

次世代ロボット知能化技術開発プロジェクト
施設内生活支援ロボット知能の研究開発

音声認識モジュール(OGSS)

マニュアル

Ver. 2.1

2012 年 1 月

九州工業大学

改變履歷

[illegible]

1. はじめに	4
1.1. 本文書について	4
1.2. 対象プラットフォーム	4
1.3. OpenRTM-aist-C++ 関係	4
1.4. OpenRTM-aist-Java 関係	4
2. 音声認識モジュール(OGSS)	6
2.1. インストール	6
2.2. ビルド	6
2.3. ポートについて	7
2.4. 起動	9
2.5. 動作検証	11

1. はじめに

1.1. 本文書について

本書は、「次世代ロボット知能化技術開発プロジェクト」の「施設内生活支援ロボット知能研究開発」において構築した音声認識モジュールについてのマニュアルです。本書は RT ミドルウェア (以下 RTM)、RT コンポーネント (以下 RTC) を用いたロボットシステム開発者を対象に記述されており、RTM、RTC や関連ツールに関する一般的な知識を持つことを前提とします。

OpenRTM-aist official website: <http://www.openrtm.org/openrtm/ja/>

1.2. 対象プラットフォーム

本モジュールは、以下の環境で動作確認しています。

- Windows XP professional 32bit
- Windows Vista Business 32bit/64bit
- Windows 7 Professional/Enterprise 32bit/64bit

1.3. OpenRTM-aist-C++ 関係

以下のソフトウェアは必要ありませんが、インストールしておいた方がよいでしょう。

- OpenRTM-aist-1.0.0-RELEASE_vc9_100212.msi

1.4. OpenRTM-aist-Java 関係

以下のソフトウェアをインストールしてください。

- Java Deverer Kit JDK 1.6.0_XX (Eclipse の関係で 32bit 版が必要)
 - 環境変数 JAVA_HOME に C:¥Program Files¥Java¥jdk1.6.0_XX を設定する。
 - 64bit 環境に 32bit 版をインストールする場合は以下の通りにする。
 - 環境変数 JAVA_HOME に C:¥Program Files (x86)¥Java¥jdk1.6.0_XX を設定する。
 - 環境変数 PATH に %JAVA_HOME%¥bin; を追加する。
- OpenRTM-aist-Java-1.0.0-RELEASE
 - OpenRTM-aist-Java-1.0.0.msi を実行する。
 - 環境変数 RTM_JAVA_ROOT に C:¥Program Files (x86)¥OpenRTM-aist¥1.0 を設定する。
- Eclipse 3.4.2 (Ganymede SR2)
 - OpenRTM 全部入り eclipse342_rtmtools110-rc2_wn32_ja.zip 等をインストールする。
 - 或は eclipse と rtmtools110-rc2_ja.zip 等をインストールする。

- ※ eclipse が 32bit 版なので、JDK は 32bit 版が必要です。
- eclipse の「ウィンドウ」メニューの「設定」の「Java」の「コンパイラ」の「コンパイラ標準レベル」を 1.6 に設定する。
- Apache Ant 1.8.2
 - 環境変数 ANT_HOME に C:\apache-ant-1.8.2 を設定。
 - 環境変数 PATH に %ANT_HOME%\bin; を追加する。
 - ※ eclipse がインストールされている場合は、環境変数 ANT_HOME に
 - \$ECLIPSE_HOME/plugins/org.apache.ant_xxx を設定して利用してもよい。

2. 音声認識モジュール(OGSS)

複数認識器による統合音声理解モジュールのための RTC です。Julius を基盤としています。本モジュールには、Julius 自体が含まれています。本モジュールでは、含まれるファイルが Win32 を前提としていますので、他の OS で動作させる場合には注意して下さい。

モジュールのディレクトリは OGSS となります。

このモジュールは、Windows の標準のマイク入力の音声を入力するモジュール(例えば、OpenHRI の portaudioinput 等)と接続し、認識結果を TimedString のデータポートから出力します。

2.1. インストール

RTC 再利用センター登録の zip ファイルをダウンロードしてください。

音声認識モジュール: OGSS.2.0.zip

以下の手順で eclipse にプロジェクトをインポートします。

1. eclipse を起動する。
2. 作業計画モジュール(SDLEngine)をインストールする。
3. ファイルメニューから「インポート」を選択し、「一般」の「既存のプロジェクトをワークスペースへ」を選ぶ。
4. 次に進み、「アーカイブ・ファイルの選択」の欄にチェックし、「参照」で OGSS.2.0.zip を選択する。
5. 「完了」を選択する。

2.2. ビルド

1. ビルドには、特に必要なソフトウェアなどはありません。
2. Eclipse によるビルドを実行して下さい。

ビルド後に問題が発生していなければ、ビルドは完了です。多数の警告が出ますがここでは無視してください。

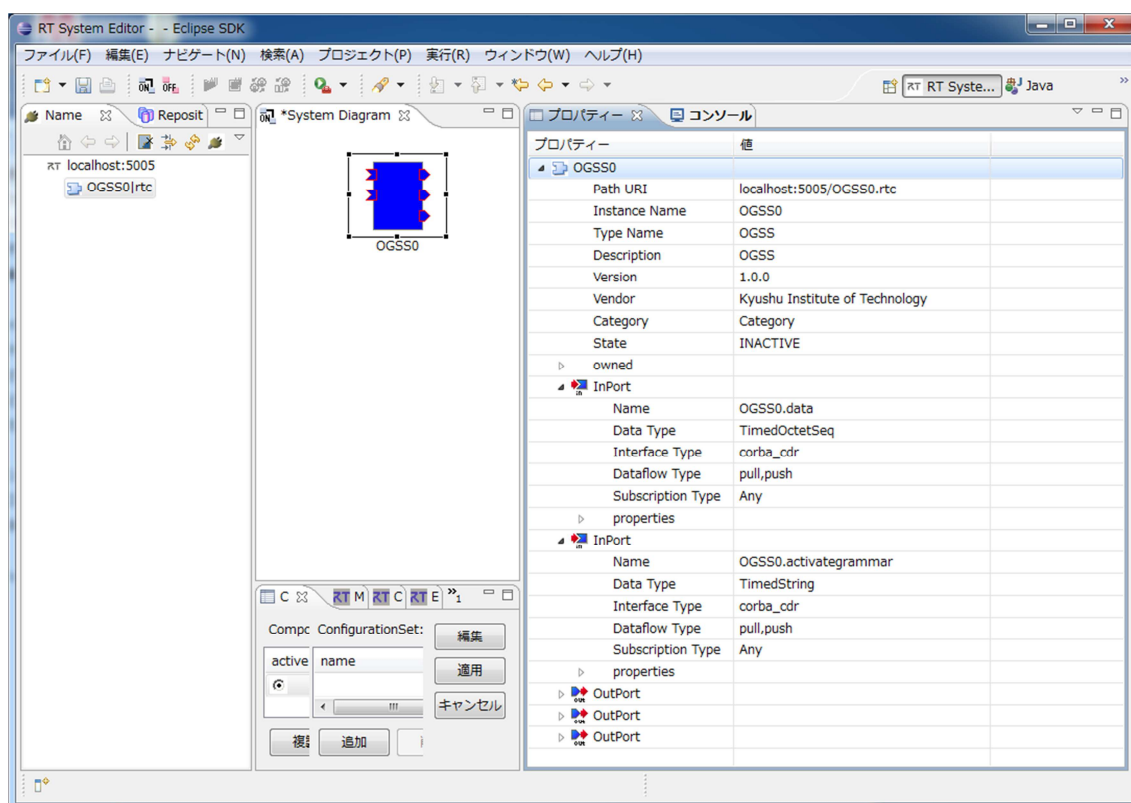
2.3. ポートについて

OGSS コンポーネントは OpenHRI の音声認識コンポーネントである **juliusrtc** と入出力ポートレベルで互換があります。ただし、音声データの入力ポートである **data** と認識結果の出力ポートである **result** 以外は未サポートですので使用しないでください。

データポートの一覧表

入出力	ポート名	データ型	説明
入力	data	TimedOctetSeq	解析する音声データ入力
入力	activegrammar	TimedString	未サポート
出力	status	TimedString	未サポート
出力	result	TimedString	解析された音声認識結果
出力	log	TimedOctetSeq	未サポート

入力ポートのプロパティ



出力ポートのプロパティ

The screenshot shows the RT System Editor interface. The central diagram area displays a component labeled 'OGSS0' with several red arrow ports. The right-hand 'Properties' window is open, showing the configuration for the selected output port, 'OGSS0.log'.

プロパティ	値
Vendor	Kyushu Institute of Technology
Category	Category
State	INACTIVE
owned	
InPort	
InPort	
OutPort	
Name	OGSS0.status
Data Type	TimedString
Interface Type	corba_cdr
Dataflow Type	pull,push
Subscription Type	flush,new,periodic
properties	
OutPort	
Name	OGSS0.result
Data Type	TimedString
Interface Type	corba_cdr
Dataflow Type	pull,push
Subscription Type	flush,new,periodic
properties	
OutPort	
Name	OGSS0.log
Data Type	TimedOctetSeq
Interface Type	corba_cdr
Dataflow Type	pull,push
Subscription Type	flush,new,periodic
properties	

2.4. 起動

このコンポーネントは、起動時に **OGSS** プロジェクトの下の **rtc.conf** をデフォルトでは読み込みます。実行環境に合わせて書き換えてください。

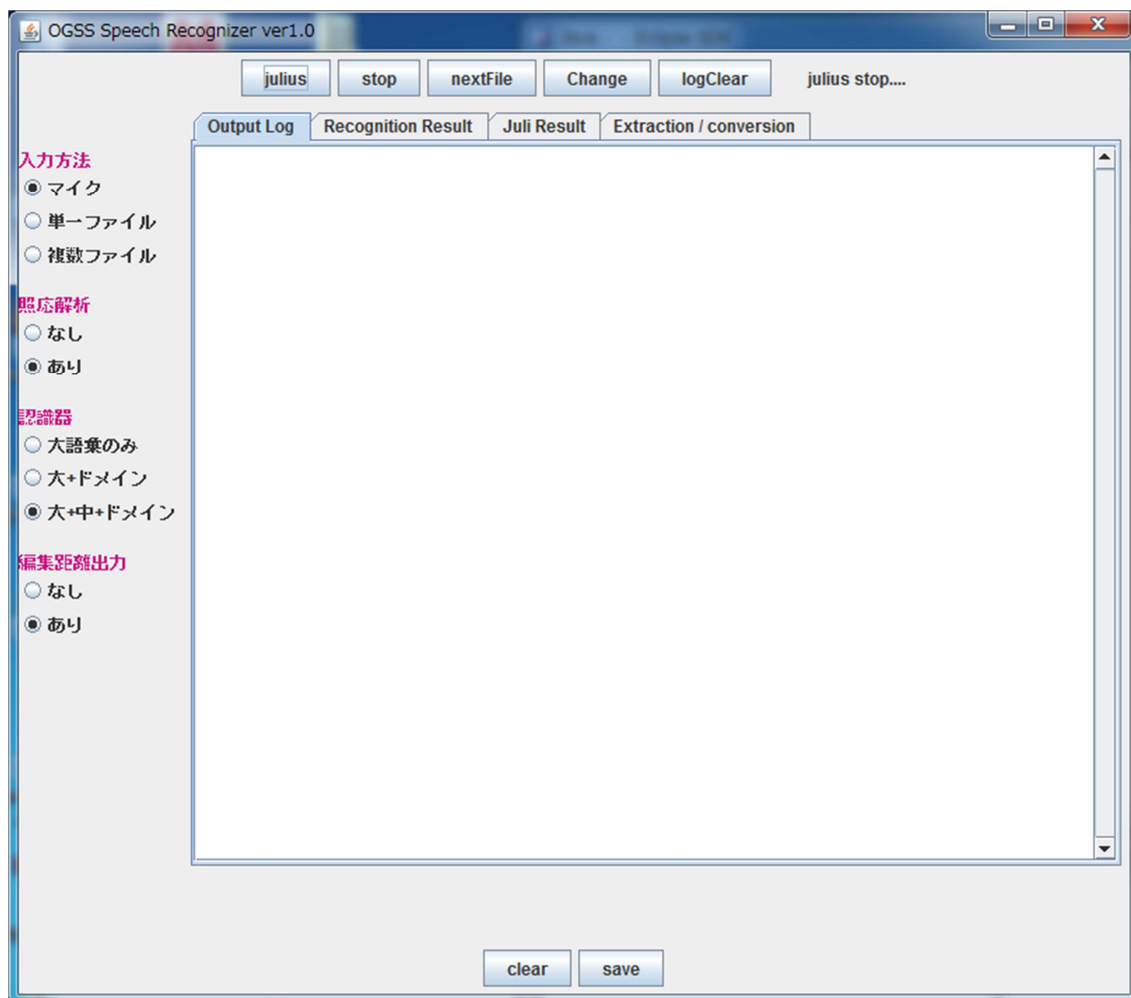
デフォルトの状態では、ネームサーバをポートを **5005** で起動しておいてください。

Eclipse の実行メニューの実行構成の **Java** アプリケーションを選び、メイン・クラスの欄に **jp.ac.kyutech.SRP** 中の **OGSSComp** クラスを選択します。後は、実行で起動して下さい。

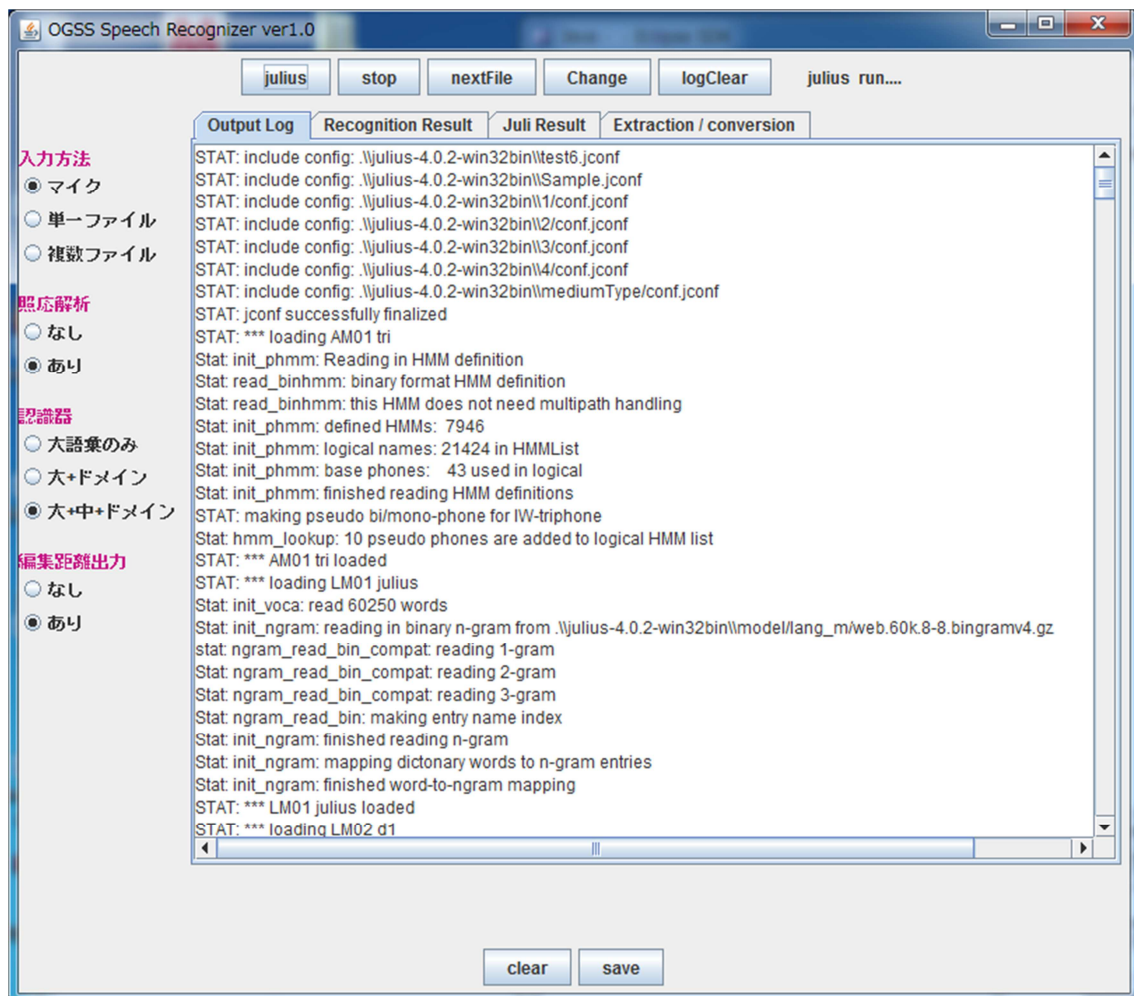
正常に起動すると、次のような画面が表示されます。

この画面の上部にある **julius** ボタンを押すと認識を開始します。以下の図は、起動時と認識開始時の実行例です。

OGSS の起動時



julius ボタンを押して認識を開始した状態



終了するには、ウィンドウの右上の×印を押してください。

2.5. 動作検証

動作検証には、作業計画モジュール (SDLEngine) と OpenHRI の portaudioinput を用いた場合を紹介します。

1. スタートメニューから OpenHRI→audio→portaudioinput を起動します。
2. OGSSコンポーネントを起動して認識を開始します。
3. SDLEngine を起動します。
4. 以下のスクリプトを SDLEngine Console で実行します。

```
env = rtc.env("localhost", 5005);
sdLEngine = rtc.local_component("SDLEngine", "SDLEngine");
env.get_handles();
env.connect(env.handles{"SDLEngine0.rtc"}.ports{"SDLEngine0.stringIn"},
env.handles{"OGSS0.rtc"}.ports{"OGSS0.result"});
env.connect(env.handles{"OGSS0.rtc"}.ports{"OGSS0.data"},
env.handles{"PortAudioInput0.rtc"}.ports{"PortAudioInput0.AudioDataOut"});
env.handles{"SDLEngine0.rtc"}.activate();
env.handles{"OGSS0.rtc"}.activate();
env.handles{"PortAudioInput0.rtc"}.activate();
sdLEngine.local_ports{"SDLEngine0.stringIn"}.addListener(new jp.ac.kyutech.SRP.Scripting.InPortListener() {
    dataReceived(event) {
        print("Received: " + event.getValue().data);
    }
});
```

5. “ペンを取って” とマイクに向かって話して、結果が XML として受け取れれば OK です。