

# Galatea Agent Manager コマンド仕様

川本真一 (北陸先端科学技術大学院大学 情報科学研究科)

2003年8月8日

## 1 エージェント管理部とコマンド仕様

### 1.1 エージェント管理部と各モジュールの関係

エージェント管理部 (AM) は大きく分けて 2 つの機能レイヤーで構成される。Direct Control Layer (AM-DCL) は、各モジュールの規定するコマンドセットを直接制御する事を可能とするレイヤーであり、多くのモジュールはこのレイヤーを介して他のモジュールとの通信を行う。Macro Control Layer (AM-MCL) は、主にタスク管理部 (TM) 向けのレイヤーで、良く使われる一連のコマンドセットをまとめたマクロコマンドとして再定義したり、モジュール間の同期管理などの低レベルなモジュール制御を請け負うことで、タスク管理部から見た利便性向上を目的としている。

エージェント管理部と各モジュールとの基本構成を図 1 に示す。

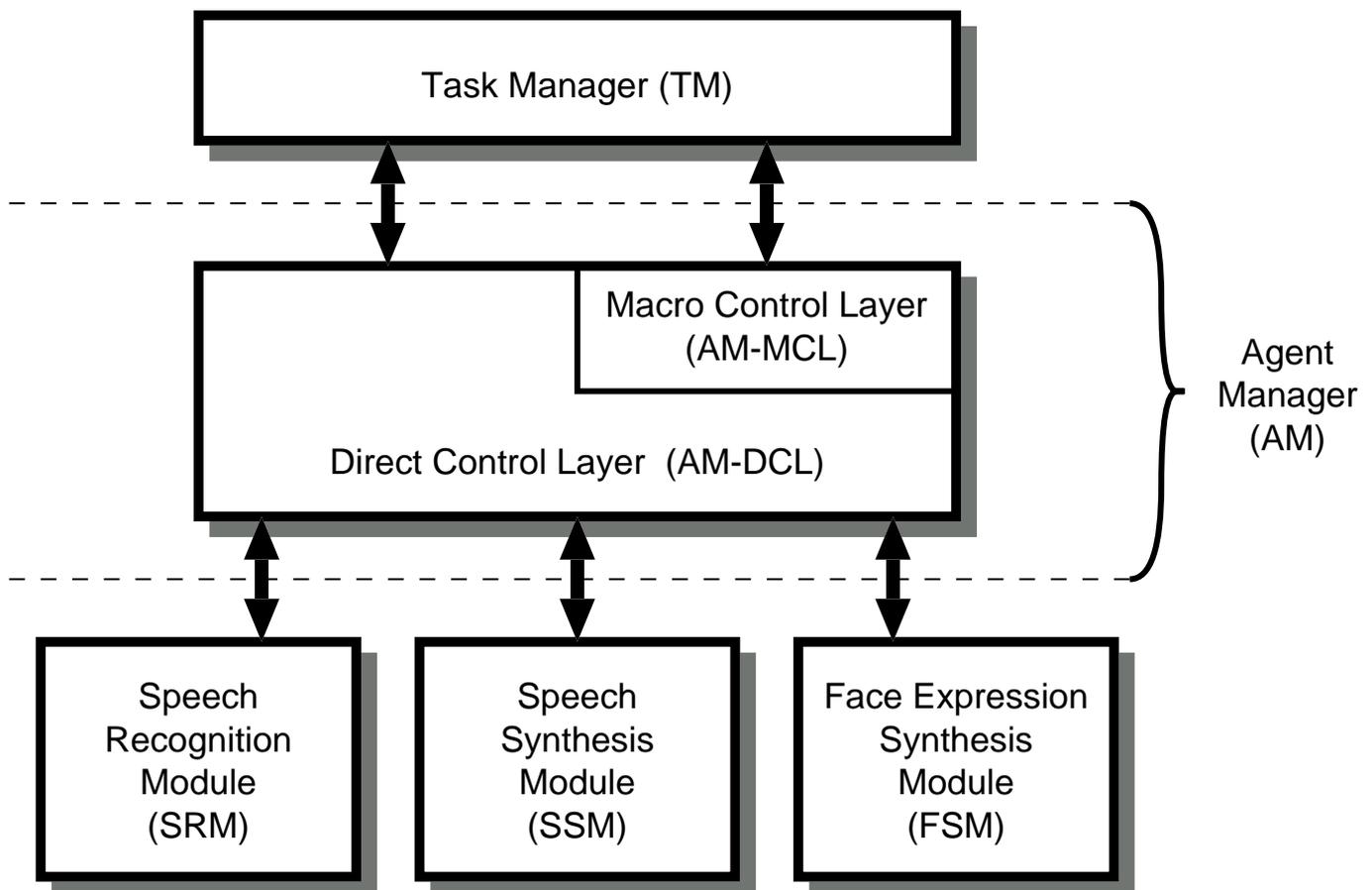


図 1. エージェント管理部と各モジュールとの基本構成図

音声認識モジュール (SRM)、音声合成モジュール (SSM)、および顔画像生成モジュール (FSM) は、原則として AM-DCL との間で通信を行う。タスク管理部は基本的に AM-MCL との間で通信を行うが、必要に応じて他のモジュールと同様に AM-DCL との通信も行うことができる。各モジュールからの全ての出力はタスク制御部に提供される。

## 1.2 仮想マシンモデル

コマンド仕様策定にあたり、各モジュールは仮想マシンモデルとして扱う。図 2 にエージェント管理部 (AM-DCL) と仮想マシンモデルとの関係図を示す。

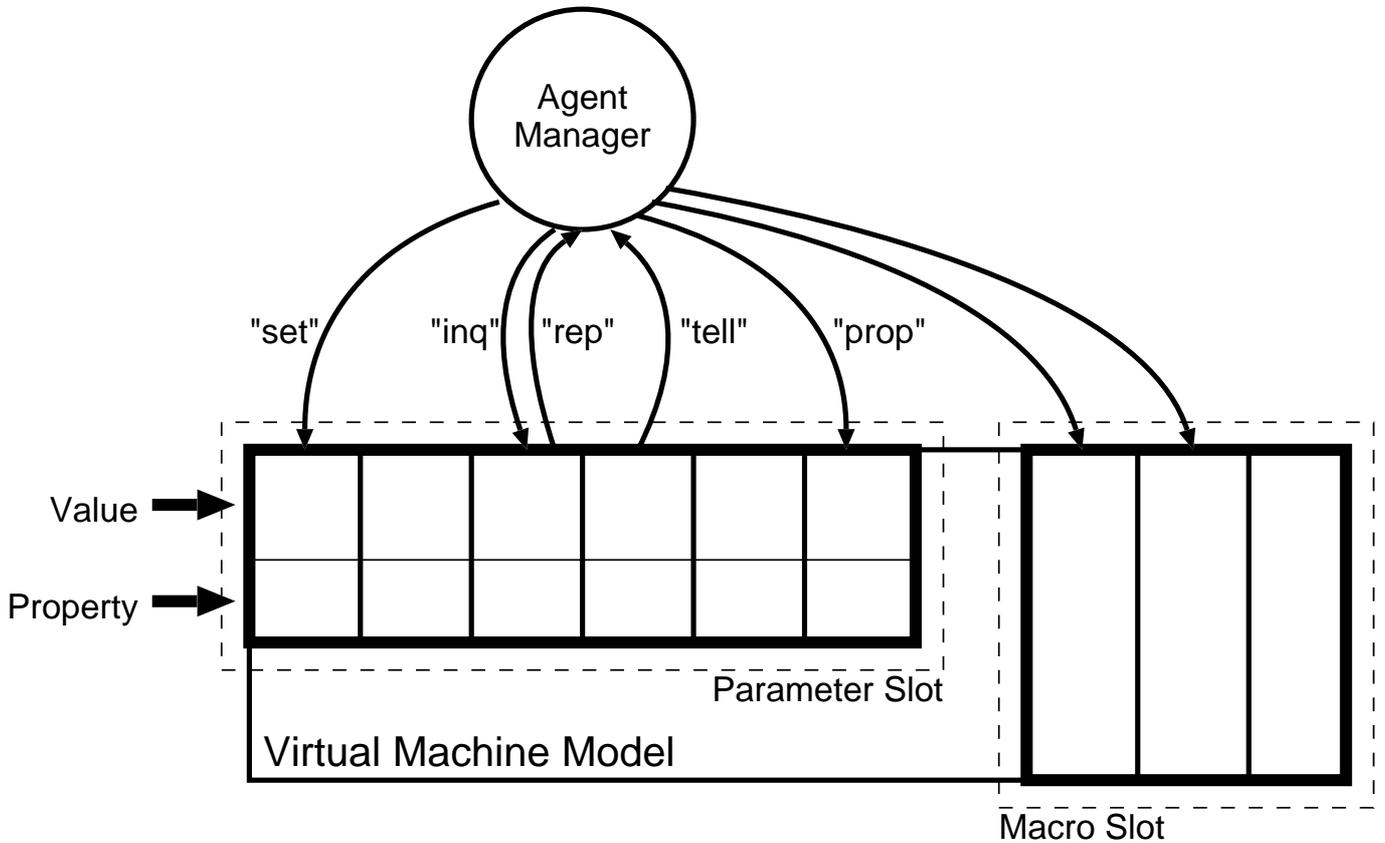


図 2. エージェント管理部と仮想マシンモデルとの関係図

各モジュールは、各入出力パラメータをパラメータスロットとして管理する。また、各モジュールにおけるマクロコマンド定義も、同様にマクロスロットとして管理する。これらのスロットは、エージェント管理部との間で共有され、これらのスロットを介して、エージェント管理部とモジュールとの間の通信を行う。

各スロットはそれぞれ値と属性を持ち、仮想マシンモデルのスイッチのように扱われる。また、各スロットは共通のコマンドにより操作される。スロットの値により、動作状態の監視や、動作の開始・終了の指示、動作環境の設定などを操作できる。スロットの値の変更は、基本的には即座に動作に反映される。つまり、例えば動作指示のスロット値を変更することは、即座に動作につながることを意味する。

これにより、各モジュールの扱いを統一化することが出来る。コマンドはモジュールに依存しない通信仕様であり、パラメータスロットは通信には依存しないモジュール依存の仕様である。つまり、モジュール間の違いは、準備されるパラメータスロットの機能の違いのみとなる。また、各モジュールを仮想マシンモデルとして抽象化して扱うことで、機能拡張やモジュールの追加が容易になる。例えば、新しい機能を追加することは、仮想マシンモデルのパラメータスロットを追加することで実現できる。モジュールの追加も他のモジュールと同様に仮想マシンモデルの考えに基づき、パラメータスロットを定義することで実現できる。

より具体的には、例えば新しく顔画像認識モジュールが追加されたならば、共通仕様であるモジュールの起動などのパラメータスロット、および顔画像認識結果の出力に関するパラメータスロット、顔画像入力的手段等を指定するパラメータスロット、認識アルゴリズムを指定するパラメータスロット、およびモデル等を指定するパラメータスロットを定義することで、顔画像認識モジュールをシステムに取り込むための最低限の枠組を考えることが出来る。また例えば、音声認識モジュールの音声分析を拡張して、基本周波数  $F_0$  を取得することが実現された場合、仮想マシンモデルは、基本周波数出力を設定するためのパラメータスロットと、基本周波数抽出アルゴリズム指定のためのパラメータスロットを追加するだけで、機能拡張にも対応できると考える。

### 1.3 汎用仮想モジュールの基本操作用コマンド

汎用仮想モジュールの基本操作に用いるコマンドは、次のようなものである。

コマンド名	機能
set	パラメータスロットの値の設定
inq	スロットの値の問い合わせ
prop	スロットの属性の設定

また、スロット値などモジュールからの出力の際に付加される識別子としては、次のような 2 種類が存在する。

コマンド名	機能
rep	スロットの値の出力
tell	スロットとして規定されない値の出力

set

書式 1 set スロット名 = 値

機能 1 指定するパラメータスロット名の値を直接設定する。

使用例 1 set *Speak* = *NOW*

使用例 2 set *FaceExp* = *HAPPY 2 100 500*

書式 2 set スロット名 < ファイル名

機能 2 指定するパラメータスロット名の値をファイルから読み込む。

書式 3 set スロット名 << 終端識別子

値 1

値 2

⋮

値 N

終端識別子

機能 3 指定するパラメータスロット名の複数行にわたる値を直接指定する。但し、指定される値に終端指定子は存在してはいけない。

使用例 set *Text* << *END*

<VOLUME LEVEL=" -30" >

<PITCH LEVEL=" +3" >

<RATE SPEED=" +3" >

こんにちは

</RATE>

</PITCH>

</VOLUME>

*END*

inq

書式 inq スロット名

機能 指定するスロット名 (パラメータ名、もしくはマクロ名) の値を問い合わせる。その際、モジュールからの応答には、識別子 rep が付加される。

使用例 inq *Speak.stat*

## **prop**

書式 `prop` スロット名 = 値

機能 指定するスロット名 (パラメータ名、もしくはマクロ名) の属性を設定する。

使用例 `prop Speak.stat = AutoOutput`

## **rep**

書式 `rep` スロット名 値

機能 指定するスロット名 (パラメータ名、もしくはマクロ名) の値を出力する際の識別子。

使用例 「`inq Param.beam1`」を実行したときの出力例。

`rep Param.beam1 = 100`

## **tell**

書式 `tell` メッセージ

機能 スロットとして規定されない値を出力する際の識別子。

使用例 エラー出力の例。

`tell` “現在は発話していません”

## 2 パラメータ一覧

### 2.1 モジュール共通パラメータ

モジュールの起動、終了、問い合わせ	
パラメータ名	機能
Run	モジュールの起動、再起動、停止 モジュールの起動状態の問い合わせ
ProtocolVersion	モジュールの準拠している標準通信プロトコルの version 情報の問い合わせ
ModuleVersion	モジュール自体の version 情報の問い合わせ

### 2.2 音声認識部

認識制御関連	
パラメータ名	機能
Run	認識処理の制御
モデル指定関連	
パラメータ名	機能
AcousticModel	認識用音響モデルの指定
Grammar	認識用文法の指定
Dic	認識用辞書の指定
探索パラメータ指定関連	
パラメータ名	機能
Param	探索パラメータの指定
出力指定関連	
パラメータ名	機能
Output	音声認識モジュールからの出力形式の指定
入力指定関連	
パラメータ名	機能
Input	音声認識モジュールへの入力に関する設定

### 2.3 音声合成部

モジュール設定関連	
パラメータ名	機能
SpeakerSet	発話可能な話者の問い合わせ
Speaker	音声合成に使用する話者の選択
合成処理関連	
パラメータ名	機能
Text	音声合成の準備
Speak	決められた時刻に発声

## 2.4 顔画像生成部

基本パラメータの直接制御	
パラメータ名	機能
Face	表情パラメータ制御コマンドの直接制御
Mouth	口形状パラメータ制御コマンドの直接制御
MouthScale	口形状の全体の変形量 [%] の設定
HeadRotRel	現在の位置を基準とした頭部の回転
HeadRotAbs	正面を基準とした頭部の回転
基本動作、基本表情	
パラメータ名	機能
FaceMot	動作パターン ID に指定された基本動作の制御
FaceExp	強度と継続長の指定された基本表情の制御 または、表情パラメータのリセット
リップシンク	
パラメータ名	機能
LipSync	音声合成との同期準備
Speak	決められた時刻に発声
顔画像の選択	
パラメータ名	機能
MaskSet	使用可能な顔画像の問い合わせ
Mask	顔画像生成に使用する顔画像の選択
自律動作	
パラメータ名	機能
Autonomous	自律動作の設定

## 2.5 エージェント管理部

エージェント管理部の Macro Control Layer (AM-MCL) を介して、各モジュールを制御するためのコマンドパラメータの一覧を以下に示す。詳細に付いては、12.1 節に記載する。

モジュールの起動、終了、問い合わせ	
パラメータ名	機能
Run	モジュールの起動、再起動、停止 モジュールの起動状態の問い合わせ
ProtocolVersion	モジュールの準拠している標準通信プロトコルの version 情報の問い合わせ
ModuleVersion	モジュール自体の version 情報の問い合わせ
同期管理マクロ	
パラメータ名	機能
Speak	リップシンクを伴う音声出力の開始、停止

## 3 モジュール共通パラメータ

### 3.1 モジュールの起動、終了、問い合わせ

#### Run

動作 set モジュールの動作を設定する。

1. set Run = INIT
  - モジュールを起動、再起動する。
2. set Run = EXIT
  - モジュールを停止する。

inq モジュールの動作状況を問い合わせに対して rep により答える。

1. inq Run
  - rep Run = LIVE
    - \* 各モジュールが処理開始可能状態である時の出力
  - rep Run = DEAD
    - \* 各モジュールが処理開始不可能状態である時の出力

#### ProtocolVersion

動作 inq モジュールの準拠している標準通信プロトコルの version 情報の問い合わせに対して rep により答える。

1. inq ProtocolVersion
  - rep ProtocolVersion = “Protocol Ver. 0.02”
    - \* モジュールのプロトコル version 情報

#### ModuleVersion

動作 inq モジュール自体の version 情報の問い合わせに対して rep により答える。

1. inq ModuleVersion
  - rep ModuleVersion = “Module Ver. 0.02”
    - \* モジュールの version 情報

## 4 音声認識部パラメータ

### 4.1 認識制御関連

#### Run

動作 **set** 認識処理の制御を行う。

1. **set Run = START**
  - 認識処理を開始する。
2. **set Run = PAUSE**
  - 認識処理を一時中断する。認識中の場合は、そこまでの認識結果を返す
3. **set Run = TERMINATE**
  - 認識処理を一時中断する。認識中でも結果を破棄し、すぐに中断する。
4. **set Run = RESUME**
  - 認識処理を再開する。
5. **set Run = STOP**
  - 認識処理を終了する。

**inq** 認識処理の状態に関する問い合わせに対して **rep** により答える。

属性 **Run.process** : 認識処理の状態。

- **inq Run.process** : 認識処理の状態を **rep** により返す。
  - \* **rep Run.process = ACTIVE**
    - ★ 音声切り出し (認識) 中の状態。
  - \* **rep Run.process = SLEEP**
    - ★ 音声切り出し (認識) 中ではない状態。

### 4.2 モデル指定関連

#### AcousticModel

動作 **set** 認識用音響モデル、および音響モデル補間リストを指定する。

1. **set AcousticModel. 属性名 = ファイル名**
  - 指定したファイルに記述された音響モデル、もしくは音響モデル補間リストを、認識に使用するモデルとして設定する。

**inq** 音響モデル、および音響モデル補間リストに関する問い合わせに対して **rep** により答える。

属性 **AcousticModel.hmm** : 音響モデル

- **set AcousticModel.hmm = male01.hmm** : 音響モデルとしてファイル “male01.hmm” を設定する。

**AcousticModel.imp** : 音響モデル補間ファイル

- **set AcousticModel.imp = male01.hmm1ist**: 音響モデル補間ファイルとして “male01.hmm1ist” を設定する。

**AcousticModel.hmmfile** : 音響モデルのファイル名

- **inq AcousticModel.hmmfile** : 音響モデルのファイル名を **rep** により返す。

**AcousticModel.impfile** : 音響モデル補間ファイルのファイル名

- **inq AcousticModel.impfile** : 音響モデル補間ファイルのファイル名を **rep** により返す。

## Grammar

動作 **set** 認識文法を指定する。

1. **set Grammar = ファイル名**
  - 指定したファイルに記述された認識文法を、認識に使用する言語モデルとして設定する。
2. **set Grammar << END**  
認識文法  
END
  - 指定された認識文法を、認識に使用する言語モデルとして設定する。

**inq** 認識文法に関する問い合わせに対して **rep** により答える。

属性 **Grammar.file** : 認識文法のファイル名

- **inq Grammar.file** : 指定した認識文法のファイル名を **rep** により返す。

**Grammar.type** : 認識文法のタイプ

- **inq Grammar.type** : 指定した認識文法の形式を **rep** により返す。

## Dic

動作 **set** 認識辞書を指定する。

1. **set Dic = ファイル名**
  - 指定したファイルに記述された認識辞書を、認識に使用する辞書として設定する。

**inq** 認識辞書に関する問い合わせに対して **rep** により答える。

属性 **Dic.file** : 認識辞書のファイル名

- **inq Dic.file** : 指定した認識辞書のファイル名を **rep** により返す。

**Dic.type** : 認識文法のタイプ

- **inq Dic.type** : 指定した認識辞書の形式を **rep** により返す。

## 4.3 探索パラメータ指定関連

### Param

動作 **set** 探索パラメータを設定する。

1. **set Param. 属性名 = パラメータ**
  - 属性値のパラメータを指定する。

**inq** 探索パラメータに関する問い合わせに対して **rep** により答える。

属性 **Param.beam1** : 第1パスのビーム幅。値は第1パスビーム幅(ノード数)。

- **set Param.beam1 = 600** : 第1パスのビーム幅を600に設定する。
- **inq Param.beam1** : 現在の設定されている第1パスのビーム幅を **rep** により返す。

**Param.beam2** : 第2パスのビーム幅。値は第2パスビーム幅(ノード数)。

- **set Param.beam2 = 30** : 第2パスのビーム幅を30に設定する。
- **inq Param.beam2** : 現在の設定されている第2パスのビーム幅を **rep** により返す。

**Param.stack** : スタックサイズを指定する。値はスタックサイズ。

- **set Param.stack = 500** : スタックサイズを500に設定する。
- **inq Param.stack** : 現在の設定されているスタックサイズを **rep** により返す。

**Param.overflow** : 仮説のオーバーフローを判定する仮説数の上限を指定する。値は仮説オーバーフロー数。

- set Param.overflow = 2000 : スタックサイズを 2000 に設定する。
- inq Param.overflow : 現在の設定されているスタックサイズを rep により返す。

**Param.lookuprange** : lookuprange 数を指定する。値は lookuprange 数。

- set Param.lookuprange = 5 : lookuprange 数を 5 に設定する。
- inq Param.lookuprange : 現在の設定されている lookuprange 数を rep により返す。

**Param.gauss** : PTM 分布の計算量を指定する。値は PTM 分布計算量 (full:no prune)。

- set Param.gauss = 2 : PTM 分布の計算量を 2 に設定する。
- inq Param.gauss : 現在の設定されている PTM 分布の計算量を rep により返す。

## 4.4 出力指定関連

### Output

**動作 set** 音声認識モジュールからの出力形式を設定する。

1. set Output. 属性名 = パラメータ
  - 属性値のパラメータを指定する。

**inq** 出力形式に関する問い合わせに対して rep により答える。

**属性 Output.nbest** : N-best 出力数を指定する。値は nbest 出力数。

- set Output.nbest = 5 : nbest 出力数を 5 に設定する。
- inq Output.nbest : 現在の設定されている N-best 出力数を rep により返す。

**Output.pass1prog** : 第 1 パス逐次出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.pass1prog = ON : 第 1 パス逐次出力を行う。
- inq Output.pass1prog : 現在の第 1 パス逐次出力指定を rep により返す。

**Output.pass1proptime** : 第 1 パス逐次出力タイミングを指定する。値はフレーム数。

- set Output.pass1proptime = 10 : 第 1 パス逐次出力タイミングを 10 フレームに指定する。
- inq Output.pass1proptime : 現在の第 1 パス逐次出力タイミング設定を rep により返す。

**Output.walign** : 単語アラインメント情報の出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.walign = ON : 単語アラインメント情報を出力する。
- inq Output.walign : 現在の単語アラインメント情報出力設定を rep により返す。

**Output.palign** : 音素アラインメント情報の出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.palign = ON : 音素アラインメント情報を出力する。
- inq Output.palign : 現在の音素アラインメント情報出力設定を rep により返す。

**Output.word** : 単語列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.word = ON : 単語列出力を行う。
- inq Output.word : 現在の単語列出力設定を rep により返す。

**Output.LMword** : 文法カテゴリ番号列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.LMword = ON : 文法カテゴリ番号列出力を行う。
- inq Output.LMword : 現在の文法カテゴリ番号列出力設定を rep により返す。

**Output.phone** : 音素列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.phone = ON : 音素列出力を行う。

- inq Output.phone : 現在の音素列出力設定を rep により返す。

**Output.score** : スコア出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.score = ON : スコア出力を行う。
- inq Output.score : 現在のスコア出力設定を rep により返す。

**Output.pass1word** : 第 1 パスの単語列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.pass1word = ON : 第 1 パスの単語列出力を行う。
- inq Output.pass1word : 現在の第 1 パスの単語列出力設定を rep により返す。

**Output.pass1LMword** : 第 1 パスの文法カテゴリ番号列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.pass1LMword = ON : 第 1 パスの文法カテゴリ番号列出力を行う。
- inq Output.pass1LMword : 現在の第 1 パスの文法カテゴリ番号列出力設定を rep により返す。

**Output.pass1phone** : 第 1 パスの音素列出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.pass1phone = ON : 第 1 パスの音素列出力を行う。
- inq Output.pass1phone : 現在の第 1 パスの音素列出力設定を rep により返す。

**Output.pass1score** : 第 1 パスのスコア出力を指定する。値は ON、もしくは OFF(ON:出力する, OFF:出力しない)。

- set Output.pass1score = ON : 第 1 パスのスコア出力を行う。
- inq Output.pass1score : 現在の第 1 パスのスコア出力設定を rep により返す。

## 4.5 入力指定関連

### Input

動作 set 音声認識モジュールへの入力に関する設定を行う。

1. set Input. 属性値 = パラメータ
  - 指定した属性値のパラメータを設定する。

属性 Input.from : 入力手段

- set Input.from = mic : 音声入力としてマイクロフォン入力を設定。

Input.level : 音声切り出しレベルの閾値

- set Input.level = 3000 : 音声切り出しレベルの閾値を 3000 に設定。

## 5 音声合成部パラメータ

### 5.1 モジュール設定関連

#### SpeakerSet

動作 inq 発話可能な話者の問い合わせに対して rep により答える。

1. inq SpeakerSet
  - rep SpeakerSet = male01 male02 female01
  - \* 発話可能な話者の ID コード列

#### Speaker

動作 set 音声合成に使用する話者を設定する。

1. set Speaker = male01
  - 音声合成に使用する male01(話者 ID) の指定する。話者 ID は SpeakerSet にて問い合わせた話者 ID を指定する。

### 5.2 合成処理関連

#### Text

動作 set 以下のようにして指定された文章から音声合成を行うための準備を行う。“JEIDA-62-2000 日本語テキスト音声合成用記号の規格”に準拠した形式で発話内容を指定する。

1. set Text = 発話内容
  - 指定した発話内容の音声合成準備を行う。

inq 音声出力に関する問い合わせに対して rep により答える。

属性 Text.text : 発話内容

- inq Text.text : 現在の設定されている発話文章を rep により返す。

Text.pho : 音素系列と音素継続長

- inq Text.pho : 現在の設定されている発話文章の音素系列と音素継続時間を rep により返す。

Text.dur : 総発話時間

- inq Text.dur : 現在の設定されている発話文章の総発話時間を rep により返す。

#### Speak

動作 set 既に音声出力準備された音声を決められた時間に出力する。もしくは、音声出力を停止する。

1. set Speak = 時刻
  - 指定した時刻に準備された音声出力を行う。時刻の指定形式は “時:分:秒.ミリ秒” (例: “11:39:30.200”)
2. set Speak = + 時刻
  - 現在時刻を基準に指定した相対時刻後に準備された音声出力を行う。時刻の指定形式は “時:分:秒.ミリ秒” (例: “11:39:30.200”)
3. set Speak = + ミリ秒
  - 現在時刻を基準指定した相対時刻後に準備された音声出力を行う。
4. set Speak = NOW

- すぐに既に準備された音声出力を行う。

5. set Speak = STOP

- 現在の音声出力の中止を指示する。

inq 音声出力に関する問い合わせに対して rep により答える。

prop 音声出力に関する属性値の出力仕様を設定する。

属性 **Speak.text** : 発話内容

- inq Speak.text  
... 現在の設定されている発話文章を rep により返す。
- prop Speak.text = AutoOutput  
... 現在の設定されている発話文章を変更された時点で自動的に rep により返すように設定する。
- prop Speak.text = NoAutoOutput  
... 現在の設定されている発話文章を inq にて問い合わせた時に rep により返すように設定する。

**Speak.pho** : 音素系列と音素継続長

- inq Speak.pho  
... 現在の設定されている発話文章の音素系列と音素継続時間を rep により返す。
- prop Speak.pho = AutoOutput  
... 現在の設定されている発話文章の音素系列と音素継続時間を変更された時点で rep により返すように設定する。
- prop Speak.pho = NoAutoOutput  
... 現在の設定されている発話文章の音素系列と音素継続時間を inq にて問い合わせた時に rep により返すように設定する。

**Speak.dur** : 総発話時間

- inq Speak.dur  
... 現在の設定されている発話文章の総発話時間を rep により返す。
- prop Speak.dur = AutoOutput  
... 現在の設定されている発話文章の総発話時間を変更された時点で自動的に rep により返すように設定する。
- prop Speak.dur = NoAutoOutput  
... 現在の設定されている発話文章の総発話時間を inq にて問い合わせた時に rep により返すように設定する。

**Speak.utt** : 既に発話が完了した音素系列

- inq Speak.utt  
... 既に発話が完了した音素系列を rep により返す。
- prop Speak.utt = AutoOutput  
... 既に発話が完了した音素系列を変更された時点で自動的に rep により返すように設定する。
- prop Speak.utt = NoAutoOutput  
... 既に発話が完了した音素系列を inq にて問い合わせた時に rep により返すように設定する。

**Speak.len** : 既に発話が完了した発話時間

- inq Speak.len  
... 既に発話が完了した発話時間を rep により返す。
- prop Speak.len = AutoOutput  
... 既に発話が完了した発話時間を変更された時点で自動的に rep により返すように設定する。
- prop Speak.len = NoAutoOutput  
... 既に発話が完了した発話時間を inq にて問い合わせた時に rep により返すように設定する。

**Speak.stat** : 現在の音声合成処理の状態

- inq Speak.stat  
... 現在の音声合成処理の状態を rep により返す。

- \* rep Speak.stat = IDLE  
... 何も処理していない状態
- \* rep Speak.stat = PROCESSING  
... 発話準備中の状態
- \* rep Speak.stat = READY  
... 発話準備が完了した状態
- \* rep Speak.stat = SPEAKING  
... 音声出力中の状態
- prop Speak.stat = AutoOutput  
... 現在の音声合成処理の状態を変更された時点で自動的に rep により返すように設定する。
- prop Speak.stat = NoAutoOutput  
... 現在の音声合成処理の状態を inq にて問い合わせた時に rep により返すように設定する。

## 6 顔画像生成部パラメータ

### 7 基本パラメータの直接制御

#### Face

動作 **set** 表情パラメータ値を直接設定する。表情パラメータ ID としては、制御対象の表情パラメータ ID を指定する。パラメータ値としては、制御対象の変形量 [%] を指定する。

1. **set Face.** 表情パラメータ ID = パラメータ値
  - 指定した表情パラメータ ID のパラメータ値を設定する。

属性 **Face.1** : 表情パラメータ ID = 1 (他のパラメータ ID も数字指定)

- **set Face.1 = 70** : 表情パラメータ ID 1 のパラメータ値を 100[%] に設定する。つまり、70[%] の変形率で変形する。

#### Mouth

動作 **set** 口形状パラメータ値を直接設定する。口形状パラメータ ID としては、制御対象の口形状パラメータ ID を指定する。パラメータ値としては、制御対象の変形量 [%] を指定する。

1. **set Mouth.** 口形状パラメータ ID = パラメータ値
  - 指定した口形状パラメータ ID のパラメータ値を設定する。

属性 **Mouth.1** : 口形状パラメータ ID = 1 (他のパラメータ ID も数字指定)

- **set Mouth.1 = 70** : 口形状パラメータ ID 1 のパラメータ値を 100[%] に設定する。つまり、70[%] の変形率で変形する。

#### MouthScale

動作 **set** 口形状の全体の変形量 [%] を設定する。

1. **set MouthScale =** パラメータ値
  - 口形状の全体の変形量を指定したパラメータ値を設定する。

例 **set MouthScale = 50**

- 口形状の全体の変形率を 50%とする

**inq** 口形状の全体の変形量 [%] を問い合わせる。

1. **inq MouthScale**
  - 現在の口形状の全体の変形量を問い合わせる。

#### HeadRotAbs

動作 **set** 正面を基準として、頭部を回転させる。回転軸 ID は制御対象の回転軸を整数で指定する。各方向回転量は [*degree*] を指定する。

1. **set HeadRotAbs.** 回転軸 ID = x 方向回転量 y 方向回転量 z 方向回転量
  - 指定した回転軸を中心に、頭部を回転させる。

属性 **HeadRotAbs.1** : 回転軸 ID = 1 (他の回転軸 ID も数字指定)

- **set HeadRotAbs.1 = 30 45 60** : 回転軸 1 を中心に、正面から x 軸方向に 30[*degree*]、y 軸方向に 45[*degree*]、z 軸方向に 60[*degree*] 頭部を回転する。

## 8 基本動作、基本表情

### FaceMot

動作 set 指定した動作 ID、動作パターン ID の基本動作を行う。動作 ID は BLINK、NOD、REFUSE、LISTEN、USERDEF の 5 種類。動作パターン ID は整数値で指定する。

1. set FaceMot = 動作 ID 動作パターン ID

- 指定した動作 ID、動作パターンの基本動作を行う。

例 1 set FaceMot = BLINK 1

- 動作パターン 1 に指定された「瞬き」を行う。

例 2 set FaceMot = NOD 1

- 動作パターン 1 に指定された「うなずき」を行う。

例 3 set FaceMot = REFUSE 1

- 動作パターン 1 に指定された「首を横に振る」を行う。

例 4 set FaceMot = LISTEN 1

- 動作パターン 1 に指定された「話を聴いているように首を傾ける」を行う。

例 5 set FaceMot = USERDEF 1

- 動作パターン 1 に指定された「ユーザ定義」の基本動作を行う。

### FaceExp

動作 set 指定した表情 ID、パターン ID の基本表情を指定の強度、継続長で行う。表情 ID は HAPPY、DISGUSTED、SAD、ANGRY、SURPRISED、FEARED、USERDEF の 7 種類。表情パターン ID は整数値で指定する。強度は表情の変形量 [%] を指定する。継続長は表情の継続長 [ms] を指定する。継続長が 0 のときは、次の表情変形指令を受けるまで、指定の表情を継続する。

1. set FaceExp = 表情 ID パターン ID 強度 継続長

- 指定した表情 ID、パターン ID の基本表情を指定の強度、継続長で行う。

例 1 set FaceExp = HAPPY 1 100 100

- 動作パターン 1 に指定された「喜び」の表情を強度 100[%]、継続長 100[ms] で行う。

例 2 set FaceExp = DISGUSTED 1 100 200

- 動作パターン 1 に指定された「嫌悪」の表情を強度 100[%]、継続長 200[ms] で行う。

例 3 set FaceExp = SAD 1 50 100

- 動作パターン 1 に指定された「悲しみ」の表情を強度 50[%]、継続長 100[ms] で行う。

例 4 set FaceExp = ANGRY 1 50 200

- 動作パターン 1 に指定された「怒り」の表情を強度 50[%]、継続長 200[ms] で行う。

例 5 set FaceExp = SURPRISED 1 100 0

- 動作パターン 1 に指定された「驚き」の表情を強度 100[%] で、次の表情変形指令を受けるまで継続する。

例 6 set FaceExp = FEARED 1 50 0

- 動作パターン 1 に指定された「恐れ」の表情を強度 50[%] で、次の表情変形指令を受けるまで継続する。

例 7 set FaceExp = USERDEF 1 50 0

- 動作パターン 2 に指定された「ユーザ定義」の表情を強度 50[%] で、次の表情変形指令を受けるまで継続する。

2. set FaceExp = NEUTRAL

- 「無表情」にする。つまり、表情パラメータをリセットする。

## 9 リップシンク

### set LipSync

動作 set 音声合成と同期した口形状アニメーションの準備を行う。

1. set LipSync. 属性名 = パラメータ
  - 指定した属性名とパラメータを基に音声同期準備を行う。

inq 口形状アニメーションの準備に関する問い合わせに対して rep により答える。

属性 LipSync.pho : 音素系列と音素継続長 [ms]。値は音声合成モジュールの Text.pho の値を利用する。

- set LipSync.pho = h 80 a 200 i 200 : 発話文章の音素系列と音素継続時間を設定する。
- inq LipSync.pho : 現在の設定されている発話文章の音素系列と音素継続時間を rep により返す。

### Speak

動作 set 既に準備された (音声出力と同期した) 口形状アニメーションを決められた時間に開始、もしくは停止する。

1. set Speak = + ミリ秒
  - 現在時刻を基準指定した相対時刻後に準備された音声出力を行う。
2. set Speak = NOW
  - すぐに既に準備された (音声出力と同期した) 口形状アニメーションを行う。
3. set Speak = STOP
  - 現在の (音声出力と同期した) 口形状アニメーションの中止を指示する。

inq (音声出力と同期した) 口形状アニメーションに関する問い合わせに対して rep により答える。

prop (音声出力と同期した) 口形状アニメーション出力に関する属性値の出力仕様を設定する。

1. prop Speak = AutoOutput
  - inq で問い合わせる出力スロットが変更された時点で rep により変更された値を答える。これは、スロット値が変更された時点で、そのスロット値を問い合わせる inq コマンドを自動生成したことに等しい。
2. prop Speak = NoAutoOutput
  - inq で問い合わせたときのみ、rep により値を答える。

属性 Speak : Speak に関する全情報

- inq Speak
  - ... 現在の Speak に設定されている全情報を rep により返す。
- prop Speak = AutoOutput
  - ... 現在の Speak に設定されている全情報を変更された時点で自動的に rep により返すように設定する。
- prop Speak = NoAutoOutput
  - ... 現在の Speak に設定されている全情報を inq にて問い合わせた時に rep により返すように設定する。

Speak.text : 発話内容

- inq Speak.text
  - ... 現在の設定されている発話文章を rep により返す。
- prop Speak.text = AutoOutput
  - ... 現在の設定されている発話文章を変更された時点で自動的に rep により返すように設定する。
- prop Speak.text = NoAutoOutput
  - ... 現在の設定されている発話文章を inq にて問い合わせた時に rep により返すように設定する。

Speak.pho : 音素系列と音素継続長

- inq Speak.pho  
... 現在の設定されている発話文章の音素系列と音素継続時間を rep により返す。
- prop Speak.pho = AutoOutput  
... 現在の設定されている発話文章の音素系列と音素継続時間を変更された時点で rep により返すように設定する。
- prop Speak.pho = NoAutoOutput  
... 現在の設定されている発話文章の音素系列と音素継続時間を inq にて問い合わせた時に rep により返すように設定する。

#### Speak.dur : 総発話時間

- inq Speak.dur  
... 現在の設定されている発話文章の総発話時間を rep により返す。
- prop Speak.dur = AutoOutput  
... 現在の設定されている発話文章の総発話時間を変更された時点で自動的に rep により返すように設定する。
- prop Speak.dur = NoAutoOutput  
... 現在の設定されている発話文章の総発話時間を inq にて問い合わせた時に rep により返すように設定する。

#### Speak.utt : 既に発話が完了した音素系列

- inq Speak.utt  
... 既に発話が完了した音素系列を rep により返す。
- prop Speak.utt = AutoOutput  
... 既に発話が完了した音素系列を変更された時点で自動的に rep により返すように設定する。
- prop Speak.utt = NoAutoOutput  
... 既に発話が完了した音素系列を inq にて問い合わせた時に rep により返すように設定する。

#### Speak.len : 既に発話が完了した発話時間

- inq Speak.len  
... 既に発話が完了した発話時間を rep により返す。
- prop Speak.len = AutoOutput  
... 既に発話が完了した発話時間を変更された時点で自動的に rep により返すように設定する。
- prop Speak.len = NoAutoOutput  
... 既に発話が完了した発話時間を inq にて問い合わせた時に rep により返すように設定する。

#### Speak.stat : 現在の音声合成処理の状態

- inq Speak.stat  
... 現在の音声合成処理の状態を rep により返す。
  - \* rep Speak.stat = IDLE  
... 何も処理していない状態
  - \* rep Speak.stat = PROCESSING  
... 発話準備中の状態
  - \* rep Speak.stat = READY  
... 発話準備が完了した状態
  - \* rep Speak.stat = SPEAKING  
... 音声出力中の状態
- prop Speak.stat = AutoOutput  
... 現在の音声合成処理の状態を変更された時点で自動的に rep により返すように設定する。
- prop Speak.stat = NoAutoOutput  
... 現在の音声合成処理の状態を inq にて問い合わせた時に rep により返すように設定する。

## 10 顔画像の選択

### MaskSet

動作 inq 使用可能な顔画像の問い合わせに対して rep により答える。

1. inq MaskSet
  - rep MaskSet = “male01, male02, female01”
  - \* 使用可能な顔画像の ID コード列

### Mask

動作 set 顔画像生成に使用する顔画像を設定する。

1. set Mask = male01
  - 顔画像生成に使用する male01(顔画像 ID) の指定する。顔画像 ID は MaskSet にて問い合わせた顔画像 ID を指定する。

## 11 自律動作

### Autonomous

動作 set 自律動作の設定を行う。動作 ID は BLINK、MOVE、USERDEF の 3 種類。パターン ID は整数値で指定する。また、「パターン ID=0」は、指定した自律動作の停止 (未設定) を意味する。

1. set Autonomous = 動作 ID パターン ID
    - 指定した動作 ID、パターン ID の自律動作を設定する。
- 例 1 set Autonomous = BLINK 1
- 動作パターン 1 に指定された「瞬き」の自律動作を設定する。
- 例 2 set Autonomous = MOVE 1
- 動作パターン 1 に指定された「顔の揺らぎ」の自律動作を設定する。
- 例 3 set Autonomous = USERDEF 1
- 動作パターン 1 に指定された「ユーザ定義」の自律動作を設定する。

## 12 エージェント 管理部の提供するコマンド 通信基本仕様

### 12.1 エージェント 管理部を介して各モジュールを制御するための通信基本仕様

タスク管理部から各モジュール間の通信は、必ずエージェント管理部を介して行われる。この各モジュールを制御するための通信仕様としては、次の3種類が提供される。

- module へ AM-DCL を介して直接コマンド送信
- module へ AM-MCL を介してコマンド送信
- module から AM-DCL を介してデータ出力

データ、及びコマンドの送信元(送信先、出力元)は、各モジュールの略称で指定する。以下に、その略称を示す。

TM ... タスク管理モジュール (Task Manager)  
AM ... エージェント管理部 (Agent Manager)  
SRM ... 音声認識モジュール (Speech Recognition Module)  
SSM ... 音声合成モジュール (Speech Synthesis Module)  
FSM ... 顔画像生成モジュール (Face expression Synthesis Module)

送信先指定では、最終的なコマンドの到達モジュール名を指定する<sup>1</sup>。  
各モジュールを制御するコマンドの詳細は、第2節以降に定義する。

#### 12.1.1 module へ AM-DCL を介して直接コマンド送信

##### 書式

to @送信先 コマンド

##### 使用例

```
to @FSM set FaceExp = HAPPY 2 100 500  
to @SSM.FSM set Speak = NOW
```

#### 12.1.2 module から AM-MCL を介してコマンド送信

##### 書式

[macro] コマンド

##### 使用例

```
macro set Speak = こんにちは  
macro set Speak = STOP  
set Speak = STOP
```

Macro Control Layer(AM-MCL) を介して、各モジュールを制御するためのコマンドパラメータの一覧を以下に示す。

<sup>1</sup> 例えば、TM から SRM にコマンド送信を行う際も AM を介して通信することになるが、TM から AM に送信するコマンドの送信先指定は SRM となる。

モジュールの起動、終了、問い合わせ	
パラメータ名	機能
Run	モジュールの起動、再起動、停止 モジュールの起動状態の問い合わせ
ProtocolVersion	モジュールの準拠している標準通信プロトコルの version 情報の問い合わせ
ModuleVersion	モジュール自体の version 情報の問い合わせ
同期管理マクロ	
パラメータ名	機能
Speak	リップシンクを伴う音声出力の開始、停止

### 12.1.3 module から AM-DCL を介してデータ出力

#### 書式

from @出力元 module からのメッセージ

#### 使用例

from @SSM rep Text.dur = 1000

## 13 エージェント 管理部の提供するコマンドパラメータ

以下に示すコマンドパラメータは、エージェント管理部の Macro Control Layer(AM-MCL) を介して、各モジュールを制御するためのコマンドパラメータを規定している。

### 13.1 モジュール起動、終了、および問い合わせ

#### Run

動作 set モジュールの動作を設定する。

- set Run. モジュール名 = EXIT
  - モジュールを停止する。

inq モジュールの動作状況を問い合わせに対して rep により答える。

- inq Run. モジュール名
  - From @モジュール名 rep Run = LIVE
    - \* 各モジュールが処理開始可能状態である時の出力
  - From @モジュール名 rep Run = DEAD
    - \* 各モジュールが処理開始不可能状態である時の出力

属性 Run.SRM : 音声認識モジュール

- inq Run.SRM : 音声認識モジュールの動作状態を rep により返す。

Run.SSM : 音声合成モジュール

- inq Run.SSM : 音声合成モジュールの動作状態を rep により返す。

Run.FSM : 顔画像生成モジュール

- inq Run.FSM : 顔画像生成モジュールの動作状態を rep により返す。

#### ProtocolVersion

動作 inq モジュールの準拠している標準通信プロトコルの version 情報の問い合わせに対して rep により答える。

1. inq ProtocolVersion. モジュール名

- From @モジュール名 rep ProtocolVersion = “Protocol Ver. 1.02”  
\* モジュールの Protocol version 情報

属性 ProtocolVersion.SRM : 音声認識モジュール

- inq ProtocolVersion.SRM : 音声認識モジュールの Protocol version 情報を rep により返す。

ProtocolVersion.SSM : 音声合成モジュール

- inq ProtocolVersion.SSM : 音声合成モジュールの Protocol version 情報を rep により返す。

ProtocolVersion.FSM : 顔画像生成モジュール

- inq ProtocolVersion.FSM : 顔画像生成モジュールの Protocol version 情報を rep により返す。

## ModuleVersion

動作 inq モジュール自体の version 情報の問い合わせに対して rep により答える。

1. inq ModuleVersion. モジュール名

- From @モジュール名 rep ModuleVersion = “Module Ver. 2.00”  
\* モジュールの version 情報

属性 ModuleVersion.SRM : 音声認識モジュール

- inq ModuleVersion.SRM : 音声認識モジュールの version 情報を rep により返す。

ModuleVersion.SSM : 音声合成モジュール

- inq ModuleVersion.SSM : 音声合成モジュールの version 情報を rep により返す。

ModuleVersion.FSM : 顔画像生成モジュール

- inq ModuleVersion.FSM : 顔画像生成モジュールの version 情報を rep により返す。

## 13.2 同期管理マクロコマンド

### Speak

動作 set 以下のようにして指定された文章を音声出力する。発話内容として “STOP” を指定した場合は、現在の発話を中止する。“JEIDA-62-2000 日本語テキスト音声合成用記号の規格” に準拠した形式で発話内容を指定する。

1. set Speak = 発話内容

- 指定した発話内容の音声出力をリップシンクを伴い行う。

2. set Speak < ファイル名

- 指定したファイルに記述された発話内容の音声出力をリップシンクを伴い行う。

3. set Speak << END

発話内容

END

- 指定した発話内容の音声出力をリップシンクを伴い行う。

4. set Speak = STOP

- 現在の音声出力の中止を指示する。