Proposal of a Hierarchical Architecture for Multimodal Interactive Systems

Masahiro Araki*1  Tsuneo Nitta*2  Kouichi Katsurada*2
Takuya Nishimoto*3  Tetsuo Amakasu*4
Shinnichi Kawamoto*5

*1Kyoto Institute of Technology
*2Toyohashi University of Technology
*3The University of Tokyo  *4NTT Cyber Space Labs.  *5ATR
Outline

• Background
  – Introduction of speech IF committee under ITSCJ
  – Introduction to Galatea toolkit

• Problems of W3C MMI Architecture
  – Modality Component is too large
  – Fragile Modality fusion and fission functionality
  – How to deal with user model?

• Our Proposal
  – Hierarchical MMI architecture
  – “Convention over Configuration” in various layers
Background(1)

• What is ITSCJ?
  – Information Technology Standards Commission of Japan
    • under IPSJ (Information Processing Society of Japan)

• Speech Interface Committee under ITSCJ
  – Mission
    • Publish TS (Trial Standard) document concerning multimodal dialogue systems
Background(2)

- Theme of the committee
  - Architecture of MMI system
  - Requirements of each component

- Future directions
  - Guideline for implementing practical MMI system
  - specify markup language
Our Aim

1. Propose an MMI architecture which can be used for advanced MMI research
   
   W3C: From the practical point of view (mobile, accessibility)

2. Examine the validity of the architecture through system implementation
   
   Galatea Toolkit

3. Develop a framework and release it as a open source
   
   towards de facto standard
Galatea Toolkit(1)

- Platform for developing MMI systems
  - Speech recognition
  - Speech Synthesis
  - Face Image Synthesis
Galatea Toolkit(2)
Galatea Toolkit(3)

- Phoenix
  - Dialogue Manager
  - Macro Control Layer (AM-MCL)
    - Direct Control Layer (AM-DCL)
      - ASR Julian
      - TTS Galatea talk
      - Face FSM

2007/11/16 W3C MMI ws
Problems of W3C MMI(1)

- The “size” of Modality Component does not suit for life-like agent control

![Diagram of Runtime Framework and Modality Components]

- Delivery Context Component
- Interaction manager
- Data Component

- Modality Component API
  - Speech Modality
    - ASR
    - TTS
  - Face Image Modality
    - FSM

2007/11/16
Problems of W3C MMI(1)

• Lip synchronization with speech output

![Diagram of MM/MI components and interactions](image)
Problems of W3C MMI(1)

- Back channeling mechanism

![Diagram of W3C MMI](image-url)
Problems of W3C MMI(2)

• Fragile Modality fusion and fission functionality

- Delivery Context Component
- Interaction manager
- Data Component

How to define "from here to there"?

Speach Modality
- ASR

Tactile Modality
- touch sensor

Runtime Framework

point (120,139)
point (200,300)

How to define multimodal grammar?
Is simple unification enough?
Problems of W3C MMI(2)

• Fragile Modality fusion and fission functionality

Delivery Context Component
Interactive manager
Data Component

Runtime Framework

"this is route map"

Speech Modality

TTS

SVG

Graphic Modality

SVG Viewer

Contents planning is suitable for adapting various devices.
Problems of W3C MMI(3)

• How to deal with user model?

Delivery Context Component  | Interaction manager  | Data Component

Runtime Framework

Where is the user model information stored?

Speech Modality
- ASR
- TTS

Face Image Modality
- FSM

fails many times
Solution

• Back to multimodal framework
  – more smaller modality component
• Separate state transition description
  – task flow
  – interaction flow
  – modality fusion/fission

 hierarchical architecture
Investigation procedure
Phase 1

- Use case analysis
- Requirement for overall systems
- Working draft for MMI architecture
## Use case analysis

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th>input modality</th>
<th>output modality</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>on-line shopping</td>
<td>mouse, speech</td>
<td>display, speech animated agent</td>
</tr>
<tr>
<td>b</td>
<td>voice search</td>
<td>mouse, speech</td>
<td>display, speech</td>
</tr>
<tr>
<td>c</td>
<td>site search</td>
<td>mouse, speech, key</td>
<td>display, speech</td>
</tr>
<tr>
<td>d</td>
<td>interaction with robot</td>
<td>speech, image, sensor</td>
<td>speech, display</td>
</tr>
<tr>
<td>e</td>
<td>negotiation with interactive agent</td>
<td>speech</td>
<td>speech, face image</td>
</tr>
<tr>
<td>f</td>
<td>kiosk terminal</td>
<td>touch, speech</td>
<td>speech, display</td>
</tr>
</tbody>
</table>
Example of use case
Interaction with robot

*Nishijin Kasuri* is a traditional texture in Kyoto.

What is *Kasuri*?
Requirements

1. general
2. input modality
3. output modality
4. architecture, integration and synchronization point
5. runtimes and deployments
6. dialogue management
7. handling of forms and fields
8. connection with outside application
9. user model and environment information
10. from the viewpoint of developer

in common with W3C
extension
Investigation procedure

Phase 2

Detailed analysis of use case

Requirements for each layer

Publish trial standard

release reference implementation
Detailed use case analysis

1. Get input and start the task (4th floor)
2. Input integration (3rd floor)
3. Output decomposition (3rd floor)
4. Sound input (2nd floor)
5. Sound output (2nd floor)

1.1: Get input and start the task
1.1.1: Get input and sound input
1.2: User information acquisition (age, gender, etc.)
1.2.3: Web page display (product list page)

2.1: Get input
2.1-A: Normal input waiting
2.1-B: Sound input waiting
2.1-B: 'Keep asking'

T-shirt

2007/11/16 W3C MMI ws
Requirements of each layer

- Clarify Input/Output with adjacent layers
- Define events
- Clarify inner layer processing
- Investigate markup language
1st layer: Input/Output module

- Function
  - Uni-modal recognition/synthesis module

- Input module
  - Input: (from outside) signal
    - (from 2nd layer) information used for recognition
  - Output: (to 2nd) recognition result
  - Example: ASR, touch input, face detection, ...

- Output module
  - Input: (from 2nd) output contents
  - Output: (to outside) signal
  - Example: TTS, Face image synthesizer, Web browser, ...
2\textsuperscript{nd} : Modality component

- **Function**
  - Lapper that absorbs the difference of 1\textsuperscript{st} layer
  - Ex) Speech Recognition component
    - Grammar: SRGS  
    - Semantic analysis: SISR  
    - Result: EMMA
  - Provide multimodal synchronization
    - Ex) TTS with lip synchronization

\begin{center}
\begin{tikzpicture}
  \node [draw, text width=2cm, align=center] (A) at (0,0) {LS-TTS};
  \node [draw, text width=1.5cm, align=center] (B) at (-1.5,-1.5) {TTS};
  \node [draw, text width=1.5cm, align=center] (C) at (1.5,-1.5) {FSM};
  \draw [->] (A) -- (B);
  \draw [->] (A) -- (C);
\end{tikzpicture}
\end{center}
3rd: Modality Fusion

- Integration of input information
  - Interpretation of sequential / simultaneous input
  - Output the integrated result as EMMA format
3rd : Modality Fission

• Rendering output information
  – Synchronization of sequential/simultaneous output
  – Coordination of output modality based on the access device

I recommend “sushi dai”.

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sushi</td>
<td>3800</td>
<td>good taste</td>
</tr>
<tr>
<td>okame</td>
<td>3650</td>
<td>good service</td>
</tr>
<tr>
<td>iwasa</td>
<td>3500</td>
<td>shelfish</td>
</tr>
</tbody>
</table>
4th: Inner task control

- Image
  - a piece of dialogue at client side

S: Please input member ID
U: 2024

S: Please select food.
U: Meat

S: Is it OK?
U: Yes.
4th : Inner task control

- Required functions
  - Error handling
    ex) check departure time < arrival time
  - Default subdialogue
    ex) confirmation, retry, ...
  - Form filling algorithm
    ex) Form Interpretation Algorithm
  - Slot update information
    ex) process of negative response to confirmation request ("NO, from Kyoto.")
4th : Inner task control

5th

control

Initialize event
start dialogue(uri or code)
data
end event(status)

4th

• FIA
• Input analysis (with error check)
• Update data module
• Update user model

Initialize event
output contents

device information
EMMA

Initialize event
Start Input
(with interruption)

device information

3rd

Modality Fusion

Modality Fission

device information
end event(status)
5th: Task control

- Image
  - describe overall task flow
  - server side controller

- Possible markup languages
  - SCXML
  - Controller definition in MVC model
    - entry points and their processing
  - Script language on Rails application framework
    - contains application logic (6th layer)
    - easy to prototype and customize
5th: Task control

- state transition
- conditional branch
- event handling
- subdialogue management

Initialize event
start dialogue (uri or code)

4th: control

2007/11/16 32W3C MMI ws
6th: Application

- **Image**
  - Processing module outside of dialogue system
    - accessed from various layers

- **modules**
  - application logic
    - ex) DB access, Web API access
    - Persist, update, delete, search of data
  - user model / device model
    - persist user’s information through sessions
    - manage device information defined in ontology
Too many markup language?

• Does each level require different markup language?
  – No.
  – simple functionality of 5th and 4th layer can provide data model approach (ex) Ruby on Rails
  – default function of 3rd layer can be realized simple principle (ex) unification in modality fusion
  – 2nd layer functions are task/domain independent

“Convention over Configuration”
Summary

• Problems of W3C MMI Architecture
  – Modality Component
  – Modality fusion and fission functionality
  – User model

• Our Proposal
  – Hierarchical MMI architecture
  – “Convention over Configuration” in various layers