

マルチモーダル対話システムにおけるユーザ環境情報の利用

大隈 祐治 桂田 浩一 新田 恒雄

豊橋技術科学大学 大学院工学研究科 知識情報工学専攻

〒441-8580 愛知県豊橋市天伯町雲雀ヶ丘 1 - 1

E-mail: {aoki,katurada,nitta}@vox.tutkie.tut.ac.jp

あらまし

本報告では、マルチモーダル対話 (MMI) システムにおいて端末情報、環境情報、ユーザプロフィール (年齢、性別、嗜好、...) (以上をまとめてユーザ環境情報と呼ぶ) を取り扱うための枠組みを検討する。まず、ユーザ環境情報を MMI システムで一元管理するために、システム内にユーザ環境情報管理部を設置する。ユーザ環境情報管理部が保持する情報は、他モジュールから 2 種類の方法で参照できるように設計する。一つはユーザ環境情報管理部から他モジュールへのイベント通知による情報参照で、ユーザ環境情報の動的な変化に対する即応型の処理を行なう際に用いる。もう一つは共通インタフェースを通じた他モジュールからの情報参照で、必要に応じて周辺環境を参照するような場合に用いる。本研究ではこれらの情報参照の方法を、筆者らのグループで策定を進めてきた MMI 記述言語 XISL 上で可能にするために、XISL の改良も同時に検討する。以上に述べたユーザ環境情報の利用により、ユーザや環境に合わせた対話戦略の切り替えや対話内容の変更が可能になるため、より自然な対話が可能である。本稿では、観光案内をタスクとした MMI アプリケーションの実装例も示す。

キーワード ユーザ環境情報、マルチモーダル対話、XISL、Dynamic Property Framework

A Framework to Handle User-Environment Information in a Multimodal Interaction System

Yuji Okuma Kouichi Katurada Tsuneo Nitta

Graduate School of Technology, Toyohashi University of Technology

1-1 Hibarigaoka, Tempaku-cho, Toyohashi 441-8580, JAPAN

E-mail: {okuma,katurada,nitta}@vox.tutkie.tut.ac.jp,

Abstract

This paper provides a framework to handle user-environment informations in a Multi-Modal Interaction (MMI) system. User-environment informations includes terminal information, environment information, user profile, and so on. For this purpose, we incorporate a user-environment information manager in the MMI system to centralize the control of the above informations. We prepared two ways to access the information. The first way is event notification to the other modules for handling dynamic change of information. The second way is common interface for referring to the information as necessary. We also modified the MMI description language XISL, which is specified by our group, for accessing the information. These improvements will realize more natural dialog than that in the previous MMI systems. This paper also demonstrates an example of MMI application of a tourist information system to show effectiveness of the improvements.

Keyword User-Environment Information, Multimodal Interaction, XISL, Dynamic Property Framework

1. はじめに

近年，マルチモーダル対話（MMI）に関する技術の標準化が活発に行われている．W3C のマルチモーダルワーキンググループ[1]では，MMI システムのフレームワーク，EMMA[2]等が標準として公表された．その他の組織においても，様々な MMI 記述言語[3][4][5]が提案されている．また，筆者らも MMI システムのアーキテクチャ[6]を検討するとともに，MMI 記述言語 XISL[7]を提案してきた．

このような状況の中，W3C において，Dynamic Properties Framework (DPF)[8]の検討が始まった．DPF は端末情報や，環境情報，ユーザプロフィール（年齢，性別，嗜好，・・・）（以上をまとめてユーザ環境情報と呼ぶ）を web ベースシステムで利用するためのフレームワークである．ユーザ環境情報を MMI システムで利用することにより，ユーザや環境に合わせた対話戦略の切り替え，対話内容の変更が可能になるため，より自然な対話の実現が期待できる．

そこで本報告では，W3C で検討されている DPF を参考に，ユーザ環境情報を取り扱うための枠組みを，筆者らがこれまで検討してきた MMI システムに導入する．まず，ユーザ環境情報を MMI システム内で一元管理するために，ユーザ環境情報管理部を MMI システム内に設ける．ユーザ環境情報管理部が保持する情報は，2 種類の方法で参照できるように設計する．一つはイベント通知による情報参照で，ユーザ環境情報の動的な変化に対する即応型の処理を行なう際に用いる．もう一つは共通インタフェースを通じた情報参照で，必要に応じて周辺環境を参照するような場合に用いる．本研究では，以上に述べた情報参照を MMI 記述言語 XISL から可能にするための改良も同時に検討する．

以下では，提案する MMI システムのアーキテクチャと MMI 記述言語 XISL について説明したあと，ユーザ環境情報管理部について説明する．続いて，MMI システムの内部処理の流れを説明する．

2. MMI システム

2.1 MMI システムのアーキテクチャ

筆者らは，図 1 に示す UMA 三階層モデル[9]に基づいて設計している．UMA 三階層モデルは，MMI システムの基本構成を与えるもので，

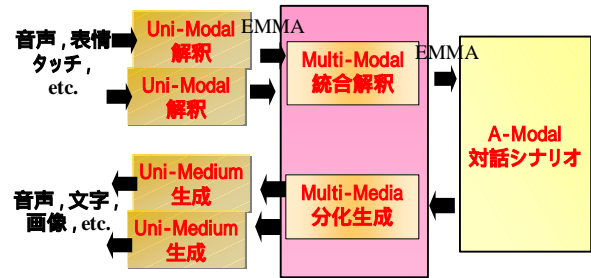


図 1 : UMA 三階層モデル

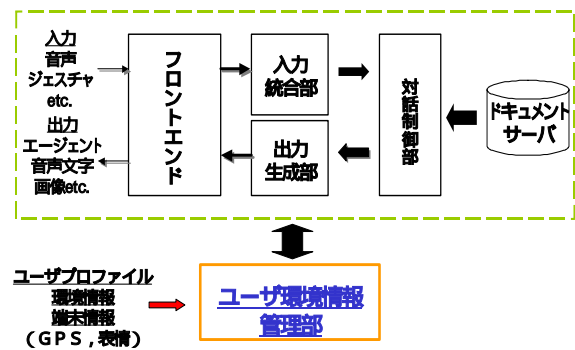


図 2 : MMI システムアーキテクチャ

Uni-modal/ Uni-medium 層，Multi-modal/ Multi-media 層，A-modal 層の三階層から成る．それぞれの階層では，個別のモダリティ/メディアの処理，複数のモダリティ/メディア情報を統合した処理，モダリティ非依存の処理が行なわれる．各層の詳細については文献[9]を参照されたい．

各層間のデータ通信には EMMA[2]を用いている．EMMA はマルチモーダル入力を表現するための統一記法で，個々のモダリティからの入力内容や，その解釈結果，それらの関係や信頼性などを表現できる．このため EMMA は，MMI システムの標準的データ送信仕様として最適なし様と考えられる．

以上に述べたモデルと通信仕様に基づき，我々は図 2（上部）に示す構成の MMI システムを構築した．モデル中の三階層はそれぞれフロントエンド部（音声認識・合成エンジン，ブラウザなどを用いて入出力を管理するモジュール），入力統合部/出力生成部（複数の入力モダリティを統合する/ユーザへの出力を生成するモジュール），対話制御部（各ドキュメントを解釈し，対話シナリオに沿って対話を制御するモジュール）として実装した．またこれらのモジュール間の通信は EMMA を用いている．本論文では，以上に説明した MMI システムに，ユーザ環境情報を扱うた

```

<?xml version="1.0" encoding="Shift-JIS"?>
<!DOCTYPE xisl SYSTEM "xisl.dtd">
<xisl version="2.0">
  <head> ... </head>
  <body>
    <form id="Shop">.....(1)
    <fe><!--html 文章--></fe>..... (2)
    <fe><!--音声認識文法--></fe>.....(3)
    <initial>
      <prompt>
        <fe><!--「商品と個数を入力して下さい」--></fe>
      </prompt>
    </initial>
    <field name="GOODS">.....(4)
    <prompt>
      <fe><!--「商品を入力してください」--></fe>
    </prompt>
    <filled>.....(5)
      <!--アクション -->
    </filled>
    </field>
    <field name="NUM">
      <prompt>
        <fe><!--「個数を入力してください」--></fe>
      </prompt>
      <filled>
        <backend action="get_num.cgi"
          namelist="GOODS" return="num"/>.....(6)
        <if cond="NUM>num">.....(7)
          <then><!--在庫がありません--> </then>
        </if>
      </filled>
    </field>
    <filled namelist="GOODS NUM">
      <goto next="next_dialog"/>.....(8)
    </filled>
  </form>
  <form id="CASH">
    :
  </form>
</body>
</xisl>

```

図3 . XISL2.0 の記述例

めのユーザ環境情報管理部を導入することを検討する。

2.2 MMI 記述言語 XISL

XISL (eXtensible Interaction Scenario Language) は, MMI システムのユーザがシステムと対話する際のシナリオを記述する言語で, 主にシステム内の対話制御部で処理される。最新バージョンである XISL2.0[10]では, VoiceXML を参考に, 従来の XISL1.1 を再設計した。前節で示した UMA 三階層モデルの導入により, 対話シナリオが A-modal 化され, この結果, 音声のみを扱う Uni-modal 言語 VoiceXML の基本構造と実行アルゴリズムを導入することが可能になった。

図3に XISL2.0 の記述例を示す。まず(1)の <form>を訪れると, フォームレベルにある(2), (3)の<fe>が実行される。<fe>は, 入力モダリティと出力メディアの動作を定義する。この例では詳細を省略しているが, html の表示と音声認識文法を有効にしている。続いて(4)の<field>が選択される。<field>は入力を待ち受けるためのスロットを表す。例えば図3の例において, ユーザ入力によって(4)の<field>が埋められた場合, (5)の<filled>が実行される。<filled>はユーザからの入力によって入力項目が埋められたあとにシステムが実行するアクションを記述する。<field>と<filled>は, VoiceXML を踏襲している。(6)の<backend>は, CGI などの外部プログラムを実行するためのタグである。(7)の<if>では条件分岐処理を行ない, 続く<then>で条件を満たしていた場合のアクションを実行する。別のフォーム項目へ対話遷移を行なう場合には(8)の<goto>を用いる。

3. ユーザ環境情報管理部の要求仕様

ユーザ環境情報管理部を設計するにあたり, まずユースケースを検討し, ユースケースに基づいて要求仕様を導き出した。以下, 主なユースケースと要求仕様を順に述べる。

3.1 デバイスの追加

MMI システムにおいて, 何らかのアプリケーションを実行中に, 入力デバイスとして新たにマイクを接続した場合を考える。このとき実行中のアプリケーションで音声入力を用いた対話を始めるには, マイクが追加されたというユーザ環境情報を取得・保持する必要がある。また, ユーザ環境情報の中には GPS 情報のように度々更新される情報もある。そこで要求仕様の一つとして, ユーザ環境情報のデータ管理に関する条件を以下のように定める。

(i) 新規情報の追加や, 情報の更新が容易なデータ構造を用いる。

3.2 オンラインショッピング

オンラインショッピングシステムにおいて商品を提示した際に, ユーザの表情に合わせて, 提示する商品を変更したり, 提示した商品をアピールしたり, といった対話切り替えをスムーズに行

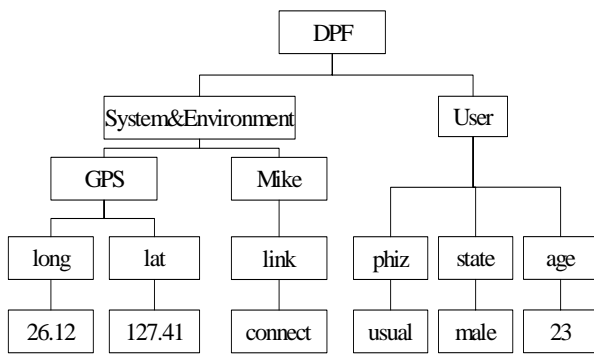


図4 . ユーザ環境情報の木構造の例

なえるなら，ユーザ-システム間の対話はより円滑に進むだろう．これを実現するには，対話進行を制御するモジュール側で，ユーザの表情の動的な変化を待ち受ける必要がある．そこで，次の項目をユーザ環境情報管理部の機能に関する要求仕様として定める．

- (ii) 情報の動的変化を MMI システム内の各モジュールに通知する機能を備える．

3.3 列車の予約

GPS 情報に基づいて列車の予約をするコンテンツを例にとると，ユーザが「ここから東京まで」と発話した際に「ここ」という発話を最依りの駅と捉えたいことがある．この実現には，ユーザ環境情報管理部からユーザの現在地を表す GPS 情報を取得し，地図データベースモジュールへその情報を伝える必要がある．また，こうした処理を実行するには，前節のケースと異なり，対話進行を制御するモジュールの側から，ユーザ環境情報管理部内の環境情報を参照/取得する必要がある．さらにアプリケーションを記述する XISL 上からも，情報参照できるようにすべきである．そこで次の二項目を要求仕様として定める．

- (iii) MMI システム内の各モジュールから，ユーザ環境情報の参照/取得を可能にする．
- (iv) MMI 記述言語 XISL から，ユーザ環境情報の参照/取得を可能にする．

4 . 要求仕様に基づく設計指針

前節で挙げた 4 つの要求仕様を基に，ユーザ環境情報管理部の設計指針を定めた．

4.1 木構造によるデータ管理

まず要求仕様(i)を満たすために，ユーザ環境情報を XML の木構造で管理するようユーザ環境情

```
<catch event="SE.chg_phiz">.....
<if cond="Session.SE.GetProperty("phiz")
eq 'baffule' && Session.SE.GetProperty("user:state")
eq 'not_move' ">.....
<then>
<fe><!--
音声合成[何かお困りでしょうか?]
--></fe>
<goto next="#search_landmark">
</then>
</if>
</catch>
```

図5 . イベント,<catch>の記述

報管理部を設計した．ユーザ環境情報は，図 4 に示すように，ルートからリーフに向けて，詳細な端末情報やユーザプロフィールに分化している．ツリーを構成する各ノードをプロパティと呼び，GPS やマイクなど子プロパティも機能や性質ごとに分化している．木構造のデータを操作する際には，一般的な XML 操作のための技術（XPath[10]や DOM[11]）を利用する．

4.2 他モジュールへのイベント通知機能

要求仕様(ii)のようにユーザ環境情報の動的変化を通知するには，イベントの利用が最適である．そこで，ユーザ環境情報管理部にイベント発生機能と，他モジュールへの通知機能を実装する．イベント通知は実行システムの全モジュールに対して行われ，特定のイベントの待ち受けやイベントの利用は，各モジュールの側で判断することとする．

4.3 他モジュールへのインタフェースの提供

要求仕様(iii)を満たすために，他のモジュールからユーザ環境情報への情報参照を可能にする共通インタフェースを提供する．インタフェースは 2 種類用意した．一つはツリーのノードを辿っていく DOM を利用する方法である．もう一つは，プログラミング言語の関数のように，予め定義した予約語（= プロパティ名．例えば User:age）を指定するインタフェースを用いる方法である．DOM を用いた情報取得には，例えば「GPS プロパティの子プロパティの中から，アクティブなプロパティの値を取得する」のような記述が必要になるため，ツリー構造を熟知することが必要である．一方，予約語を用いる場合には，DOM のように複雑なツリー構造を指定する必

要がないため、ツリー構造の知識なしに利用でき

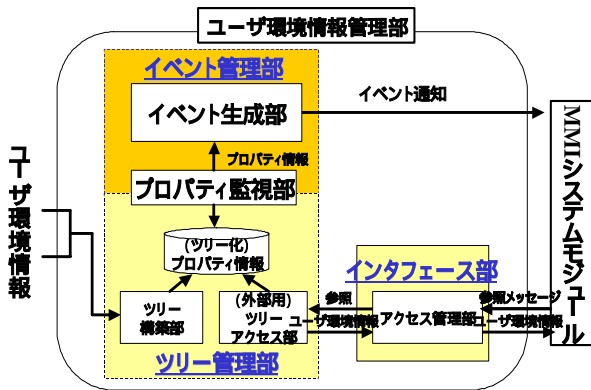


図6. ユーザ環境情報管理部の構成

る利点がある。

4.4 XISL からの利用方法

要求仕様(iv)に従い、XISL からの情報参照を可能にするため、4.2 節と 4.3 節で述べたそれぞれの参照方法が記述できるよう仕様を変更した。

まず、4.2 節のイベント通知による情報参照を実現する記法方法を説明する。XISL ではイベントを取得するための要素として、<catch>要素が用意されている。<catch>は、アプリケーション内の任意の要素で発生したイベントを取得し、イベントに応じた処理を実行できる。本システムでは、ユーザ環境情報管理部からのイベントを他のイベントと区別するため、図5の に示すように”SE”で始まる文字列でユーザ環境情報管理部からのイベントを表すことにした。

また、4.3 節の共通インタフェースを用いた参照を可能にするため、ユーザ環境情報を常時保持するセッション変数を用意し、XISL 内部での参照が発生する度に、ユーザ環境情報管理部に問い合わせる仕様とした。図5の では、セッション変数のスコープを表す“Session.”に、ユーザ環境情報管理部へのアクセスを表す記述“SE.インタフェース名(“予約語”)”を組み合わせることで参照を実現している。

5. ユーザ環境情報管理部の構成

以上に説明した設計指針に基づいて、図6に示すユーザ環境情報管理部を設計した。

5.1 ツリー管理部

ツリー管理部ではユーザ環境情報を収集・解析

し、新規プロパティの追加や情報の更新を行なう。内部モジュールの一つであるツリーアクセス部は、他のモジュールがユーザ環境情報を参照/取得する際に、参照メッセージの解析、プロパティツリーの探索、ユーザ環境情報の取得を行なう。もう一つの内部モジュールであるプロパティ監視部は、プロパティの動的な変化を監視しており、プロパティの内容が変化したときに、変化のあったプロパティの情報を取得し、イベント管理部へプロパティ情報を渡す。

5.2 イベント管理部

イベント管理部はイベント発生に関する処理全般を行なう。イベント生成部は、プロパティ監視部から渡されたプロパティ情報が、プロパティ毎にそれぞれ設定されたイベント生成条件を満たす場合に、イベントを生成して MMI システムの各モジュールへ通知する。イベント発生から通知までの流れは以下の通りである。

情報の動的変化が生じると、情報更新メッセージがユーザ環境情報管理部へ送信される。

ツリー管理部では更新メッセージを解析し、解析結果に従い、プロパティツリーの更新を行なう。

変化したプロパティ情報が、プロパティ監視部からイベント管理部へ渡される。

イベント管理部でイベント生成条件を判定し、イベントが生成・通知される。

5.3 インタフェース部

インタフェース部は、各モジュールからのユーザ環境情報を参照するための窓口である。アクセス管理部は、システムの各モジュールから同時アクセスを受け付けて処理するためにサーバの形をとっている。各モジュールからのメッセージ解析やユーザ環境情報取得は、ツリー管理部にあるツリーアクセス管理部が行なう。ユーザ環境情報の参照から取得に至るまでを以下に示す。

- A) 対話制御部、入力統合部などのモジュールから、ユーザ環境情報管理部に対して参照メッセージが送信される。
- B) ユーザ環境情報管理部のインタフェース部を介して、ツリー管理部において、参照メッセージの解析、プロパティツリーの探

```

<catch event="SE.chg_phiz">.....
<if cond="Session.SE.GetProperty("phiz")
eq 'baffule' && Session.SE.GetProperty("user:state")
eq 'not_move' ">.....
<then>
<backend action="search.cgi"
namelist="Session.SE.GetProperty(GPS:lat)"
return="current" />.....
<backend action="GUIDE.cgi"
namelist="Session.SE.GetProperty(GPS:lat)"
return="guide" />
<fe>
<!-- 音声合成[現在地は"current"です。
目的地へ向かうには, "guide"です。] -->.....
</fe>
</if>
</catch>

```

(a)

```

<form id="GUIDE">
<field name="Distinate">.....
<fe>
<!-- 音声合成[目的地はどこですか] -->
</fe>
</field>
<filled>
<if cond=" SessionSE.GetProperty("user:sex")
eq 'female' ">.....
<then>
<call next="female_guide" />.....
</then>
<else>
<call next="male_guide" />.....
</else>
</if>
</filled>
</form>

```

(b)

図 7 . ユーザ環境情報の利用例

素, プロパティ情報の取得が行われる .

- C) 参照メッセージを送信したモジュールへ, 取得したプロパティ情報が送信される .

6 . ユーザ環境情報の利用例

図 7 に示す XISL 文書は, 観光案内タスクの中で想定される対話の 1 シーンである . (a) の対話では, ユーザが道順を追えずに迷った様子 ("phiz"eq"baffule") であると判断した場合に, システム側からユーザに情報提供を行なうことを想定している . (a) の対話の流れについて簡単に説明する . まず の<catch>で, ユーザの表情変化イベントを待ち受ける . ユーザの表情変化を受けユーザ環境情報管理部が表情変化イベントを生成し, 対話制御部がこれを受け取ると の<if>以下が実行される . ここでは, セッション変数を用いてユーザ環境情報を参照し, 現在のユー

ザの表情とユーザの移動状態を取得する . 条件を満たすと, 以下が実行され, CGI から現在地と目的地までの案内を取得したのち, の<fe>が実行され, 案内のガイドが出力される .

(b) の対話では, ユーザの性別に従って目的地の紹介対話を変更している . (b) の対話の流れを簡単に説明する . の<field>でユーザから目的地の入力を受け付け, 入力項目が埋まると, の<if>でユーザの性別を取得するため, セッション変数によるユーザ環境情報の参照が実行される . ユーザの性別が判ると, <call>により または の対話へ遷移する .

7 . まとめ

ユーザ環境情報を MMI システムで利用するために, ユーザ環境情報管理部の導入を検討した . ユーザ環境情報の利用により, ユーザの表情やプロフィール, 環境変化などに対応した柔軟な対話構築が可能になることを示した . 今後は, 本システムの有効性を示すために実装作業を進めていきたい .

8 . 参考文献

- [1] <http://www.w3.org/2002/mmi/>
- [2] <http://www.w3.org/TR/emma/>
- [3] <http://www.saltforum.org/>
- [4] <http://www.voicexml.org/specs/multimodal/x+v/12/spec.html>
- [5] <http://www.scansoft.com/speechworks/xhmi/>
- [6] 桂田浩一, 他 " 多様な端末からのアクセスを可能にする MMI アーキテクチャ ", 情報処理学会研究報告 2002-SLP-40, pp.51-56(2002) .
- [7] 桂田浩一, 他 " MMI 記述言語 XISL の提案 ", 情報処理学会論文誌 Vol.44 , No.11 , pp.2681-2689 (2003-11) .
- [8] <http://www.w3.org/TR/DPF/>
- [9] 新田恒雄, " マルチモーダル対話の深化と記述言語の今後 ", 情報処理学会研究報告, 2004-SLP-50 , pp.15-22 , Feb.2004 .
- [10] 青木一峰, 他, " 異なる端末環境から利用可能な MMI アプリケーション開発における記述負担の軽減 ", 電子情報通信学会技術研究報告, SP2004-126 , pp.25-30 , 情報処理学会研究報告, 2004-SLP-54 , pp.271-276 , (2004-12) .
- [11] <http://www.w3.org/TR/xpath>
- [12] <http://www.w3.org/DOM/>