

次世代ロボット知能化技術開発プロジェクト
対話履歴管理コンポーネント
マニュアル

2011年 2月 (Ver. A)

株式会社国際電気通信基礎技術研究所
知能ロボティクス研究所

・改訂履歴

Ver.	改訂日付	改訂内容	作成
NC	2011-01-17	初版	岩崎
A	2012-02-21	誤記訂正	岩崎

目 次

1. 目的.....	3
2. 単体テスト接続使用方法	3
3. 組み合わせテスト接続使用方法	4
4. テスト入力サンプル（参考）	4
5. テスト用モジュール仕様（参考）	5
5.1 動作環境（共通）	5
5.2 DebugConInComp	6
5.3 DebugConOutComp	6

1. 目的

対話履歴管理コンポーネント(HistoryManModule)の動作確認テストの方法について記述する。
コンポーネントの詳細については、以下のドキュメントを参考のこと。

- ・対話履歴管理コンポーネント外部仕様書

また、対話履歴管理コンポーネント内で使用しているデータベースを直接操作するためのツール(HistoryDBConfigure)を添付している。

ツールの詳細については、以下のドキュメントを参考のこと。

- ・HistoryDBConfigure 使用説明書

2. 単体テスト接続使用方法

(1) RTM 用 CORBA ネームサーバー(rtm-naming.bat)を起動する。

(2) Windows 上のコマンドプロンプトで、以下のコンポーネントを起動する。

HistoryManModuleComp - 動作生成コンポーネント

DebugConInComp - デバッグ用コンソール入力コンポーネント

DebugConOutComp - デバッグ用コンソール出力コンポーネント

(3) 図 1 の通り RTSystemEditor で接続し、RTSystemEditor で All Activate する。

(4) 4 項のテスト入力サンプルを参考にして、DebugConInComp からコマンドを入力し、
DebugConOutComp に応答が返ることを確認する。

DebugConInComp 入力例:

```
Microsoft Windows XP [Version 5.1.2600]
```

```
C:\¥..¥components> DebugConInComp
```

```
[DebugConIn] START
```

```
Input= <SEL human1 1 name1>
```

```
Input=
```

DebugConOutComp 出力例:

```
Microsoft Windows XP [Version 5.1.2600]
```

```
C:\¥... ¥components> DebugConOutComp
```

```
[DebugConOut] START
```

```
recv=[<SEL OK 太郎>]
```

- (5) コンポーネントの終了時は、RTSystemEditor で All Deactive にしてから Exit した後、各コマンドプロンプト画面において、Ctrl-C でプログラムを終了させる。

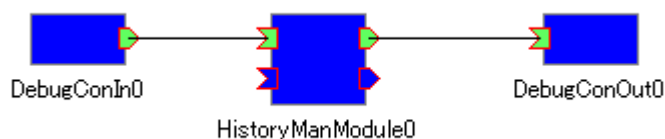


図 1. RTSystemEditor 接続

3. 組み合わせテスト接続使用方法

組み合わせテストについては、以下のドキュメントを参考のこと

- ・対話制御マニュアル（対話フロー制御に含まれる）

4. テスト入力サンプル（参考）

以下のコマンドは、データベースの動きを理解した上で入力すること。

●対話履歴ログ保存

```
<LOG 20080101000000 20080101000010 1 2 3,4 1 name1 talk1 こんにちは>  
<LOG 20080101000020 20080101000030 2 3 4,5 1 name2 talk2 さようなら>  
<LOG 20080101000040 20080101000050 3 4 5,6 1 name3 talk3 またあした>  
<LOG 20080101000060 20080101000070 4 5 6,7 0 name4 talk4 いい天気>
```

●行追加

```
<INS human1 2>  
<INS human1 3>  
<INS human1 4>  
<INS human1 5>
```

●データ更新

```
<UPD human1 1 name1 太郎>  
<UPD human1 2 name1 次郎>  
<UPD human1 3 name1 三郎>  
<UPD human1 4 name1 次郎>
```

```
<UPD human1 1 name2 山田>  
<UPD human1 2 name2 田中>  
<UPD human1 3 name2 本田>
```

<UPD human1 4 name4 エラー>

●行削除

<DEL human1 1>

<DEL human1 2>

<DEL human1 3>

<DEL human1 4>

<DEL human1 5>

●データ照会

<SEL human1 1 name1>

<SEL human1 1 name1 name2>

●データ検索

<SRC human1 name1 太郎>

<SRC human1 name1 次郎>

●データ増加

<INC human1 1 count 1>

●行カウント

<CNT human1 name1 太郎>

<CNT human1 name1 次郎>

<CNT human1 name1 次郎>

●列追加

<ACL human1 age INTEGER>

<ACL human1 name3 TEXT>

<ACL human1 name4 TEXT>

●列名取得

<GCN human1>

5. テスト用モジュール仕様（参考）

5.1 動作環境（共通）

- ・使用 OS : WindowsXP SP3 + Microsoft .NET Framework 3.5
- ・Visual C++ 2008 Express Edition
- ・RT ミドルウェア(OpenRTM-aist-1.0.0)

5.2 DebugConInComp

デバッグ用コンソール入力コンポーネントであり、コンソールから入力された文字列(S-JIS 入力)を、TimedString 型(UTF-8)で出力する。

入出力ポートの構成を以下に示す。

(1)コマンド出力ポート

ポート名 : DBGout

型 : OutPort TimedString×1

漢字コード : UTF-8N

5.3 DebugConOutComp

デバッグ用コンソール出力コンポーネントであり、TimedString 型(UTF-8)で入力された文字列を、コンソール(S-JIS)に出力する。

入出力ポートの構成を以下に示す。

(1)デバッグ入力ポート

ポート名 : DBGin

型 : InPort TimedString×1

漢字コード : UTF-8N