

Interaction Builder*

5C-01 マルチモーダル対話システムのためのプロトタイピングツール

足立裕秋† 小林剛典† 桂田浩一† 小林聡‡ 山田博文‡ 新田恒雄‡

豊橋技術科学大学 大学院工学研究科† 豊橋技術科学大学 工学部‡

1. はじめに

本報告では、マルチモーダル対話(MMI)システムの構築を支援するツール、Interaction Builder(IB)について述べる。我々は、MMIを記述する言語、XISL(Extensible Interaction-Sheet Language)を提案してきた[1], [2], [3]。XISLは、複数のモダリティを連携させ、自然で判りやすい対話を実現するための言語である。XISLの現在の仕様では、開発者がシステムとユーザとのやり取りを詳細に記述できる反面、記述リストが長くなるという問題がある。

IBは、GUI操作によってXISLを記述することを目的としたプロトタイピングツールで、完成後、ツールの一部(音声対話と擬人化エージェントの統合)は、情報処理振興協会(IPA)の擬人化音声対話エージェント基本ソフトウェア[4], [5]に含まれる予定である。システムとユーザ間の様々なモダリティを介したやり取りは、このツールを利用することで容易に記述でき、XISLベースのラピッドプロトタイピングが可能となることが期待されている。

MMIを持つプロトタイプシステムでは、システムにより、使用するモダリティが異なる。例えばPC上で動作するシステムでは、ディスプレイ、音声合成、キーボード、マウス等が、一方、携帯電話では音声入力、DTMFキー等が用いられる。従ってIBが生成するXISLは、対象のプロトタイプシステムにより異なるモダリティをサポートする必要がある。XISLは入出力モダリティを記述する際に、新規モダリティの追加がし易いといった、自由度を持っている。今回のIBのバージョンでは、PC端末上で動作

```
hello.xisl
<?xml version="1.0" encoding="Shift-JIS" ?>
<!DOCTYPE xisl SYSTEM "xisl.dtd" >
<xisl version="1.0" >
  <head>...</head>
  <body>
    <dialog id="hello World" >
      <exchange>
        <operation target="hello.xml" >
          <input type="touch" event="click"
            match="/page" >
        </operation>
        <action>
          <output type="speech" event="tts-speech" >
            <![CDATA[
              <param name="speech-text" >
                Hello World!
              </param>
            ]]>
          </output>
        </action>
      </exchange>
    </dialog>
  </body>
</xisl>
```

図1 XISLの記述例

するオンラインショッピングシステム(OLSシステム)[6]を対象に、システムで利用されるモダリティに関する記述をサポートした。

以降、XISL及び、OLSシステムについて説明した後、IBの機能と実行例を述べる。

2. マルチモーダル対話記述言語 XISL

XISLは、XMLコンテンツに対するMMIを記述する言語である。XISLでは、ある目的を持った1組みの対話をdialogで記述し、複数のdialogにより対話シナリオを構成する。各dialogは対話の最小単位を表すexchangeから構成されている。また、各exchangeはoperation(入力記述部)とaction(動作記述部)をそれぞれ一つずつ持つ。

*Interaction Builder : a Prototyping tool for Multi-Modal Interaction Systems.
H. Adachi†, T. Kobayashi†, K. Katsurada†, S. Kobayashi ‡, H. Yamada†, T. Nitta†
†Graduate School of Engineering, Toyohashi Univ. of Tech. ‡ Faculty of Engineering, Toyohashi Univ. of Tech. ‡

単一の入出力モダリティは、operation 内部の input と action 内部の output により表される。input にはシステムが受け付けるユーザからの入力を、output にはそれに対応する動作を記述する。また、operation や action 内では、マルチモーダルな入出力を制御するタグが用意されている。これらのタグによって、逐次的(sequential)入出力、同時並行的(parallel)入出力、および択一的(alternative)入出力を記述することができる。

図1に XISL の例を示す。この例では、hello.xml の<page>の内容が、XSL によりブラウザに表示されていることを想定し、表示指定箇所をクリックされたときに、システムが“Hello World”と音声で応答する対話例の記述を示している。

3. OLS システムの概要

OLS システムは、XISL を用いたプロトタイプシステムの一つで、我々のグループで開発中の MMI プラットホーム上に、OLS 用の XML, XSL, XISL ドキュメントを組み込み、開発している。ドキュメント制作では、最初に XML で商品情報と画面情報等のコンテンツを記述し、次に XSL で画面のスタイルを指定し、最後に XISL によりユーザとシステムとのインタラクションを記述する。

3.1 OLS システムの入出力モダリティ

OLS システムは PC 端末上で動作し、MMI のための様々なモダリティを利用できる。表1に今回開発した OLS システム上で利用可能なモダリティを示した。

表1 OLS システムで利用できるモダティ

	モダリティ	イベント (入力)、動作 (出力)
入力モダリティ	マウス	クリック、ダブルクリックなど
	音声	音声認識
	キーボード	キー入力
出力モダリティ	ウィンドウ	コンテンツの表示
	サウンド/動画	サウンド、動画の再生
	擬人化エージェント	アクション スピーチ、バルーン
	音声合成	テキストの読み上げ

3.2 XISL による対話シナリオの記述

OLS システムにおいて、対話シナリオを XISL で記述する際のメリットを以下にまとめて述べる。

- ・ 保守性・再利用性の向上
 - OLS システムでは、商品・顧客データ(XML)、スタイル(XSL)、及び対話シナリオ(XISL)の三つの構成要素をそれぞれ独立に取り扱っている。これにより、商品在庫数や顧客情報の管理がし易く、データの保守性が向上する。また、対話シナリオやビューを独立に変更できることで、再利用性の高いシステムを実現している。
- ・ 新たなモダリティが追加し易い
 - XISL では、モダリティ制御の記述に関して、端末とのデータ授受の形式を規定するに留めている。これにより、XISL の仕様を変更することなく、OLS システムのインタフェースに、新たなモダリティを追加することができる。
- ・ 詳細な対話シナリオを記述できる
 - XISL のマルチモーダル対話記述力により、システムの動作を詳細に指定することができる。

一方、多くのメリットの反面、システム開発者は以下の負担も生じる。

- ・ 扱うデータの構造を把握しなければならない。
- ・ XISL による記述量が増加し、OLS システムのドキュメントを製作する負担が大きくなる。
- ・ 端末に依存したモダリティに関して、動作記述方法を習得しなければならない。

こうしたことから、OLS プロトタイプシステムを迅速に開発することは困難であった。

4. IB の機能

IB は、これまでに述べた問題点を克服し、OLS システムの迅速な構築を目指す。図2に IB の全体図を示す。以降 IB の機能について述べる。

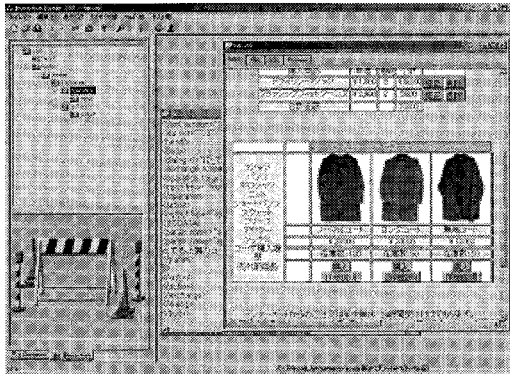


図 2 IB の実行画面



図 3 入出力モダリティ指定 Box

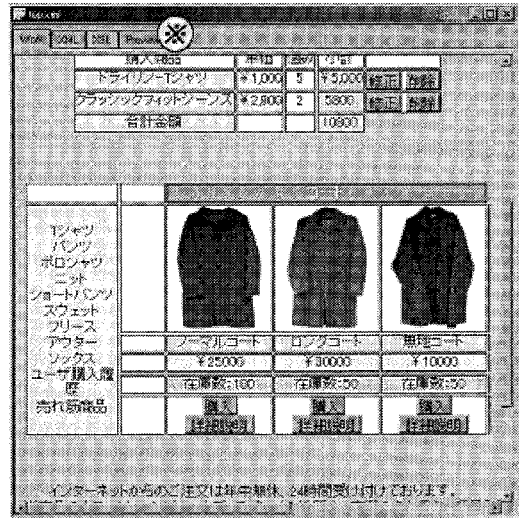


図 4 ページウィンドウ(PW)

4.1 モダリティの指定

IB の現バージョンで取り扱うことのできるモダリティは、表 1 で示したモダリティのうち、マウス、音声入出力、擬人化エージェントを組み合わせたものである。他のモダリティについても、今後、順次サポートしていく予定である。

システム開発者がモダリティ制御に関する記述を行なう場合には、IB 上部のツールバーにある入出力モダリティ指定 Box (Input Tool Box, Output Tool Box) を使用する。図 3 はこれを拡大したものである。左の Box から順に、音声入力、マウス入力、音声とマウスによる複合入力、及び二種類の擬人化エージェントによる応答を表している。

各 Box を選択すると、それぞれのモダリティの記述に必要な情報(音声入力では、語彙・文法など、擬人化エージェント出力では発話内容・アニメーションの種類など)を指定するダイアログボックスが表示される。

Tool Box を利用することで、システム開発者は OLS が規定しているモダリティの制御仕様を理解していなくとも、入力操作とそれに対する出力応答を XISL で記述することができる。

4.2 IB のビューの構成

IB は、複数の子ウィンドウを表示することができる。ウィンドウには、図 4 に示すページウィンドウ (PW) と、図 5 に示すインタラクションウィンドウ (IW) の二種類がある。

4.2.1 ページウィンドウ(PW)

PW には、図 4 の※印の箇所に示した、各種データを表示するビューを切り替えるタブが付いており、1つの PW で OLS システムの 1 ページを管理する。このタブをクリックすることにより、表示させるビューを選択できる。PW のビューの種類を、以下に示し説明する。

・ Work View

ページに対する 1 ターンの対話を定義する exchange を生成する作業画面である。1 ターンの対話とは、例えば、ボタンがクリックされると、擬人化エージェントがあいさつをするといった、ユーザからの入力操作に対する、システムからの出力応答の組みを指す。

IB では、ページ上に表示された画像やボタン、文字列などをページオブジェクトとして扱う。これらのオブジェクトは、Work View 上に、実際の画面

と同様の配置で表示される。開発者はここで、オブジェクトが受け付けることのできる操作や、それに対する応答を、上述した入出力モダリティ指定 Box を利用して、対話シナリオを作成する。

・ XML View 及び XSL View

OLS システムの 1 ページは、画面情報を記述した XML ドキュメントと、そのページのスタイルを記述した XSL ドキュメントから構成されている。IB では、XML View 及び XSL View により、これらのドキュメントを表示する。

・ Preview View

実際のシステムに表示される画面を確認することができる。現バージョンでは画面表示機能のみであるが、今後、作成した XISL の動作テストを行なう機能等の拡張を考えている。

4.2.2 インタラクシオンウィンドウ(IW)

Work View でページに対するインタラクシオンを指定することにより、XISL ドキュメントが生成される。図 5 に示す IW では、実際に生成された XISL ドキュメントを確認するための、XISL View を提供する。この View は、テキストエディタの機能を備え、XISL を直接編集することができる。

今後、XISL View の他、ページ間の移動など対話

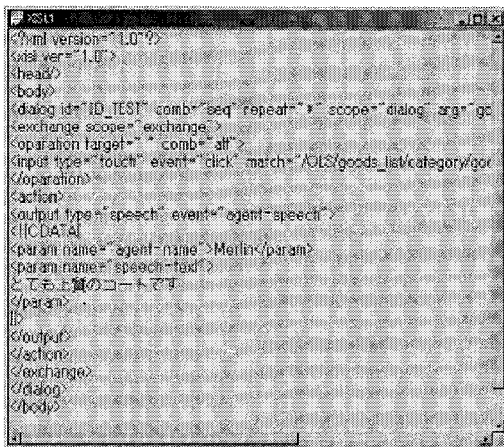


図 5 インタラクシオンウィンドウ(IW)

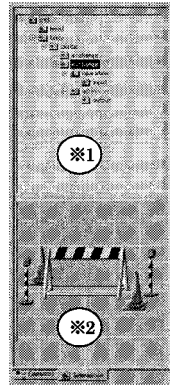


図 6 XISL ツリー

の遷移をグラフ表示する機能を持った View も提供する予定である。

4.3 XISL ツリー

図 6 は IB の左側に配置されるドッキングウィンドウを拡大したものである。生成された XISL ドキュメントは、図 6 の“※1”印に示す XISL Tree View に、その構造がツリー形式で表示される。例えば、Work View で exchange を生成すると、このツリーの項目中 exchange が追加される。このツリーの項目中からは、作成された情報(この場合は exchange に関する情報)を参照することができる。

例として挙げた exchange の情報は、図 6 の“※2”印に示す exchange View に表示する予定である。ここでは、使用されたモダリティや、モダリティの利用形態(sequential, parallel, alternative)を確認することができる。

5. 実行例

この節では実際に IB を用いて、OLS システムの 1 ページに対する対話シナリオ作成の過程を、画面例から説明する。

Step-1:

メニューバー上の“ファイル”から、メニュー“開く”を選択すると、OLS ページ選択のためのダイアログボックスが表示される。ここで目的のページの画面情報を記述した XML ファイルを選択すると、

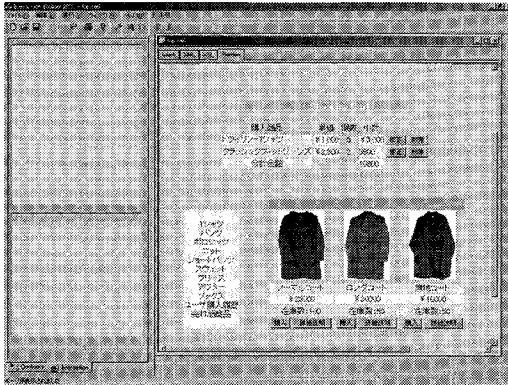


図 7 Step-1 の実行画面

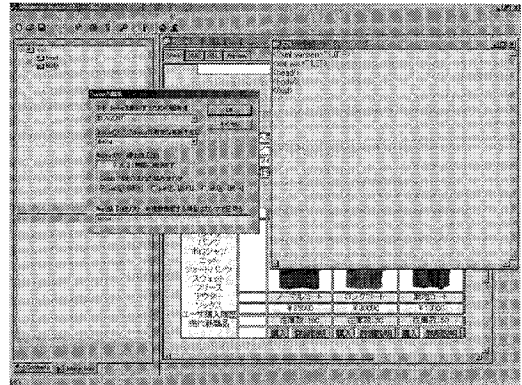


図 9 Step-3 の実行画面

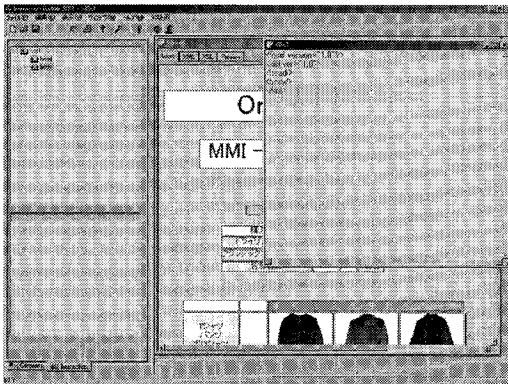


図 8 Step-2 の実行画面

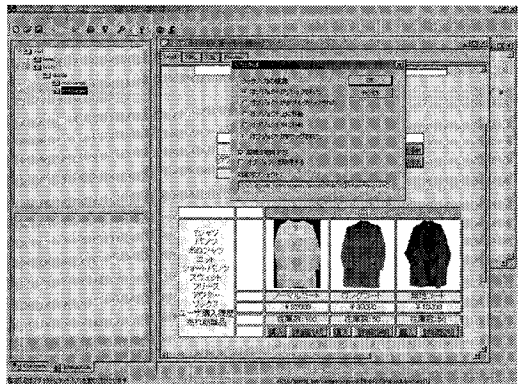


図 10 Step-5 の実行画面

図 7 に示すように、PW が開かれる。同時に、読み込んだ XML データ中の stylesheet 処理命令で指定されている XSL ファイルも読み込まれ、Preview View に画面が表示される。

XML・XSL View では、XML、XSL の各ドキュメントが表示され、ここで編集を行なうことも可能である。

Step-2:

PW 上の Work タブを選択し、ページに対するインタラクションを貼り付ける Work View を呼び出すと、図 8 に示すように、同時に IW が立ち上がり、XISL の雛型が生成される。

Step-3:

XISL TreeView を右クリックすると、XISL メニュー

が表示される。そこから“dialog の作成”を選択すると、図 9 に示すように、dialog の名前や繰り返し回数等を指定するダイアログボックスが表示される。ここで生成される dialog は、ある目的を持った 1 組の対話を構成する。

Step-4:

XISL メニューから、“exchange の作成”を選択し、先に生成された dialog に exchange を登録する。exchange は対話の最小単位で、以降、全ての exchange に、受け付ける入力と対応する応答を指定していく。

Step-5:

インタラクションの対象となるオブジェクトを Work View から選択した後、Input Tool box から入力

モダリティを指定する。図 10 では商品画像を選択し、タッチ入力を指定している。すると、タッチ入力を受け付ける上で必要な属性等を指定するダイアログボックスが開かれる。

Step-6:

Step-5 で指定した入力に対するシステムの応答動作を、Output Tool Box から指定する。図 11 では、擬人化エージェントを選択し、商品の説明を行なわせるための属性ほかを記述している。

Step-7:

図 12 は、Step-4 ~ Step-6 の操作で作成された対話記述リストを示す。このリストは、OLS システムのページに表示された商品画像をクリックし、擬人化エージェントに商品説明を行なわせるという場面に相当する。

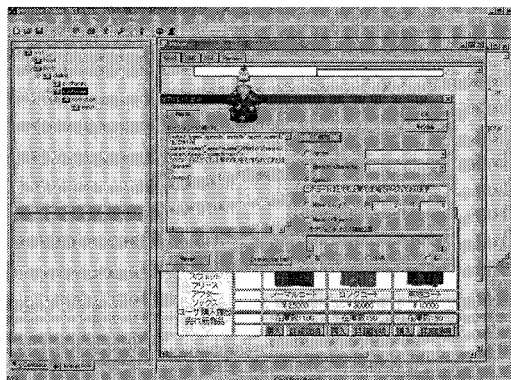


図 11 Step-6 の実行画面

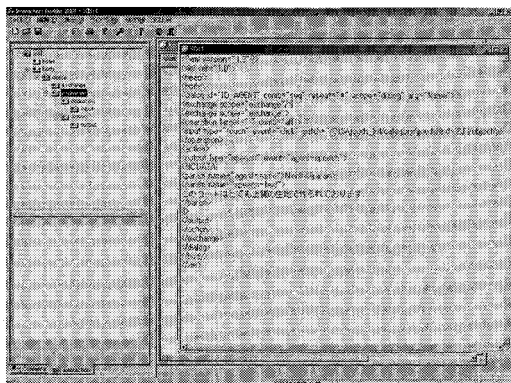


図 12 1対話に相当する XISL リスト

以上の操作を繰り返すことで、複数の exchange を生成しながら、組み合わせることでマルチモーダル対話のシナリオを記述していく。

6. まとめ

本報告では、XISL に基づくプロトタイピングツール IB について述べた。本ツールを利用することにより、OLS で規定したモダリティを用いた対話シナリオを容易に開発することができる。また、新たなモダリティの追加に柔軟に対応できるように設計されており、OLS だけでなく、他のドメインのプロトタイピングにも適用できると考えている。

今後は、評価テストを通して IB の有効性を検証するとともに、IB の機能を一層高度化したい。

参考文献

- [1] 小林聡他：“マルチモーダル対話記述言語XISLの提案”，情報処理学会研究報告，2001-SLP-37，pp.43-48，(2001).
- [2] 中村有作他：“XISL：コンテンツ記述とインタラクション記述分離の試み”，情報処理学会第62回全国大会講演論文集(分冊4)，7Q-1，pp.71-72，(2001).
- [3] 桂田浩一他：“音声対話記述言語 VoiceXML と MMI 記述言語 XISL の比較”，情報処理学会研究報告 2001-SLP-38，pp.49-54 (2001).
- [4] 川本真一他：“擬人化音声対話エージェントツールキットの基本設計”，情報処理学会研究報告 2001-SLP-40-11，Feb. (2002).
- [5] 西本卓也他：“対話システムにおけるタスク記述とプロトタイプ作成支援”，情報処理学会研究報告 2001-SLP-40-12，Feb. (2002).
- [6] 小林剛典他：“MMI 記述言語 XISL を用いたオンラインショッピングシステムの試作”，情報処理学会第 64 回全国大会講演論文集，6V-04，(2001).