)
```

[引数]

•(model alias)

回転するモデル名。

•(x position),(y position),(z position)

XYZ 方向の回転。

カンマ区切りで X 方向の回転,Y 方向の回転,Z 方向の回転の順番に記述する。

•(GLOBAL or LOCAL position) ※ 省略可能。省略時 : GLOBAL

絶対座標または相対座標。

GLOBAL=原点を中心とした絶対座標、LOCAL=現在のモデル位置を中心とした相対座標。

•(rotation speed) ※ 省略可能。省略時 : -1

回転速度。

-1 を設定した場合、速い速度かつ滑らかに回転する。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、絶対座標で X 方向にモデルを回転させる。  
# 4 キーまたは 6 キー押下時、相対座標で X 方向にモデルを回転させる。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 TURN_START|mei|-5.0,0.0,0.0|GLOBAL|5.0  
1 1 KEY|3 TURN_START|mei|5.0,0.0,0.0|GLOBAL|5.0  
1 1 KEY|4 TURN_START|mei|-5.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|6 TURN_START|mei|5.0,0.0,0.0|LOCAL|5.0
```

▼関連メッセージ

TURN_STOP

TURN_EVENT_START

TURN_EVENT_STOP

TURN_STOP

▼説明

モデルの回転を停止させる。

モデルの回転停止タイミングで TURN_EVENT_STOP が出力される。

▼構文

```
TURN_STOP|(model alias)
```

[引数]

・(model alias)

回転を停止するモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを X 方向に回転する。  
# その後、回転中に 0 キーを押下すると回転を停止する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 TURN_START|mei|-5.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|3 TURN_START|mei|5.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|0 TURN_STOP|mei
```

▼関連メッセージ

[TURN_START](#)

[TURN_EVENT_START](#)

[TURN_EVENT_STOP](#)

TURN_EVENT_START

▼説明

モデルの回転が開始したタイミングで出力される。

▼構文

```
TURN_EVENT_START|(model alias)
```

[引数]

・(model alias)

回転が開始されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを X 方向に回転し、  
# モデルの回転開始を待った後、0 キー押下の回転停止を受け付ける。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 TURN_START|mei|-5.0,0.0,0.0|LOCAL|5.0  
1 2 KEY|3 TURN_START|mei|5.0,0.0,0.0|LOCAL|5.0  
2 3 TURN\_EVENT\_START|mei <eps>  
3 1 KEY|0 TURN_STOP|mei  
3 1 TURN\_EVENT\_STOP|mei <eps>
```

▼関連メッセージ

[TURN_START](#)

[TURN_STOP](#)

[TURN_EVENT_STOP](#)

TURN_EVENT_STOP

▼説明

モデルの回転が完了または停止したタイミングで出力される。

▼構文

```
TURN_EVENT_STOP|(model alias)
```

[引数]

・(model alias)

回転が完了または停止されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、モデルを X 方向に回転し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 TURN_START|mei|-5.0,0.0,0.0|LOCAL|5.0
1 2 KEY|3 TURN_START|mei|5.0,0.0,0.0|LOCAL|5.0
2 3 TURN_EVENT_STOP|mei SYNTH_START|mei|mei_voice_normal|回転完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[TURN_START](#)

[TURN_STOP](#)

[TURN_EVENT_START](#)

ROTATE_START

▼説明

モデルを XYZ 軸基準で回転させる。

モデルの回転開始タイミングで ROTATE_EVENT_START、回転完了タイミングで ROTATE_EVENT_STOP が出力される。

▼構文

```
ROTATE_START|(model alias)|(x rotation),(y rotaion),(z rotation)|(GLOBAL  
or LOCAL rotation)|(rotation speed)
```

[引数]

•(model alias)

回転するモデル名。

•(x rotation),(y rotaion),(z rotation)

XYZ 軸に対する回転。

カンマ区切りで X 軸回転,Y 軸回転,Z 軸回転の順番に記述する。

•(GLOBAL or LOCAL rotation) ※ 省略可能。省略時 : GLOBAL

絶対座標または相対座標。

GLOBAL=原点を中心とした絶対座標、LOCAL=現在のモデル位置を中心とした相対座標。

•(rotation speed) ※ 省略可能。省略時 : -1

回転速度。

-1 を設定した場合、速い速度かつ滑らかに回転する。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、X 軸に対して絶対座標でモデルを回転させる。  
# 4 キーまたは 6 キー押下時、X 軸に対して相対座標でモデルを回転させる。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 ROTATE_START|mei|-30.0,0.0,0.0|GLOBAL|5.0  
1 1 KEY|3 ROTATE_START|mei|30.0,0.0,0.0|GLOBAL|5.0  
1 1 KEY|4 ROTATE_START|mei|-30.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|6 ROTATE_START|mei|30.0,0.0,0.0|LOCAL|5.0
```

▼関連メッセージ

ROTATE_STOP

ROTATE_EVENT_START

ROTATE_EVENT_STOP

ROTATE_STOP

▼説明

モデルの回転を停止させる。

回転停止タイミングで ROTATE_EVENT_STOP が出力される。

▼構文

```
ROTATE_STOP|(model alias)
```

[引数]

・(model alias)

回転を停止するモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、X 軸に対してモデルを回転する。  
# その後、回転中に 0 キーを押下すると回転を停止する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 ROTATE_START|mei|-30.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|3 ROTATE_START|mei|30.0,0.0,0.0|LOCAL|5.0  
1 1 KEY|0 ROTATE_STOP|mei
```

▼関連メッセージ

[ROTATE_START](#)

[ROTATE_EVENT_START](#)

[ROTATE_EVENT_STOP](#)

ROTATE_EVENT_START

▼説明

モデルの回転が開始したタイミングで出力される。

▼構文

```
ROTATE_EVENT_START|(model alias)
```

[引数]

・(model alias)

回転が開始されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、X 軸に対してモデルを回転し、  
# モデルの回転開始を待った後、0 キー押下の回転停止を受け付ける。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 2 KEY|1 ROTATE_START|mei|-30.0,0.0,0.0|LOCAL|5.0  
1 2 KEY|3 ROTATE_START|mei|30.0,0.0,0.0|LOCAL|5.0  
2 3 ROTATE_EVENT_START|mei <eps>  
3 1 KEY|0 ROTATE_STOP|mei  
3 1 ROTATE_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[ROTATE_START](#)

[ROTATE_STOP](#)

[ROTATE_EVENT_STOP](#)

ROTATE_EVENT_STOP

▼説明

モデルの回転が完了または停止したタイミングで出力される。

▼構文

```
ROTATE_EVENT_STOP|(model alias)
```

[引数]

・(model alias)

回転が完了または停止されたモデル名。

▼サンプルスクリプト

```
# 1 キーまたは 3 キー押下時、X 軸に対してモデルを回転し完了時に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 ROTATE_START|mei|-30.0,0.0,0.0|LOCAL|5.0
1 2 KEY|3 ROTATE_START|mei|30.0,0.0,0.0|LOCAL|5.0
2 3 ROTATE_EVENT_STOP|mei SYNTH_START|mei|mei_voice_normal|回転完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[ROTATE_START](#)

[ROTATE_STOP](#)

[ROTATE_EVENT_START](#)

SOUND_START

▼説明

WAV ファイルまたは MP3 ファイルを再生する。

サウンドの再生開始タイミングで SOUND_EVENT_START、再生終了タイミングで

SOUND_EVENT_STOP が出力される。また、WAV ファイル再生時に同名の [lab ファイル](#) が存在する場合、モデルが自動的にリップシンクする。

▼構文

```
SOUND_START|(sound alias)|(sound file name)
```

[引数]

•(sound alias)

再生するサウンドを参照するための別名。

•(sound file name)

再生するサウンドファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト

```
# 1 キー押下時、サウンドを再生する。  
0 1 KEY|1 SOUND_START|sound|Sound¥sample.wav
```

▼関連メッセージ

[SOUND_STOP](#)

[SOUND_EVENT_START](#)

[SOUND_EVENT_STOP](#)

SOUND_STOP

▼説明

サウンドを停止させる。

再生停止タイミングで SOUND_EVENT_STOP が出力される。

▼構文

```
SOUND_STOP|(sound alias)
```

[引数]

・(*sound alias*)

停止するサウンド名。

▼サンプルスクリプト

```
# 1 キー押下時、再生中のサウンドを停止させる。  
0 1 <eps> SOUND_START|sound|Sound¥sample.wav  
1 2 KEY|1 SOUND_STOP|sound
```

▼関連メッセージ

[SOUND_START](#)

[SOUND_EVENT_START](#)

[SOUND_EVENT_STOP](#)

SOUND_EVENT_START

▼説明

サウンドの再生が開始したタイミングで出力される。

▼構文

```
SOUND_EVENT_START|(sound alias)
```

[引数]

・(sound alias)

再生が開始されたサウンド名。

▼サンプルスクリプト

```
# 1 キー押下時、サウンドを再生し SOUND_EVENT_START の出力を待つ。  
# その後、2 キー押下のサウンド停止を受け付ける。  
0 1 KEY|1          SOUND_START|sound|Sound¥sample.wav  
1 2 SOUND_EVENT_START|sound <eps>  
2 3 KEY|2          SOUND_STOP|sound
```

▼関連メッセージ

SOUND_START

SOUND_STOP

SOUND_EVENT_STOP

SOUND_EVENT_STOP

▼説明

サウンドの再生が終了または停止したタイミングで出力される。

▼構文

```
SOUND_EVENT_STOP|(sound alias)
```

[引数]

•(sound alias)

再生が終了または停止されたサウンド名。

▼サンプルスクリプト

```
# 1 キー押下時、2 つのサウンドを順番に再生する。  
0 1 KEY|1 SOUND_START|sound|Sound¥sample-01.wav  
1 2 SOUND_EVENT_STOP|sound SOUND_START|sound|Sound¥sample-02.wav  
2 0 SOUND_EVENT_STOP|sound <eps>
```

▼関連メッセージ

SOUND_START

SOUND_STOP

SOUND_EVENT_START

STAGE

▼説明

ステージファイルを設定する。

PMD 形式のステージファイルまたはフロア画像と背景画像に分離された BMP ファイルを指定する 2 種類の構文が存在する。

▼構文①

```
STAGE |(stage file name)
```

[引数]

・(stage file name)

ステージ用 PMD ファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト①

```
# PMD ファイルをステージに設定する。  
0 1 <eps> STAGE |Stage¥sample.pmd
```

▼構文②

```
STAGE|(bitmap file name for floor),(bitmap file name for background)
```

[引数]

•(bitmap file name for floor)

フロア用 BMP ファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

•(bitmap file name for background)

背景用 BMP ファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト②

```
# BMP ファイルをステージに設定する。  
0 1 <eps> STAGE|Stage¥building2¥floor.bmp,Stage¥building2¥background.bmp
```

LIGHTCOLOR

▼説明

光源の色を設定する。

▼構文

```
LIGHTCOLOR | (red), (green), (blue)
```

[引数]

・(red),(green),(blue)

光源の色。

カンマ区切りで赤、緑、青の順番に 0.0～1.0 の範囲で記述する。

▼サンプルスクリプト

```
# 1 キー押下時、光源の色を赤に設定する。  
# 2 キー押下時、光源の色を緑に設定する。  
# 3 キー押下時、光源の色を青に設定する。  
0 1 <eps> MODEL_ADD | mei | Model¥mei¥mei.pmd | 0.0,0.0,0.0  
1 1 KEY|1 LIGHTCOLOR|1.0,0.0,0.0  
1 1 KEY|2 LIGHTCOLOR|0.0,1.0,0.0  
1 1 KEY|3 LIGHTCOLOR|0.0,0.0,1.0
```

▼関連メッセージ

[LIGHTDIRECTION](#)

LIGHTDIRECTION

▼説明

光源の位置を設定する。

▼構文

```
LIGHTDIRECTION|(x position),(y position),(z position)
```

[引数]

・(x position),(y position),(z position)

光源の位置。

カンマ区切りで X 座標, Y 座標, Z 座標の順番に記述する。

▼サンプルスクリプト

```
# 1 キー押下時、光源の位置を変更する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0  
1 1 KEY|1 LIGHTDIRECTION|0.0,10.0,0.0
```

▼関連メッセージ

[LIGHTCOLOR](#)

CAMERA

▼説明

カメラを設定する。

カメラ位置、回転、距離、画角、変化時間を記述または VMD 形式のモーションファイルを設定する 2 種類の構文が存在する。

▼構文①

```
CAMERA|(x position),(y position),(z position)|(x rotation),(y rotation),(z rotation)|(distance)|(fovy)|(time)
```

[引数]

•(x position),(y position),(z position)

カメラの位置。

カンマ区切りで X 座標, Y 座標, Z 座標の順番に記述する。

•(x rotation),(y rotation),(z rotation)

カメラの 3D 回転。

カンマ区切りで X 軸回転, Y 軸回転, Z 軸回転の順番に記述する。

•(distance)

カメラと注視点の距離。

•(fovy)

カメラの画角。

•(time)

変化に必要な時間。単位は秒。

▼サンプルスクリプト①

```
# 1 キー押下時、カメラと注視点の距離を遠ざける。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 1 KEY|1 CAMERA|0.0,13.0,0.0|0.0,0.0,0.0|300|16.0|0
```

▼構文②

```
CAMERA|(motion file name)
```

[引数]

•(motion file name)

カメラ用モーションファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト②

```
# 1 キー押下時、VMD 形式のカメラモーションを設定する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0  
1 1 KEY|1 CAMERA|Motion¥Camera¥camera.vmd
```

▼備考

1. 現在のカメラ設定の確認方法

デバッグ画面（D キー押下）を表示することで確認することができる。

RECOG_EVENT_START

▼説明

音声認識を開始したタイミングで出力される。

▼構文

```
RECOG_EVENT_START
```

▼サンプルスクリプト

```
# 音声認識が開始されたタイミングで音声合成する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0  
1 2 RECOG_EVENT_START SYNTH_START|mei|mei_voice_normal|音声認識開始  
2 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[RECOG_EVENT_STOP](#)

RECOG_EVENT_STOP

▼説明

音声認識が完了したタイミングで出力される。

▼構文

```
RECOG_EVENT_STOP | (word sequence)
```

[引数]

• (word sequence)

音声認識の結果。

複数の単語を組み合わせた AND 条件を記述する場合、条件となる単語をカンマ区切りで記述する。

また、[辞書ファイル](#)に記述された単語のみが出力される。

▼サンプルスクリプト

```
# 「こんにちは」または「こんにちわ」を音声認識した場合、「こんにちは」を音声合成する。
# 「何してるの」を音声認識した場合、「案内をしています」を音声合成する。
0 1 <eps>                                MODEL_ADD |mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 RECOG_EVENT_STOP|こんにちは          SYNTH_START|mei|mei_voice_normal|こんにちは
1 2 RECOG_EVENT_STOP|こんにちわ          SYNTH_START|mei|mei_voice_normal|こんにちは
2 1 SYNTH_EVENT_STOP|mei                  <eps>
1 3 RECOG_EVENT_STOP|何,し,てる,の       SYNTH_START|mei|mei_voice_normal|案内をしています
3 1 SYNTH_EVENT_STOP|mei                  <eps>
```

▼備考

1. 認識されている単語の確認方法
デバッグ画面（D キー押下）を表示することで確認することができる。
2. 複数の単語を OR 条件で認識させる方法
状態番号が同じ行を複数記述することで OR 条件を指定できる。

▼関連メッセージ

[RECOG_EVENT_START](#)

RECOG_MODIFY|GAIN

▼説明

音声入力の振幅調整パラメータを設定する。

設定完了タイミングで RECOG_EVENT_MODIFY|GAIN が出力される。

▼構文

```
RECOG_MODIFY|GAIN|(ratio)
```

[引数]

・(ratio)

音声認識の振幅調整パラメータ。

単位は比率。

[備考]

入力音声はデバイスから録音された後、振幅値にこの値が倍率としてかけられ、その後音声認識処理が行われる。既定値は 1.0（調整なし）となり、0 を与えると音声入力をミュートにすることができる。

▼サンプルスクリプト

```
# 1 キー押下時、音声入力をミュートに変更する。  
0 1 KEY|1 RECOG_MODIFY|GAIN|0
```

▼関連メッセージ

[RECOG_EVENT_MODIFY|GAIN](#)

RECOG_EVENT_MODIFY|GAIN

▼説明

音声入力の振幅調整パラメータ設定が完了したタイミングで出力される。

▼構文

```
RECOG_MODIFY|GAIN
```

▼サンプルスクリプト

```
# 1 キー押下時、音声入力をミュートに変更し完了後に音声合成する。  
0 1 KEY|1 RECOG_MODIFY|GAIN|0  
1 2 RECOG_EVENT_MODIFY|GAIN SYNTH_START|mei|mei_voice_normal|設定完了
```

▼関連メッセージ

[RECOG_MODIFY|GAIN](#)

RECOG_MODIFY|USERDICT_SET

▼説明

MMDAgent で使用するユーザー辞書ファイルを変更する。

変更完了タイミングで RECOG_EVENT_MODIFY|USERDICT_SET が出力される。

▼構文

```
RECOG_MODIFY|USERDICT_SET|(dictionary file name)
```

[引数]

・(dictionary file name)

変更後の辞書ファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト

```
# 1 キー押下時、ユーザー辞書を変更する。  
0 1 KEY|1 RECOG_MODIFY|USERDICT_SET|Test.dic
```

▼関連メッセージ

RECOG_MODIFY|USERDICT_UNSET

RECOG_EVENT_MODIFY|USERDICT_SET

RECOG_EVENT_MODIFY|USERDICT_UNSET

RECOG_EVENT_MODIFY|USERDICT_SET

▼説明

ユーザー辞書ファイルの変更が完了したタイミングで出力される。

▼構文

```
RECOG_EVENT_MODIFY|USERDICT_SET
```

▼サンプルスクリプト

```
# 1 キー押下時、ユーザー辞書を変更し完了後に音声合成する。
0 1 KEY|1 RECOG_MODIFY|USERDICT_SET|Test.dic
1 2 RECOG_EVENT_MODIFY|USERDICT_SET SYNTH_START|mei|mei_voice_normal|変更完了
```

▼関連メッセージ

[RECOG_MODIFY|USERDICT_SET](#)

[RECOG_MODIFY|USERDICT_UNSET](#)

[RECOG_EVENT_MODIFY|USERDICT_UNSET](#)

RECOG_MODIFY|USERDICT_UNSET

▼説明

MMDAgent が読み込んでいるユーザー辞書ファイルをメモリから消去する。

消去完了タイミングで RECOG_EVENT_MODIFY|USERDICT_UNSET が出力される。

▼構文

```
RECOG_MODIFY|USERDICT_UNSET
```

▼サンプルスクリプト

```
# 1 キー押下時、ユーザー辞書をメモリから消去する。  
0 1 KEY|1 RECOG_MODIFY|USERDICT_UNSET
```

▼関連メッセージ

RECOG_MODIFY|USERDICT_SET

RECOG_EVENT_MODIFY|USERDICT_SET

RECOG_EVENT_MODIFY|USERDICT_UNSET

RECOG_EVENT_MODIFY|USERDICT_UNSET

▼説明

ユーザー辞書ファイルをメモリから消去したタイミングで出力される。

▼構文

```
RECOG_EVENT_MODIFY|USERDICT_UNSET
```

▼サンプルスクリプト

```
# 1 キー押下時、ユーザー辞書をメモリから消去し完了後に音声合成する。  
0 1 KEY|1 RECOG_MODIFY|USERDICT_UNSET  
1 2 RECOG_EVENT_MODIFY|USERDICT_UNSET SYNTH_START|mei|mei_voice_normal|消去完了
```

▼関連メッセージ

RECOG_MODIFY|USERDICT_SET

RECOG_MODIFY|USERDICT_UNSET

RECOG_EVENT_MODIFY|USERDICT_SET

SYNTH_START

▼説明

リアルタイム音声合成を実行し、モデルに発声させる。

音声合成の開始タイミングで SYNTH_EVENT_START、終了タイミングで SYNTH_EVENT_STOP が出力される。また、セリフ発声時は LIPSYNC_START が発生し、指定したモデルが自動的にリップシンクする。

▼構文

```
SYNTH_START|(model alias)|(voice alias)|(synthesized text)
```

[引数]

•(model alias)

発声するモデル名。

•(voice alias)

使用する発話スタイル名。

[ojt ファイル](#)に定義した名前を記述する。

•(synthesized text)

発声するセリフ。100 文字以内を推奨。

音声合成に要する時間は、セリフの長さや PC 性能によって決まる。

※注意※

バージョン 1.5 において、セリフの長さが 223 文字を超える場合、実行時に MMDAgent が強制終了する。

▼サンプルスクリプト

```
# 1 キー押下時、mei_voice_normal を使用し「こんにちは」を音声合成する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0  
1 2 KEY|1 SYNTH_START|mei|mei_voice_normal|こんにちは。  
2 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

SYNTH_STOP

SYNTH_EVENT_START

SYNTH_EVENT_STOP

LIPSYNC_START

SYNTH_STOP

▼説明

音声合成を停止させる。

音声合成の停止タイミングで SYNTH_EVENT_STOP が出力される。

▼構文

```
SYNTH_STOP|(model alias)
```

[引数]

・(*model alias*)

発声を停止するモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、音声合成を開始する。
# その後、音声合成中に 2 キーを押下すると音声合成を停止する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1 SYNTH_START|mei|mei_voice_normal|こんにちは。私の名前はメイです。
2 1 SYNTH_EVENT_STOP|mei <eps>
2 3 KEY|2 SYNTH_STOP|mei
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[SYNTH_START](#)

[SYNTH_EVENT_START](#)

[SYNTH_EVENT_STOP](#)

SYNTH_EVENT_START

▼説明

音声合成の開始タイミングで出力される。

▼構文

```
SYNTH_EVENT_START | (model alias)
```

[引数]

・(model alias)

発声が開始されたモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、音声合成を実行し SYNTH_EVENT_START の出力を待つ。  
# その後、2 キー押下の音声合成停止を受け付ける。  
0 1 <eps> MODEL_ADD |mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0  
1 2 KEY|1 SYNTH_START |mei|mei_voice_normal|こんにちは。私の名前はメイです。  
2 3 SYNTH_EVENT_START |mei <eps>  
3 1 SYNTH_EVENT_STOP |mei <eps>  
3 4 KEY|2 SYNTH_STOP |mei  
4 1 SYNTH_EVENT_STOP |mei <eps>
```

▼関連メッセージ

[SYNTH_START](#)

[SYNTH_STOP](#)

[SYNTH_EVENT_STOP](#)

SYNTH_EVENT_STOP

▼説明

音声合成の終了または停止タイミングで出力される。

▼構文

```
SYNTH_EVENT_STOP|(model alias)
```

[引数]

・(model alias)

発声が終了または停止されたモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、「はじめまして！」を音声合成する。その後、音声合成が完了した後、  
# 「私の名前はメイと言います。」を音声合成する。  
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0  
1 2 KEY|1 SYNTH_START|mei|mei_voice_normal|はじめまして！  
2 3 SYNTH_EVENT_STOP|mei SYNTH_START|mei|mei_voice_normal|私の名前はメイと言います。  
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[SYNTH_START](#)

[SYNTH_STOP](#)

[SYNTH_EVENT_START](#)

LIPSYNC_START

▼説明

モデルにリップシンクさせる。

リップシンクの開始タイミングで LIPSYNC_EVENT_START、終了タイミングで

LIPSYNC_EVENT_STOP が出力される。通常は SYNTH_START のセリフ発声時に自動的に呼びだされるが、本メッセージでリップシンクのみを実行することができる。

▼構文

```
LIPSYNC_START|(model alias)|(phoneme and millisecond pair sequence)
```

[引数]

・(model alias)

リップシンクさせるモデル名。

・(phoneme and millisecond pair sequence)

音素と持続時間。

持続時間の単位はミリ秒。

[記述例]

「おはよう」をリップシンクする場合の記述例を以下に記す。

```
o,105,h,65,a,60,y,40,o,110,o,170
```

▼サンプルスクリプト

```
# 1 キー押下時、「おはよう」のリップシンクを開始する。
```

```
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
```

```
1 1 KEY|1 LIPSYNC_START|mei|o,105,h,65,a,60,y,40,o,110,o,170
```

```
1 1 LIPSYNC_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[SYNTH_START](#)

[LIPSYNC_STOP](#)

[LIPSYNC_EVENT_START](#)

[LIPSYNC_EVENT_STOP](#)

LIPSYNC_STOP

▼説明

リップシンクを停止させる。

リップシンクの停止タイミングで LIPSYNC_EVENT_STOP が出力される。

▼構文

```
LIPSYNC_STOP|(model alias)
```

[引数]

・(model alias)

リップシンクを停止させるモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、リップシンクを開始し、
# 2 キー押下のリップシンク停止を受け付ける。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,0.0
1 2 KEY|1 LIPSYNC_START|mei|o,105,h,65,a,60,y,40,o,110,o,170
2 1 LIPSYNC_EVENT_STOP|mei <eps>
2 3 KEY|2 LIPSYNC_STOP|mei
3 1 LIPSYNC_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[LIPSYNC_START](#)

[LIPSYNC_EVENT_START](#)

[LIPSYNC_EVENT_STOP](#)

LIPSYNC_EVENT_START

▼説明

リップシンクの開始タイミングで出力される。

▼構文

```
LIPSYNC_EVENT_START | (model alias)
```

[引数]

・(model alias)

リップシンクが開始したモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、リップシンク開始と同時にモーションを再生する。
0 1 <eps>                                MODEL_ADD |mei |Model¥mei¥mei.pmd |0.0,0.0,0.0
1 2 KEY |1                                LIPSYNC_START |mei |o,105,h,65,a,60,y,40,o,110,o,170
2 3 LIPSYNC_EVENT_START |mei             MOTION_ADD |mei |bye |Motion¥mei_bye¥mei_bye.vmd |PART |ONCE
3 1 MOTION_EVENT_DELETE |mei |bye <eps>
```

▼関連メッセージ

[LIPSYNC_START](#)

[LIPSYNC_STOP](#)

[LIPSYNC_EVENT_STOP](#)

LIPSYNC_EVENT_STOP

▼説明

リップシンクの終了または停止タイミングで出力される。

▼構文

```
LIPSYNC_EVENT_STOP | (model alias)
```

[引数]

・(model alias)

リップシンクが終了または停止されたモデル名。

▼サンプルスクリプト

```
# 1 キー押下時、リップシンクを開始し、リップシンク終了時にモーションを再生する。
0 1 <eps>                                MODEL_ADD |mei |Model¥mei¥mei.pmd |0.0,0.0,0.0
1 2 KEY |1                                LIPSYNC_START |mei |o,105,h,65,a,60,y,40,o,110,o,170
2 3 LIPSYNC_EVENT_STOP |mei              MOTION_ADD |mei |bye |Motion¥mei_bye¥mei_bye.vmd |PART |ONCE
3 1 MOTION_EVENT_DELETE |mei |bye <eps>
```

▼関連メッセージ

[LIPSYNC_START](#)

[LIPSYNC_STOP](#)

[LIPSYNC_EVENT_START](#)

VALUE_SET

▼説明

変数を設定する。

任意の値を指定または乱数（少数含む）を生成する 2 種類のメッセージが存在する。

変数に値の設定が完了したタイミングで VALUE_EVENT_SET が出力される。

▼構文①

```
VALUE_SET|(variable alias)|(value)
```

[引数]

•(variable alias)

変数を参照するための別名。

•(value)

変数に設定する値。

▼サンプルスクリプト①

```
# 1 キー押下時、変数を設定する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1 VALUE_SET|var|0
2 3 VALUE_EVENT_SET|var SYNTH_START|mei|mei_voice_normal|設定完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼構文②

```
VALUE_SET|(variable alias)|(minimum value for random)|(maximum value for random)
```

[引数]

•(variable alias)

変数を参照するための別名。

•(minimum value for random)

生成する乱数の最小値。

•(maximum value for random)

生成する乱数の最大値。

▼サンプルスクリプト②

```
# 1 キー押下時、変数にランダムな値を設定する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1 VALUE_SET|var|0|100
2 3 VALUE_EVENT_SET|var SYNTH_START|mei|mei_voice_normal|設定完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[VALUE_UNSET](#)

[VALUE_EVAL](#)

[VALUE_EVENT_SET](#)

[VALUE_EVENT_UNSET](#)

[VALUE_EVENT_EVAL](#)

VALUE_UNSET

▼説明

変数を削除する。

削除完了タイミングで VALUE_EVENT_UNSET が出力される。

▼構文

```
VALUE_UNSET|(variable alias)
```

[引数]

・(variable alias)

削除する変数名。

▼サンプルスクリプト

```
# 起動時に変数 var を設定。その後、1 キーを押下すると変数 var を削除する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps> VALUE_SET|var|0
2 3 KEY|1 VALUE_UNSET|var
3 4 VALUE_EVENT_UNSET|var SYNTH_START|mei|mei_voice_normal|削除完了
4 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[VALUE_SET](#)

[VALUE_EVAL](#)

[VALUE_EVENT_SET](#)

[VALUE_EVENT_UNSET](#)

[VALUE_EVENT_EVAL](#)

VALUE_EVAL

▼説明

変数と値を比較する。

比較完了タイミングで VALUE_EVENT_EVAL が出力される。

▼構文

```
VALUE_EVAL|(variable alias)|(EQ or NE or LE or LT or GE or GT for evaluation)|(value)
```

[引数]

•(variable alias)

比較する変数名。

不等号の左側に位置する。

•(EQ or NE or LE or LT or GE or GT for evaluation)

変数と値を比較する不等号。

以下の表に従い記述する。

表記	不等号
EQ	=
NE	≠
LE	≤
LT	<
GE	≥
GT	>

•(variable alias)

比較する値。

不等号の右側に位置する。

▼サンプルスクリプト

```

# 1 キー、2 キー、3 キー押下時、変数に1~3の値をセットする。
# セットされた値を比較 (EQ) し、結果に応じて音声合成のセリフを変更する。
# 1 キー：音声合成「1 がセットされています。」
# 2 キー：音声合成「2 がセットされています。」
# 3 キー：音声合成「3 がセットされています。」

0 1 <eps>                                MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1                                 VALUE_SET|v|1
1 2 KEY|2                                 VALUE_SET|v|2
1 2 KEY|3                                 VALUE_SET|v|3
2 3 VALUE_EVENT_SET|v                     VALUE_EVAL|v|EQ|1
3 9 VALUE_EVENT_EVAL|v|EQ|1|TRUE          SYNTH_START|mei|mei_voice_normal|1 がセットされています。
3 4 VALUE_EVENT_EVAL|v|EQ|1|FALSE        VALUE_EVAL|v|EQ|2
4 9 VALUE_EVENT_EVAL|v|EQ|2|TRUE          SYNTH_START|mei|mei_voice_normal|2 がセットされています。
4 5 VALUE_EVENT_EVAL|v|EQ|2|FALSE        VALUE_EVAL|v|EQ|3
5 9 VALUE_EVENT_EVAL|v|EQ|3|TRUE          SYNTH_START|mei|mei_voice_normal|3 がセットされています。
5 1 VALUE_EVENT_EVAL|v|EQ|3|FALSE        <eps>
9 1 SYNTH_EVENT_STOP|mei                 <eps>

```

▼関連メッセージ

[VALUE_SET](#)[VALUE_UNSET](#)[VALUE_EVENT_SET](#)[VALUE_EVENT_UNSET](#)[VALUE_EVENT_EVAL](#)

VALUE_EVENT_SET

▼説明

変数の設定が完了したタイミングで出力される。

▼構文

```
VALUE_EVENT_SET|(variable alias)
```

[引数]

・(variable alias)

値が設定された変数名。

▼サンプルスクリプト

```
# 1 キー押下時、変数を設定し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1 VALUE_SET|var|0
2 3 VALUE_EVENT_SET|var SYNTH_START|mei|mei_voice_normal|設定完了
3 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[VALUE_SET](#)

[VALUE_UNSET](#)

[VALUE_EVAL](#)

[VALUE_EVENT_UNSET](#)

[VALUE_EVENT_EVAL](#)

VALUE_EVENT_UNSET

▼説明

変数の削除が完了したタイミングで出力される。

▼構文

```
VALUE_EVENT_UNSET|(variable alias)
```

[引数]

・(variable alias)

削除された変数名。

▼サンプルスクリプト

```
# 起動時に変数 var を設定。その後、1 キーを押下すると変数を削除し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps> VALUE_SET|var|0
2 3 KEY|1 VALUE_UNSET|var
3 4 VALUE_EVENT_UNSET|var SYNTH_START|mei|mei_voice_normal|削除完了
4 1 SYNTH_EVENT_STOP|mei <eps>
```

▼関連メッセージ

[VALUE_SET](#)

[VALUE_UNSET](#)

[VALUE_EVAL](#)

[VALUE_EVENT_SET](#)

[VALUE_EVENT_EVAL](#)

VALUE_EVENT_EVAL

▼説明

変数と値の比較が完了したタイミングで出力される。

▼構文

```
VALUE_EVENT_EVAL|(variable alias)|(EQ or NE or LE or LT or GE or GT for
evaluation)|(value)|(TRUE or FALSE)
```

[引数]

•(variable alias)

比較された変数名。

•(EQ or NE or LE or LT or GE or GT for evaluation)

変数と値を比較した不等号。

以下の表に従い記述する。

表記	不等号
EQ	=
NE	≠
LE	≤
LT	<
GE	≥
GT	>

•(value)

比較した値。

•(TRUE or FALSE)

比較した結果。

TRUE=真、FALSE=偽。

▼サンプルスクリプト

```

# 1 キー、2 キー、3 キー押下時、変数に1~3の値をセットする。
# セットされた値を比較 (EQ) し、結果に応じて音声合成のセリフを変更する。
# 1 キー：音声合成「1 がセットされています。」
# 2 キー：音声合成「2 がセットされています。」
# 3 キー：音声合成「3 がセットされています。」

0 1 <eps>                                MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 KEY|1                                 VALUE_SET|v|1
1 2 KEY|2                                 VALUE_SET|v|2
1 2 KEY|3                                 VALUE_SET|v|3
2 3 VALUE_EVENT_SET|v                    VALUE_EVAL|v|EQ|1
3 9 VALUE_EVENT_EVAL|v|EQ|1|TRUE         SYNTH_START|mei|mei_voice_normal|1 がセットされています。
3 4 VALUE_EVENT_EVAL|v|EQ|1|FALSE       VALUE_EVAL|v|EQ|2
4 9 VALUE_EVENT_EVAL|v|EQ|2|TRUE         SYNTH_START|mei|mei_voice_normal|2 がセットされています。
4 5 VALUE_EVENT_EVAL|v|EQ|2|FALSE       VALUE_EVAL|v|EQ|3
5 9 VALUE_EVENT_EVAL|v|EQ|3|TRUE         SYNTH_START|mei|mei_voice_normal|3 がセットされています。
5 1 VALUE_EVENT_EVAL|v|EQ|3|FALSE       <eps>
9 1 SYNTH_EVENT_STOP|mei                <eps>

```

▼関連メッセージ

[VALUE_SET](#)[VALUE_UNSET](#)[VALUE_EVAL](#)[VALUE_EVENT_SET](#)[VALUE_EVENT_UNSET](#)

TIMER_START

▼説明

タイマ（一定時間後にメッセージを出力）を開始する。

タイマ開始タイミングで `TIMER_EVENT_START`、タイマ停止タイミングで `TIMER_EVENT_STOP` が出力される。

▼構文

```
TIMER_START|(count down alias)|(value)
```

[引数]

・*(count down alias)*

タイマを参照するための別名。

・*(value)*

タイマに設定する時間。

0.1 秒単位で記述する。

▼サンプルスクリプト

```
# 1 キー押下時、タイマを起動し 3 秒後に音声合成する。  
0 1 KEY|1                TIMER_START|wait|30  
1 2 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|3 秒経過
```

▼関連メッセージ

[TIMER_STOP](#)

[TIMER_EVENT_START](#)

[TIMER_EVENT_STOP](#)

TIMER_STOP

▼説明

タイマを停止させる。

停止タイミングで `TIMER_EVENT_STOP` が出力される。

▼構文

```
TIMER_STOP|(count down alias)
```

[引数]

・(count down alias)

停止させるタイマ名。

▼サンプルスクリプト

```
# 1 キー押下時、5 秒後に音声合成するタイマを起動する。  
# その後、5 秒が経過する前に 2 キーを押下するとタイマを停止させる。  
0 1 KEY|1                TIMER_START|wait|5  
1 2 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|5 秒経過  
1 3 KEY|2                TIMER_STOP|wait  
3 4 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|タイマ停止
```

▼関連メッセージ

[TIMER_START](#)

[TIMER_EVENT_START](#)

[TIMER_EVENT_STOP](#)

TIMER_EVENT_START

▼説明

タイマの開始タイミングで出力される。

▼構文

```
TIMER_EVENT_START|(count down alias)
```

[引数]

・(count down alias)

開始したタイマ名。

▼サンプルスクリプト

```
# 1 キー押下時、タイマを起動し TIMER_EVENT_START の出力を待つ。
# その後、TIMER_EVENT_STOP または 2 キー押下のタイマ停止を受け付ける。
0 1 KEY|1                TIMER_START|wait|5
1 2 TIMER_EVENT_START|wait <eps>
2 3 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|5 秒経過
2 4 KEY|2                TIMER_STOP|wait
4 5 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|タイマ停止
```

▼関連メッセージ

[TIMER_START](#)

[TIMER_STOP](#)

[TIMER_EVENT_STOP](#)

TIMER_EVENT_STOP

▼説明

タイマの停止タイミングで出力される。

▼構文

```
TIMER_EVENT_STOP|(count down alias)
```

[引数]

・(count down alias)

停止したタイマ名。

▼サンプルスクリプト

```
# 1 キー押下時、5 秒後に音声合成するタイマを起動する。  
# 5 秒が経過する前に 2 キーを押下することで強制的にタイマを停止させる。  
0 1 KEY|1          TIMER_START|wait|5  
1 2 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|5 秒経過  
1 3 KEY|2          TIMER_STOP|wait  
3 4 TIMER_EVENT_STOP|wait SYNTH_START|mei|mei_voice_normal|タイマ停止
```

▼関連メッセージ

[TIMER_START](#)

[TIMER_STOP](#)

[TIMER_EVENT_START](#)

PLUGIN_ENABLE

▼説明

プラグインを有効化する。

プラグインが有効化されたタイミングで PLUGIN_EVENT_ENABLE が出力される。

▼構文

```
PLUGIN_ENABLE | (plugin name)
```

[引数]

・(plugin name)

有効化するプラグインの名前。

▼サンプルスクリプト

```
# 1 キー押下時、LookAt プラグインを有効化する。
# 2 キー押下時、LookAt プラグインを無効化する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 2 KEY|1 PLUGIN_ENABLE|LookAt
2 2 KEY|2 PLUGIN_DISABLE|LookAt
```

▼備考

1. MMDAgent に標準搭載されている各種プラグインの名前

Audio、Julius、LookAt、Open_JTalk、Variables、VIManager、WindowController

▼関連メッセージ

[PLUGIN_DISABLE](#)

[PLUGIN_EVENT_ENABLE](#)

[PLUGIN_EVENT_DISABLE](#)

PLUGIN_DISABLE

▼説明

プラグインを無効化する。

プラグインが無効化されたタイミングで PLUGIN_EVENT_DISABLE が出力される。

▼構文

```
PLUGIN_DISABLE|(plugin name)
```

[引数]

・(*plugin name*)

無効化するプラグインの名前。

▼サンプルスクリプト

```
# 1 キー押下時、LookAt プラグインを有効化する。
# 2 キー押下時、LookAt プラグインを無効化する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 2 KEY|1 PLUGIN_ENABLE|LookAt
2 2 KEY|2 PLUGIN_DISABLE|LookAt
```

▼備考

1. MMDAgent に標準搭載されている各種プラグインの名前
Audio、Julius、LookAt、Open_JTalk、Variables、VIManager、WindowController

▼関連メッセージ

[PLUGIN_ENABLE](#)

[PLUGIN_EVENT_ENABLE](#)

[PLUGIN_EVENT_DISABLE](#)

PLUGIN_EVENT_ENABLE

▼説明

プラグインが有効化されたタイミングで出力される。

▼構文

```
PLUGIN_EVENT_ENABLE |(plugin name)
```

[引数]

・(plugin name)

有効化されたプラグインの名前

▼サンプルスクリプト

```
# 1 キー押下時、プラグインを有効化し完了後に音声合成する。
# 2 キー押下時、プラグインを無効化し完了後に音声合成する。

0 1 <eps>                                MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps>                                MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 3 KEY|1                                 PLUGIN_ENABLE|LookAt
3 2 PLUGIN\_EVENT\_ENABLE|LookAt SYNTH_START|mei|mei_voice_normal|プラグイン有効化
2 4 KEY|2                                 PLUGIN_DISABLE|LookAt
4 2 PLUGIN\_EVENT\_DISABLE|LookAt SYNTH_START|mei|mei_voice_normal|プラグイン無効化
```

▼備考

1. MMDAgent に標準搭載されている各種プラグインの名前
Audio、Julius、LookAt、Open_JTalk、Variables、VIManager、WindowController

▼関連メッセージ

[PLUGIN_ENABLE](#)

[PLUGIN_DISABLE](#)

[PLUGIN_EVENT_DISABLE](#)

PLUGIN_EVENT_DISABLE

▼説明

プラグインが無効化されたタイミングで出力される。

▼構文

```
PLUGIN_EVENT_DISABLE|(plugin name)
```

[引数]

・(plugin name)

無効化されたプラグインの名前。

▼サンプルスクリプト

```
# 1 キー押下時、プラグインを有効化し完了後に音声合成する。
# 2 キー押下時、プラグインを無効化し完了後に音声合成する。
0 1 <eps> MODEL_ADD|mei|Model¥mei¥mei.pmd|0.0,0.0,-14.0
1 2 <eps> MOTION_ADD|mei|base|Motion¥mei_wait¥mei_wait.vmd|FULL|LOOP
2 3 KEY|1 PLUGIN_ENABLE|LookAt
3 2 PLUGIN_EVENT_ENABLE|LookAt SYNTH_START|mei|mei_voice_normal|プラグイン有効化
2 4 KEY|2 PLUGIN_DISABLE|LookAt
4 2 PLUGIN_EVENT_DISABLE|LookAt SYNTH_START|mei|mei_voice_normal|プラグイン無効化
```

▼備考

1. MMDAgent に標準搭載されている各種プラグインの名前
Audio、Julius、LookAt、Open_JTalk、Variables、VIManager、WindowController

▼関連メッセージ

[PLUGIN_ENABLE](#)

[PLUGIN_DISABLE](#)

[PLUGIN_EVENT_ENABLE](#)

DRAGANDDROP

▼説明

MMDAgent にファイルがドラッグ&ドロップされたタイミングで出力される。

MMDAgent には特定ファイルがドラッグ&ドロップされると、[ファイルの種類に応じた動作](#)を行う機能が実装されている。なお、本メッセージではドロップされたファイルのフルパスと合わせて、ドロップされた位置の座標が出力されるため、このメッセージを fst ファイル内の遷移条件として記述することは難しい。そのため、サンプルスクリプトの記述は省略する。

▼構文

```
DRAGANDDROP|(file name)|(px)|(py)
```

[引数]

•(file name)

ドロップされたファイルのフルパス。

•(px)

ドロップされた位置の X 座標。

単位はスクリーン上のピクセル座標。

•(py)

ドロップされた位置の Y 座標。

単位はスクリーン上のピクセル座標。

KEY

▼説明

キー入力が発生したタイミングで出力される。

▼構文

```
KEY|(key name)
```

[引数]

•(key name)

入力されたキーの名前。

▼サンプルスクリプト

```
# 1 キー押下時、「1 キーが押された」を音声合成する。  
# a キー押下時、「a キーが押された」を音声合成する。  
0 0 KEY|1 SYNTH_START|mei|mei_voice_normal|1 キーが押された  
0 0 KEY|a SYNTH_START|mei|mei_voice_normal|a キーが押された
```

▼備考

1. 入力されたキーの名前を確認する方法

デバッグ画面（D キー押下）を表示することで、実際に入力されたキーを確認することができる。

EXECUTE

▼説明

外部プログラムを実行する。

テキストファイルや URL を記述した場合、OS 依存で自動的に対応したソフトウェアが起動する。

▼構文

```
EXECUTE|(file name)
```

[引数]

・(file name)

実行ファイルのパス。

カレントディレクトリを起点にした相対パスまたは絶対パスで指定する。

▼サンプルスクリプト

```
# 1 キー押下時、別プロセスの MMDAgent.exe を起動する。  
# 2 キー押下時、http://www.mmdagent.jp を開く。  
0 0 KEY|1 EXECUTE|MMDAgent.exe  
0 0 KEY|2 EXECUTE|http://www.mmdagent.jp
```

KEY_POST

▼説明

他のアプリケーションにキー入力を送信する。

動作は OS 依存となる。

▼構文

```
KEY_POST|(window class name)|(key name)|(ON or OFF for shift-key)|(ON or  
OFF for ctrl-key)|(On or OFF for alt-key)
```

[引数]

- (window class name)
キー入力を送信するアプリケーション名。
- (key name)
送信するキーの名前。
- (ON or OFF for shift-key)
[Shift]キーの入力を送信するか。
ON=送信する、OFF=送信しない。
- (ON or OFF for ctrl-key)
[Ctrl]キーの入力を送信するか。
ON=送信する、OFF=送信しない。
- (On or OFF for alt-key)
[Alt]キーの入力を送信するか。
ON=送信する、OFF=送信しない。

▼サンプルスクリプト

```
# 1 キー押下時、他のアプリケーションにキー入力を送信する。  
0 0 KEY|1 KEY_POST|SampleWindow|a|OFF|OFF|OFF
```

6. mdf ファイルについて

概要

MMDAgent のシステム共通設定が定義されているファイル。

MMDAgent.exe の引数に mdf ファイルを指定しない場合は、[システムディレクトリ](#)の mdf ファイルが読み込まれる。

mdf ファイルを指定した場合は、[システムディレクトリ](#)の mdf ファイルが読み込まれた後、[コンテンツディレクトリ](#)の mdf ファイルが読み込まれる。その際、同一の設定値が存在する場合には、[コンテンツディレクトリ](#)の mdf ファイルの設定値が反映される。

構文

各設定項目に対する設定値を 1 行単位で記述する。

また、[#] で開始した行はコメントとして扱われる。

```
設定項目名=設定値
```

▼記述例

セルシェーディング関連の設定を行う記述例を以下に記す。

```
# セルシェーディング設定
use_cartoon_rendering=true
use_mmd_like_cartoon=true
cartoon_edge_width=0.35
cartoon_edge_step=1.2
cartoon_edge_selected_color=1.0,0.0,0.0,1.0
```

設定項目一覧

use_cartoon_rendering

▼説明

セルシェーディングの ON/OFF を設定する。

セルシェーディング ON=true、セルシェーディング OFF=false。

▼記述例（既定値）

```
use_cartoon_rendering=true
```

use_mmd_like_cartoon

▼説明

セルシェーディング ON 時の MMD 互換性を設定する。

MMD 互換性 ON=true、MMD 互換性 OFF=false。

▼記述例（既定値）

```
use_mmd_like_cartoon=true
```

cartoon_edge_width

▼説明

セルシェーディング ON 時のエッジ幅を設定する。

[設定範囲]

0.001 ~ 1000.0

▼記述例（既定値）

```
cartoon_edge_width=0.7
```

cartoon_edge_step

▼説明

セルシェーディング ON 時のエッジ調整幅を設定する。

[設定範囲]

1.0 ~ 10.0

▼記述例 (既定値)

```
cartoon_edge_step=1.2
```

cartoon_edge_selected_color

▼説明

セルシェーディング ON 時の選択時エッジ色を設定する。

カンマ区切りで赤,緑,青,アルファの順番に記述する。

[設定範囲]

0.0 ~ 1.0

▼記述例 (既定値)

```
cartoon_edge_selected_color=1.0,0.0,0.0,1.0
```

camera_rotation

▼説明

カメラの 3D 回転を設定する。

カンマ区切りで X 軸回転,Y 軸回転,Z 軸回転の順番に記述する。

[設定範囲]

0.001 ~ 1000.0

▼記述例 (既定値)

```
camera_rotation=0.0,0.0,0.0
```

camera_transition

▼説明

カメラの注視点を設定する。
カンマ区切りで X 座標, Y 座標, Z 座標の順番に記述する。

[設定範囲]

-10000.0 ~ 10000.0

▼記述例 (既定値)

```
camera_transition=0.0,13.0,0.0
```

camera_distance

▼説明

カメラと注視点の距離を設定する。

[設定範囲]

0.0 ~ 100000.0

▼記述例 (既定値)

```
camera_distance=100.0
```

camera_fovy

▼説明

カメラの画角を設定する。

[設定範囲]

0.0 ~ 180.0

▼記述例 (既定値)

```
camera_fovy=16.0
```

stage_size

▼説明

ステージのサイズを設定する。

カンマ区切りで Width,Depth,Height の順番に記述する。

[設定範囲]

0.001 ~ 1000.0

▼記述例 (既定値)

```
stage_size=25.0,25.0,40.0
```

show_fps

▼説明

FPS 表示の ON/OFF を設定する。

FPS 表示=true、FPS 非表示=false。

▼記述例 (既定値)

```
show_fps=true
```

fps_position

▼説明

FPS 表示 ON 時の表示座標を設定する。

カンマ区切りで X 座標,Y 座標,Z 座標の順番に記述する。

▼記述例 (既定値)

```
fps_position=-2.5,22.0,3.0
```

window_size

▼説明

起動時のウィンドウサイズを設定する。
カンマ区切りで Width,Height の順番に記述する。

[設定範囲]

1 ~ 4096

▼記述例 (既定値)

```
window_size=600,600
```

full_screen

▼説明

起動時のフルスクリーン状態を設定する。
フルスクリーン ON =true、フルスクリーン OFF=false。

▼記述例 (既定値)

```
full_screen=false
```

log_size

▼説明

ログ画面のサイズを設定する。
カンマ区切りで Width,Height の順番に記述する。

[設定範囲]

1 ~ 4096

▼記述例 (既定値)

```
log_size=80,30
```

log_position

▼説明

ログ画面の表示座標を設定する。

カンマ区切りで X 座標, Y 座標, Z 座標の順番に記述する。

▼記述例 (既定値)

```
log_position=-17.5,3.0,-15.0
```

log_scale

▼説明

ログ画面の拡大率を設定する。

[設定範囲]

0.001 ~ 1000.0

▼記述例 (既定値)

```
log_scale=1.0
```

light_direction

▼説明

光源を「同次座標系」で指定する。

カンマ区切りで X, Y, Z, W の順番に記述する。

W が 0 のとき、光源は (X, Y, Z) の方向にある無限遠方の光源 (並行光源) となる。

W がそれ以外るとき、光源 (X/W, Y/W, W/WZ) の位置にある点光源となる。

▼記述例 (既定値)

```
light_direction=0.5,1.0,0.5,0.0
```

light_intensity

▼説明

光源の強さを設定する。

[設定範囲]

0.0 ~ 1.0

▼記述例（既定値）

```
light_intensity=0.6
```

light_color

▼説明

光源の色を設定する。

カンマ区切りで赤、緑、青の順番に記述する。

[設定範囲]

0.0 ~ 1.0

▼記述例（既定値）

```
light_color=1.0,1.0,1.0
```

campus_color

▼説明

背景色を設定する。

カンマ区切りで赤、緑、青の順番に記述する

[設定範囲]

0.0 ~ 1.0

▼記述例（既定値）

```
campus_color=0.0,0.0,0.2
```

max_multi_sampling

▼説明

OpenGL のマルチサンプリング数を設定する。

[設定範囲]

0 ~ 32

▼記述例 (既定値)

```
max_multi_sampling=4
```

motion_adjust_time

▼説明

モーション開始タイミングを遅延させる時間を設定する。

単位は秒。

[設定範囲]

-10.0 ~ 10.0

▼記述例 (既定値)

```
motion_adjust_time=0.0
```

lypsync_priority

▼説明

リップシンクのモーション優先度を設定する。

[設定範囲]

-1000.0 ~ 1000.0

▼記述例 (既定値)

```
lypsync_priority=100.0
```

bullet_fps

▼説明

物理演算の演算フレームを設定する。

単位は演算フレーム/秒。

[設定範囲]

1 ~ 120

▼記述例 (既定値)

```
bullet_fps=120
```

gravity_factor

▼説明

物理演算の重力スケール係数を設定する。

[設定範囲]

0.0 ~ 1024.0

▼記述例 (既定値)

```
gravity_factor=2.0
```

rotate_step

▼説明

回転操作の速度係数 (敏感さ) を指定する

[設定範囲]

0.001 ~ 180.0

▼記述例 (既定値)

```
rotate_step=4.5
```

translate_step

▼説明

並行移動操作の速度係数（敏感さ）を指定する

[設定範囲]

0.001 ~ 1000.0

▼記述例（既定値）

```
translate_step=0.5
```

distance_step

▼説明

カメラと注視点の距離調整操作の速度係数（敏感さ）を指定する。

[設定範囲]

0.001 ~ 1000.0

▼記述例（既定値）

```
distance_step=4.0
```

fovy_step

▼説明

カメラの画角調整操作の速度係数（敏感さ）を指定する。

[設定範囲]

0.001 ~ 1000.0

▼記述例（既定値）

```
fovy_step=1.0
```

use_shadow_mapping

▼説明

シャドウマッピングの ON/OFF を設定する。

シャドウマッピング ON=true、シャドウマッピング OFF=false。

▼記述例（既定値）

```
use_shadow_mapping=false
```

shadow_mapping_texture_size

▼説明

シャドウマッピングのテクスチャサイズを設定する。

[設定範囲]

1 ~ 8192

▼記述例（既定値）

```
shadow_mapping_texture_size=1024
```

shadow_mapping_self_density

▼説明

シャドウマッピング ON 時のモデル自身に落ちる影の濃さを設定する。

[設定範囲]

0.0 ~ 1.0

▼記述例（既定値）

```
shadow_mapping_self_density=1.0
```

shadow_mapping_floor_density

▼説明

シャドウマッピング ON 時の床に落ちる影の濃さを設定する。

[設定範囲]

0.0 ~ 1.0

▼記述例 (既定値)

```
shadow_mapping_floor_density=0.5
```

shadow_mapping_light_first

▼説明

シャドウマッピング ON 時の日向部分と影部分のレンダリング順を変更する。

日向を先にレンダリング=true、影を先にレンダリング=false。

[備考]

背景 PMD 使用時に背景が全て影になってしまう場合は true を指定することで改善されるが、背景が余分にレンダリングされるため、計算に要する時間が増加する。

▼記述例 (既定値)

```
shadow_mapping_light_first=true
```

display_comment_time

▼説明

起動時に PMD モデル内のコメントを表示する時間を設定する。
単位は秒。

[設定範囲]

0.0 ~ 30.0

▼記述例 (既定値)

```
display_comment_time=5.0
```

max_num_model

▼説明

最大表示モデル数を設定する。

[設定範囲]

1 ~ 1024

▼記述例 (既定値)

```
max_num_model=10
```

7. ojt ファイルについて

概要

音声合成で使用されるボイスファイルと発話スタイルが定義されているファイル。

構文

ボイスファイルの定義数、ボイスファイルのパス、発話スタイルの定義数、発話スタイル名と設定値を順番に記述する。また、[#] で開始された行はコメントとして扱われる。

▼記述例

3 個のボイスファイル (normal、angry、bashful) を元に 7 個の発話スタイル (normal、angry、bashful、fast、slow、high、low) を定義する場合の記述例を以下に記す。

```
# ボイスファイルの定義数
3
# ボイスファイルのパス
Voice¥normal.htsvoice
Voice¥angry.htsvoice
Voice¥bashful.htsvoice
# 発話スタイルの定義数
7
# 発話スタイル名と設定値
normal  1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0  0.0 0.55 1.0
angry   0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 1.1 -0.5 0.55 1.1
bashful 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 1.0  0.5 0.55 0.9
fast    1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 2.0  1.0 0.55 1.0
slow    1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 0.5  1.0 0.55 1.0
high    1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0  4.0 0.55 1.0
low     1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 -2.0 0.55 1.0
```

▼説明

1. ボイスファイルの定義数

使用するボイスファイルの数を記述する。

2. ボイスファイルのパス

使用するボイスファイルのパスをカレントディレクトリを起点にした相対パスまたは絶対パスで記述する。

3. 発話スタイルの定義数

定義する発話スタイルの数を記述する。

4. 発話スタイル名と設定値

各発話スタイルに対する設定値を以下の構文に従い 1 行単位で記述する。

また、構文中の区切り文字には [space] または [tab] を使用する。

発話スタイル名 声質の補間重み×n 音高の補間重み×n 話速の補間重み×n 全体の話し速 全体の音高 全体の声質 全体の音量

※ n = ボイスファイルの定義数。ボイスファイルのパスに記述した順番に適用される。

項目名	説明	値の範囲
発話スタイル名	発話スタイルの名前（別名）を記述する。	-
声質の補間重み × n	各ボイスの声質に対する比率を記述する。 値の合計値が 1.0 になるよう設定する。	0.0 ~ 1.0
音高の補間重み × n	各ボイスの音高に対する比率を記述する。 値の合計値が 1.0 になるよう設定する。	0.0 ~ 1.0
話速の補間重み × n	各ボイスの話速に対する比率を記述する。 値の合計値が 1.0 になるよう設定する。	0.0 ~ 1.0
全体の話し速	発話スタイル全体の話し速を記述する。	0.1 ~ 10.0 既定値 = 1.0
全体の音高	発話スタイル全体の音高を記述する	-12.0 ~ 12.0 既定値 = 0.0
全体の声質	発話スタイル全体の声質を記述する。	-0.9 ~ 0.9 既定値 = 0.55 女性声 < 0.55 < 男性声
全体の音量	発話スタイル全体の音量を記述する。	0.1 ~ 10.0 既定値 = 1.0

8. dic ファイルについて

概要

システム辞書に存在しない単語を認識単語として定義するためのユーザー辞書ファイル。

fst ファイル内で `RECOG_MODIFY|USERDICT_SET` を呼び出すことで、任意のユーザー辞書ファイルを設定することもできる。

▼システム辞書ファイル格納先

`<SysDir>/AppData/Julius/lang_m/web.60k.htkdic`

構文

追加したい単語の情報を以下の構文に従い 1 行単位で記述する。

また、構文中の区切り文字には [space] または [tab] を使用する。

単語の出現を表す文字列 @出現確率 単語の登録情報 [認識結果] 読みを表す音素列

▼記述例

メイをユーザー辞書に登録する場合の記述例を以下に記す。

花子:ハナコ:花子:513 @2.0 メイ:メイ:メイ:513 [メイ] m e i

▼説明

1. 単語の出現を表す文字列 (第 1 フィールド)
登録する単語と使われ方が似ている単語をシステム辞書から探し、システム辞書に記述された第 1 フィールドと同じ内容を記述する。不明な場合は<unk>と記述することが可能。
2. @出現確率 (第 2 フィールド)
登録する単語の出現確率 (対数確率) を記述する。
既定値は 0 となり、値が大きいくほど認識される確率が上昇するが、誤認識の可能性も高くなる。
3. 単語の登録情報 (第 3 フィールド)
登録する単語の情報を、表記:発音:基本形:品詞の番号 (第 1 フィールドで設定した数値) の順番に記述する。不明な場合は<unk>と記述することが可能。

4. [認識結果] (第 4 フィールド)
認識結果として出力される単語名を記述する。

5. 読みを表す音素列 (第 5 フィールド)
登録する単語の読みを[音声認識用の音素対応表](#)に従って記述する。

9. lip ファイルについて

概要

モデルファイルの口形モーションを定義したファイル。

モデルファイルが格納されているディレクトリに<モデルファイル名.lip>の形で格納されている。

なお、対象の lip ファイルが存在しない場合、自動的にシステム lip ファイルが読み込まれる。

▼システム lip ファイル格納先

<SysDir>/AppData/lip.txt

構文

口形モーションの定義数、口形モーションの定義名、音素数、音素名と口形モーションの補間重みを順番に記述する。また、[#] で開始された行はコメントとして扱われる。

▼記述例

3 個の口形モーション（あ、い、う）を元に 4 個の音素（a、i、u、g）を定義する場合の記述例を以下に記す。

```
# 口形モーションの定義数
3
# 口形モーションの定義名
あ
い
う
# 音素数
4
# 音素名と口形モーションの補間重み
a 1.0 0.0 0.0
i 0.0 1.0 0.0
u 0.0 0.0 1.0
g 0.1 0.3 0.2
```

▼説明

1. 口形モーションの定義数
定義する口形モーションの数を記述する。
2. 口形モーションの定義名
モデルファイルに登録されている口形モーションの定義名を記述する。
3. 音素数
定義する音素数を記述する。
4. 音素名と口形モーションの補間重み
各音素に対する設定値を以下の構文に従い 1 行単位で記述する。
また、構文中の区切り文字には [space] または [tab] を使用する。

音素名 口形モーションの補間重み×n

※ n = 口形モーションの定義数。口形モーションの定義名に記述した順番に適用される。

項目名	説明	値の範囲
音素名	音素名を音声合成用の音素対応表に従って記述する。	-
口形モーションの補間重み × n	各口形モーションに対する比率を記述する。	0.0 ~ 1.0

[備考]

口形モーションの補間重み（比率）の合計が 1.0 にならない場合、各数値を%とみなし、口形モーションが生成される。

10. lab ファイルについて

概要

WAV ファイル再生中にモデルをリップシンクさせるための情報が定義されたファイル。

SOUND_START 実行時、自動的に WAV ファイルと同じディレクトリに格納された <WAV ファイル名.lab>を読み込み、モデルのリップシンクを行う。

構文

各音素に対するリップシンクの開始時間、終了時間、音素名を以下の構文に従い 1 行単位で記述する。また、構文中の区切り文字には [space] または [tab] を使用する。

開始時間 終了時間 音素名

▼記述例

「こんにちは」をリップシンクする場合の記述例を以下に記す。

0	1050000	k
1050000	2200000	o
2200000	2650000	N
2650000	3150000	n
3150000	3950000	i
3950000	4750000	ch
4750000	5300000	i
5300000	5850000	w
5850000	7700000	a

▼説明

1. 開始時間（第 1 フィールド）

リップシンクを開始する時間を記述する。

単位は 100ns。

2. 終了時間（第 2 フィールド）

リップシンクを終了する時間を記述する。

単位は 100ns。

3. 音素名（第 3 フィールド）

使用する音素名を[音声合成用の音素対応表](#)に従って記述する。

※注意※

バージョン 1.5 において、lab ファイル中の音素数が 339 個を超える場合、実行時に MMDAgent が強制終了する。

11. 音素対応表

概要

モーラと音素の対応を記述した表。音声認識用と音声合成用の2種類が存在する。

▼モーラと音素について

モーラ	<p>言語の音の分節単位。日本語では基本的に仮名1文字が1モーラにあたるが、「ちゃ」などの拗音も1モーラである。</p> <p>[例] 「めいちゃん」をモーラで表すと「me i cha N」となり、4モーラから構成される。</p>
音素	<p>モーラよりも小さい音の分節単位。1モーラは単一または複数の音素から構成される。</p> <p>[例] 「めいちゃん」を音素で表すと「m e i ch a N」となり、6音素から構成される。</p>

音声認識用の音素対応表

音声認識用の音素とモーラの対応表を以下に記す。

正音					拗音				
ア	イ	ウ	エ	オ	キャ		キュ		キョ
a	i	u	e	o	ky a		ky u		ky o
カ	キ	ク	ケ	コ	シャ		シュ	シェ	ショ
ka	ki	ku	ke	ko	sh a		sh u	sh e	sh o
サ	シ	ス	セ	ソ	チャ		チュ	チェ	チョ
sa	sh i	su	se	so	ch a		ch u	ch e	ch o
タ	チ	ツ	テ	ト	ニャ		ニュ		ニョ
ta	ch i	tsu	te	to	ny a		ny u		ny o
ナ	ニ	ヌ	ネ	ノ	ヒャ		ヒュ		ヒョ
na	ni	nu	ne	no	hy a		hy u		hy o
ハ	ヒ	フ	ヘ	ホ	ミャ		ミュ		ミョ
ha	hi	fu	he	ho	my a		my u		my o
マ	ミ	ム	メ	モ	リャ		リュ		リョ
ma	mi	mu	me	mo	ry a		ry u		ry o
ヤ		ユ		ヨ	ギャ		ギユ		ギョ
ya		yu		yo	gy a		gy u		gy o
ラ	リ	ル	レ	ロ	ジャ		ジュ	ジェ	ジョ
ra	ri	ru	re	ro	ja		ju	je	jo
ワ		ヲ		ン	ビャ		ビュ		ビョ
wa		o		N	by a		by u		by o
濁音・半濁音					ピャ		ピュ		ピョ
ガ	ギ	グ	ゲ	ゴ	py a		py u		py o
ga	gi	gu	ge	go	ファ	フィ	フユ	フェ	フォ
ザ	ジ	ズ	ゼ	ゾ	fa	fi	hy u	fe	fo
za	zi	zu	ze	zo		ウィ		ウエ	ウオ
ダ	ヂ	ヅ	デ	ド		wi		we	wo
da	zi	zu	de	do	ヴァ	ヴィ		ヴェ	ヴォ
バ	ビ	ブ	ベ	ボ	ba	bi		be	bo
ba	bi	bu	be	bo	ツァ	ツイ		ツエ	ツォ
パ	ピ	プ	ペ	ポ	ts a	ts i		ts e	ts o
pa	pi	pu	pe	po		ティ	テユ		
特殊な音素						ti	tu		
促音(っ)	q	(例)ラッコ		ra q ko		ディ	デュ		
長音(ー)	:	(例)チーター		ch i: t a:		di	du		
							ドウ		
							du		

音声合成用の音素対応表

音声合成用の音素とモーラの対応表を以下に記す。

正音					拗音				
ア	イ	ウ	エ	オ	キャ		キュ	キエ	キョ
a	i	u	e	o	ky a		ky u	ky e	ky o
カ	キ	ク	ケ	コ	シャ		シュ	シェ	ショ
ka	ki	ku	ke	ko	sh a		sh u	sh e	sh o
サ	シ	ス	セ	ソ	チャ		チュ	チェ	チョ
sa	sh i	su	se	so	ch a		ch u	ch e	ch o
タ	チ	ツ	テ	ト	ニャ		ニュ	ニエ	ニョ
ta	ch i	tsu	te	to	ny a		ny u	ny e	ny o
ナ	ニ	ヌ	ネ	ノ	ヒャ		ヒュ	ヒエ	ヒョ
na	ni	nu	ne	no	hy a		hy u	hy e	hy o
ハ	ヒ	フ	ヘ	ホ	ミャ		ミュ	ミエ	ミョ
ha	hi	fu	he	ho	my a		my u	my e	my o
マ	ミ	ム	メ	モ	リャ		リュ	リエ	リョ
ma	mi	mu	me	mo	ry a		ry u	ry e	ry o
ヤ		ユ		ヨ	ギャ		ギユ	ギエ	ギョ
ya		yu		yo	gy a		gy u	gy e	gy o
ラ	リ	ル	レ	ロ	ジャ		ジュ	ジェ	ジョ
ra	ri	ru	re	ro	ja		ju	je	jo
ワ		ヲ		ン	ビャ		ビュ	ビエ	ビョ
wa		o		N	by a		by u	by e	by o
濁音・半濁音					ピャ		ピュ	ピエ	ピョ
ガ	ギ	グ	ゲ	ゴ	py a		py u	py e	py o
ga	gi	gu	ge	go	ファ	フィ		フェ	フォ
ザ	ジ	ズ	ゼ	ゾ	fa	fi		fe	fo
za	ji	zu	ze	zo		ウィ		ウエ	ウオ
ダ	ヂ	ヅ	デ	ド		wi		we	wo
da	ji	zu	de	do	ヴァ	ヴィ	ヴ	ヴェ	ヴォ
ba	bi	bu	be	bo	va	vi	vu	ve	vo
パ	ピ	プ	ペ	ポ	ヴァ		ヴュ		ヴォ
pa	pi	pu	pe	po	by a		by u		by o
促音					ツァ	ツイ		ツエ	ツォ
ッ					ts a	ts i		ts e	ts o
cl					テャ	テイ	テュ		テョ
					ty a	ti	ty u		ty o
					デャ	デイ	デュ	デエ	デョ
					by a	di	by u	by e	by o
							トゥ		
							tu		
							ドゥ		
							du		
						スイ			
						si			
						ズイ			
						zi			
					クワ				
					kw a				
					グワ				
					gw a				
								イエ	
								ye	

