COMP349
Spoken Language Dialog System
The W3C Speech Interface Framework

Rolf Schwitter
schwitt@ics.mq.edu.au
Today’s Program

• The Voice Browser Working Group
• The W3C Speech Interface Framework
  – VoiceXML
  – Speech Synthesis Markup Language
  – Speech Recognition Grammar Specification
  – Semantic Interpretation for Speech Recognition
  – Call Control eXtensible Markup Language
  – [Extensible Multimodal Markup Annotation]
The Voice Browser Working Group

• The Voice Browser Working Group (VBWG)
  – focuses on speech-enabled Web applications
  – develops recommendations for voice browsers
  – develops specifications for markup languages.
• The framework of markup languages for speech processing systems is known as W3C Speech Interface Framework.
• See: http://www.w3.org/Voice/
Speech-enabled Web Applications

• Statements from VBWG:

Some possible applications include:

• Accessing business information, including the corporate “front desk” asking callers who or what they want, automated telephone ordering services, support desks, order tracking, airline arrival and departure information, cinema and theater booking services, and home banking services.

• Accessing public information, including community information such as weather, traffic conditions, school closures, directions and events; local, national and international news; national and international stock market information; and business and e-commerce transactions.

• Accessing personal information, including calendars, address and telephone lists, to-do lists, shopping lists, and calorie counters.

• Assisting the user to communicate with other people via sending and receiving voice-mail and email messages.
Components of the Speech Interface Framework
A Speech Application Fragment

...<form id = "travel">
   <field name = "destination">
      <prompt>
         Do you want to fly to
         <emphasis level = "strong"> New York </emphasis> or to <emphasis level = "strong"> Washington </emphasis>
      </prompt>
      <grammar mode = "voice" root = "destination-city">
         <rule id = "destination-city">
            <one-of>
               <item tag = "NEW-YORK"> New York </item>
               <item tag = "NEW-YORK"> Big Apple </item>
               <item tag = "WASHINGTON"> Washington </item>
               <item tag = "WASHINGTON"> The Capital </item>
            </one-of>
         </rule>
      </grammar>
   </field>
</form>
VoiceXML

- VoiceXML (Voice Extensible Markup Language)
  - is a dialog markup language for specifying conversational dialogs
  - is based upon extensive industry experience
  - was contributed to W3C by members of the VoiceXML forum.
- See:
  - VoiceXML 2.0: http://www.w3.org/TR/voicexml20/
  - VoiceXML 2.1: http://www.w3.org/TR/voicexml21/
  - VoiceXML Forum: http://www.voicexml.org
History of VoiceXML

- 1995: Project at AT&T led to the PhoneMarkup Language (PML).
- 1998: AT&T and Lucent had variants of PML,
  Motorola had VoxML,
  IBM had Speech ML,
  HP had TalkML, and
  PipeBeach had VoiceHTML.
- 1999: VoiceXML Forum (AT&T, IBM, Lucent, Motorola) produced VoiceXML 0.9.
History of VoiceXML

- 2000: VoiceXML Forum submitted the spec to W3C.
- 2002: VoiceXML 2.0 was released by the W3C.
- 2004: VoiceXML 2.0 is a W3C recommendation.
- 2007: VoiceXML 2.1 is a W3C recommendation.
- VoiceXML 3.0 First Public Working Draft is expected late 2007.
VoiceXML: Example 1

```xml
<?xml version="1.0" encoding="UTF-8"?>
<vxml version="2.0" xmlns="http://www.w3.org/2001/vxml">
<form>
  <block> Hello World! </block>
</form>
</vxml>
```
<?xml version="1.0" encoding="UTF-8"?>
<vxml version = "2.0" xmlns = "http://www.w3.org/2001/vxml">

<var name = "hi" expr = "'Hello World!'"/>

<form id = "say_hi">
    <block>
        <prompt> <value expr = "hi"/> </prompt>
        <goto next = "#say_goodbye"/>
    </block>
</form>

<form id = "say_goodbye">
    <block>
        Goodbye!
    </block>
</form>

</vxml>
<?xml version="1.0" encoding="UTF-8"?>
<vxml version="2.0" xmlns="http://www.w3.org/2001/vxml">
  <form>
    <field name="destination">
      <prompt>
        Would you like to fly to New York or Boston?
      </prompt>
      <grammar src="destination.grxml"
        type="application/srgs+xml"/>
    </field>
    <block>
      <submit next="http://www..../destination.py"/>
    </block>
  </form>
</vxml>
SSML

• The Speech Synthesis Markup Language (SSML)
  – describes how text is presented as audio to the user
  – improves the quality of synthesized content
  – offers a standard way to control aspects of speech.

• See: http://www.w3.org/TR/speech-synthesis/
Aspects of Speech

- Aspects of speech
  - pronunciation,
  - word emphasis,
  - prosody,
  - speed,
  - volume,
  - pitch and other voice characteristics.
SSML for VoiceXML 2.0

• Examples of SSML markup in VoiceXML:
  
  `<audio>` specifies audio files to be played and spoken
  `<break>` specifies a pause in the speech output
  `<emphasis>` specifies that text should be spoken with emphasis
  `<say-as>` specifies the type of text construct within the element.
SSML: Examples

• <prompt> Do you want to fly to
  <emphasis level = "strong">
    New York
  </emphasis>
  or to
  <emphasis level = "strong">
    Washington
  </emphasis>
</prompt>

• <prompt> Press 1 or wait for the tone.
  <break time = "3s"/>
</prompt>
SSML: Examples

<prompt> Welcome to the Bird Seed Emporium.
<audio src = "http://www.birdsounds.com/thrush.wav"/></prompt>

We have 250 kilogram drums of thistle seed for
<say-as interpret-as = "currency"> $299.95 </say-as>
plus shipping and handling this month.

<audio src = "http://www.birdsounds.com/mourningdove.wav"/></prompt>
• The Speech Recognition Grammar Specification (SRGS) specifies the words and phrases which a user may speak in response to a prompt.

• The syntax of the grammar format can be presented in two forms:
  – Augmented BNF Form
  – XML Form.

• The specification makes the two representations mappable to allow automatic transformations between the two forms.

• See: http://www.w3.org/TR/speech-grammar/
SRGS: Example

ABNF Form

$destination-city =
  to London {"LONDON"} |
  to Paris {"PARIS"}

XML Form

<rule id = "destination-city">
  <one-of>
    <item tag = "LONDON">
      to London
    </item>
    <item tag = "PARIS">
      to Paris
    </item>
  </one-of>
</rule>
The Semantic Interpretation for Speech Recognition (SISR) spec describes the tags (annotations) that can be added to speech recognition grammars for extracting the semantic results from recognition.

The annotations are expressed in a syntax based on ECMAScript.

Provides a solution for alternative utterances with the same meaning:
- "Big Apple" translates into "NEW-YORK"
- "The Capital" translates into "WASHINGTON".

Check: http://www.w3.org/TR/semantic-interpretation/
SISR: Example

<grammar mode = "voice" root = "destination-city">
  <rule id = "destination-city">
    <one-of>
      <item tag = "NEW-YORK"> New York </item>
      <item tag = "NEW-YORK"> Big Apple </item>
      <item tag = "WASHINGTON"> Washington </item>
      <item tag = "WASHINGTON"> The Capital </item>
    </one-of>
  </rule>
</grammar>
CCXML

- The Call Control Extensible Markup Language (CCXML)
  - provides call control for VoiceXML telephony platforms
  - allows for controlling how phone calls are placed, answered, transferred, conferenced.
- Traditionally, call control has required interaction with the telephony API's which often change from one platform to another.
• Basically, CCXML provides support for VoiceXML to move calls around and connect them to dialog resources.
• For example, being able to conditionally answer a call is one of the features that CCXML brings to VoiceXML applications.
• See: http://www.w3.org/TR/2007/WD-ccxml-20070119/
<xml version="1.0" encoding="UTF-8"?>
<ccxml version="1.0">
  <eventhandler>
    <!-- Lets handle the incoming call -->
    <transition event="connection.CONNECTION_ALERTING" name="evt">
      <log expr="'The caller ID is ' + evt.callerid + '.'"/>
      <if cond="evt.callerid == '8315551234'">
        <accept/>
      </if>
      <else/>
      <reject/>
    </if>
  </transition>

  <!-- Lets handle the call being answered -->
  <transition event="connection.CONNECTION_CONNECTED">
  </transition>
</eventhandler>
</ccxml>
Multimodality

- Multimodal systems process two or more combined user inputs:
  - speech, pen, touch, gaze, head and body movement.
- They incorporate one or more recognition technology.
- Example: “Zoom in here”.
- They have the potential to function more robust.
- Main research focuses on:
  - speech and pen input
  - speech and lip movement
EMMA

• Extensible Multimodal Markup Annotation (EMMA) intended use is to represent the semantics for information entered via various input modalities and for the resulting integrated information.
• Each of the modality-specific recognizers/interpreters convert the user supplied information into an EMMA representation.
• The EMMA representation is then processed by a dialog manager.
• See: http://www.w3.org/TR/emma/
Figure 1: W3C Multimodal Framework
EMMA: Example

• The speech utterance "Zoom in here" is represented as

```xml
<emma:interpretation>
  <command>
    <zoom><location/></zoom>
  </command>
</emma:interpretation>
```
EMMA: Example

- The area circled by a pen is represented as a series of points:

  <emma:interpretation>
    <area>
      (200,200), (200,400), (400,400), (400,200)
    </area>
  </emma:interpretation>
EMMA: Example

- The integration of information from both the speech and pen modalities is represented as:

```xml
<emma:interpretation>
  <command>
    <zoom>
      <area>
        (200,200), (200,400), (400,400), (400,200)
      </area>
    </zoom>
  </command>
</emma:interpretation>
```
Take-Home Messages

• The VBWG works on a suite of markup languages for speech applications known as the W3C Speech Interface Framework.

• The most important markup languages are:
  – VoiceXML
  – Speech Synthesis Markup Language
  – Speech Recognition Grammar Specification
  – Semantic Interpretation for Speech Recognition
  – Call Control XML
  – [Extensible Multimodal Markup Annotation].