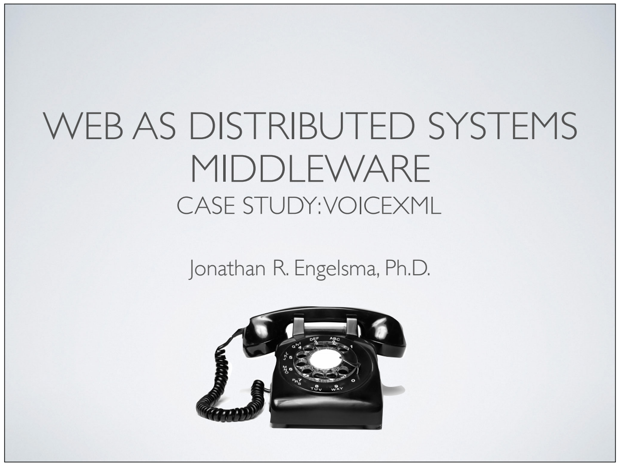


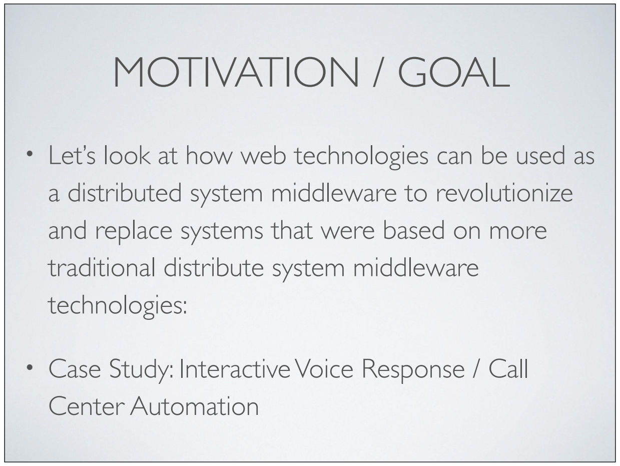
WEB AS DISTRIBUTED SYSTEMS MIDDLEWARE

CASE STUDY: VOICEXML

Jonathan R. Engelsma, Ph.D.

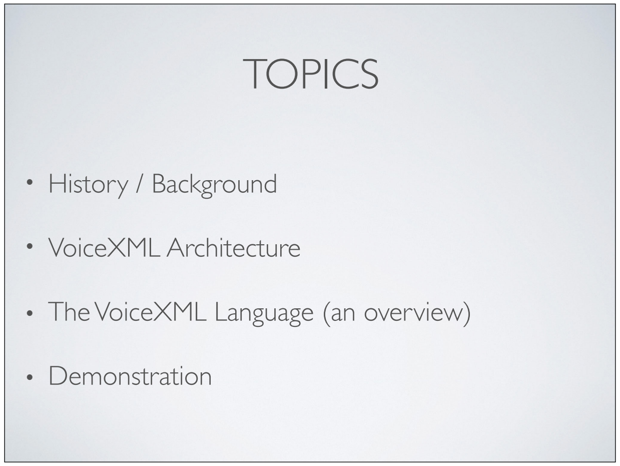
MOTIVATION / GOAL

- So far we have:
  - seen how the web gave the revolutionized content dissemination/creation on the Internet (e.g. the conventional world-wide web)
  - seen how web technologies can give our programs the same benefits it gives humans (e.g. web APIs)



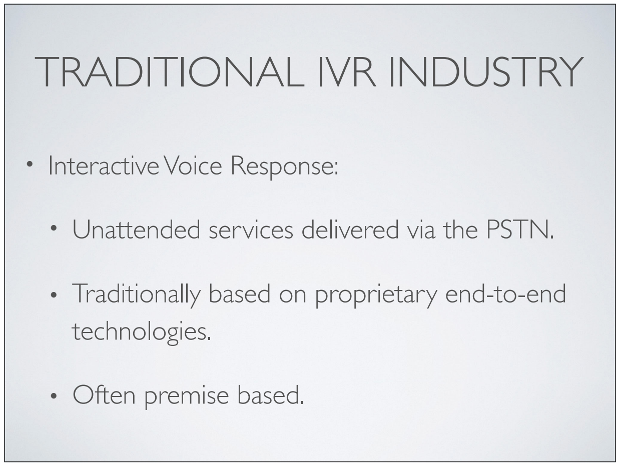
MOTIVATION / GOAL

- Let's look at how web technologies can be used as a distributed system middleware to revolutionize and replace systems that were based on more traditional distribute system middleware technologies:
- Case Study: Interactive Voice Response / Call Center Automation



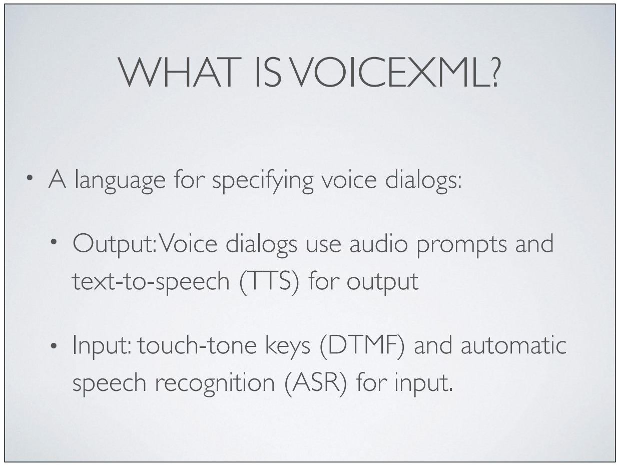
TOPICS

- History / Background
- VoiceXML Architecture
- The VoiceXML Language (an overview)
- Demonstration



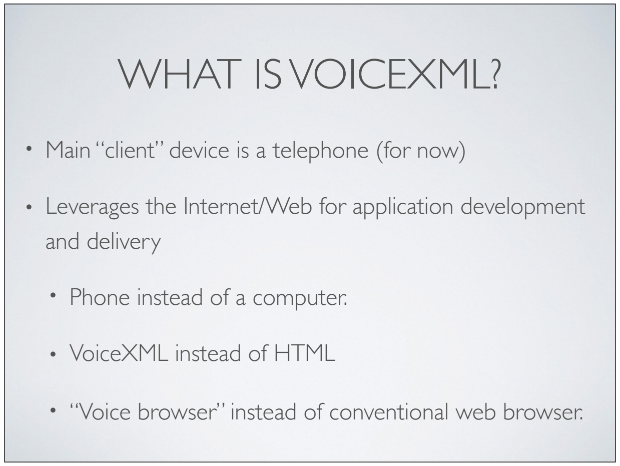
TRADITIONAL IVR INDUSTRY

- Interactive Voice Response:
  - Unattended services delivered via the PSTN.
  - Traditionally based on proprietary end-to-end technologies.
  - Often premise based.



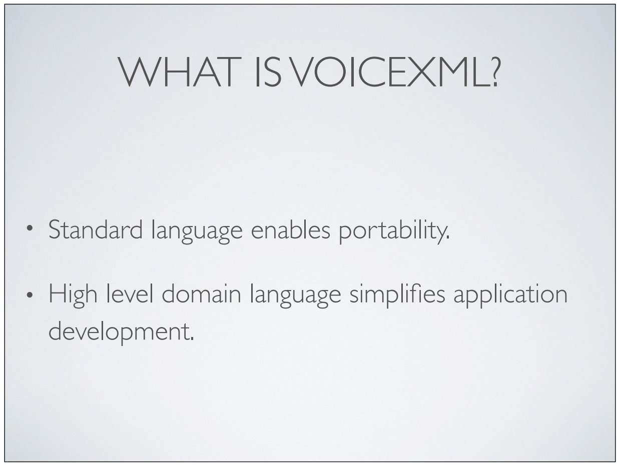
WHAT IS VOICEXML?

- A language for specifying voice dialogs:
  - Output: Voice dialogs use audio prompts and text-to-speech (TTS) for output
  - Input: touch-tone keys (DTMF) and automatic speech recognition (ASR) for input.



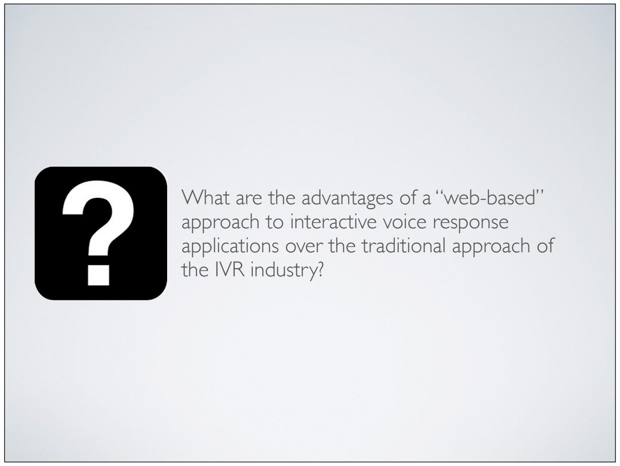
WHAT IS VOICEXML?

- Main "client" device is a telephone (for now)
- Leverages the Internet/Web for application development and delivery
  - Phone instead of a computer.
  - VoiceXML instead of HTML
  - "Voice browser" instead of conventional web browser.



WHAT IS VOICEXML?

- Standard language enables portability.
- High level domain language simplifies application development.



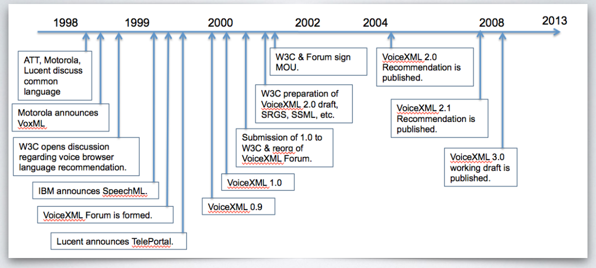
?

What are the advantages of a "web-based" approach to interactive voice response applications over the traditional approach of the IVR industry?

ADVANTAGES

- Leverage existing web application development tools.
- Leverage existing web infrastructure for application delivery.
- A high-level, domain-specific language greatly simplifies programming.
- Consolidate voice and web applications.
- Open up telephony platform to third party applications.

VOICEXML HISTORY



1998

1999

2000

2002

2004

2008

2013

ATT, Motorola, Lucent discuss common language

Motorola announces VoxML

W3C opens discussion regarding voice browser language recommendation.

IBM announces SpeechML.

VoiceXML Forum is formed.

Lucent announces TelePortal.

VoiceXML 0.9

VoiceXML 1.0

Submission of 1.0 to W3C & reorg of VoiceXML Forum.

W3C preparation of VoiceXML 2.0 draft, SRGS, SSML, etc.

W3C & Forum sign MOU.

VoiceXML 2.0 Recommendation is published.

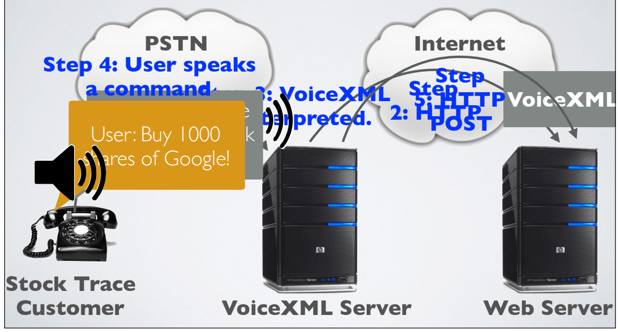
VoiceXML 2.1 Recommendation is published.

VoiceXML 3.0 working draft is published.

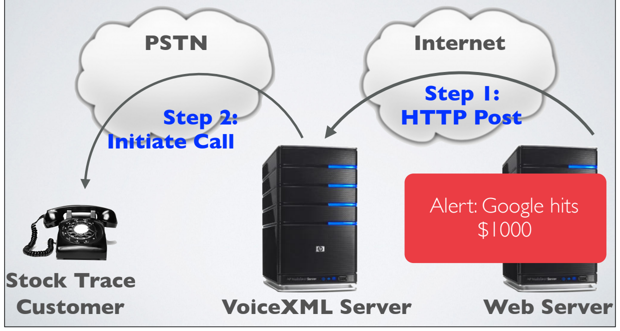
VOICEXML ARCHITECTURE

- Key goals:
  - Establish a Standard/Common high-level language
  - Leverage open, standard, known technology
  - Separate service logic from underlying telephony hardware

ARCHITECTURE - USER INITIATED CALL



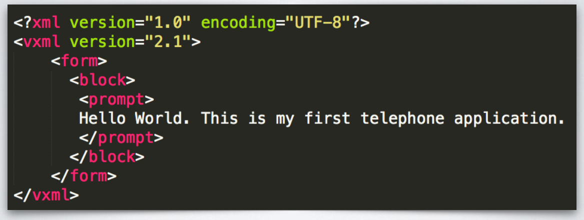
ARCHITECTURE - NETWORK INITIATED CALL



THE VOICEXML LANGUAGE

- an XML application
- Standardized by the W3C (along with a family of related specifications - Voice Browser Working Group)
- Utilizes ECMAScript (Javascript)
- Supports directed and mixed-initiative dialogs

HELLO WORLD!



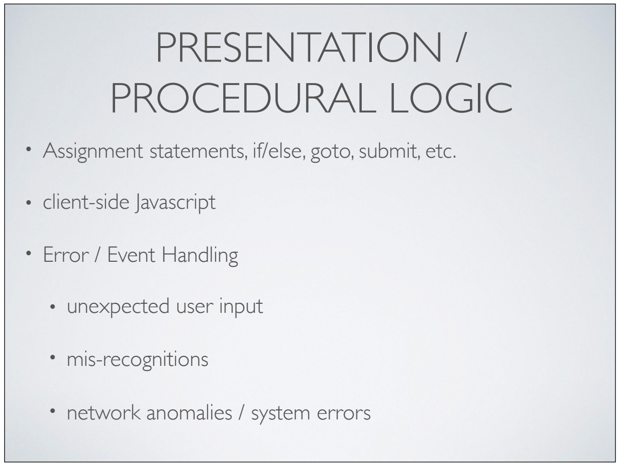
```
<?xml version="1.0" encoding="UTF-8"?>
<vxml version="2.1">
    <form>
      <block>
       <prompt>
       Hello World. This is my first telephone application.
       </prompt>
      </block>
    </form>
</vxml>
```

HUMAN-MACHINE INTERACTION

- Audio Output
  - text-to-speech (TTS)
  - pre-recorded audio
- Audio Input
  - speech recognition (ASR)
  - audio recording

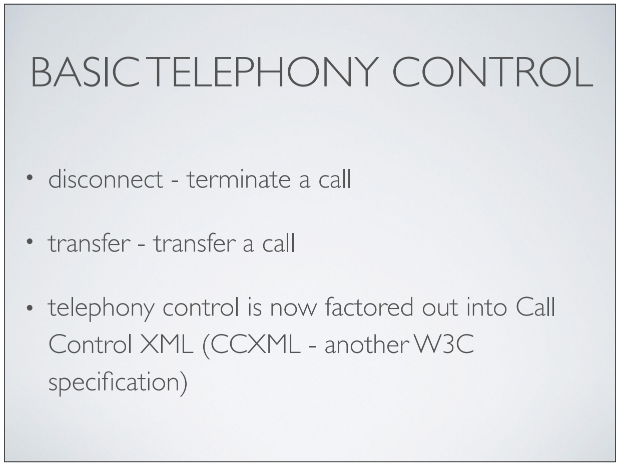
HUMAN-MACHINE INTERACTION

- Character Input
  - TouchTone™ (DTMF)
- Presentation Logic
  - client-side scripting in ECMAScript (Javascript)



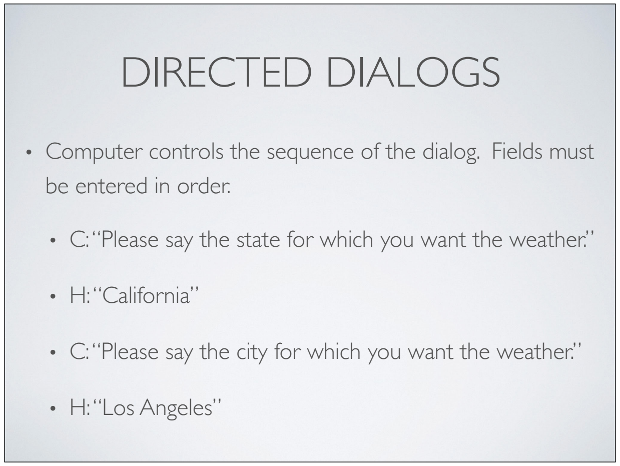
PRESENTATION / PROCEDURAL LOGIC

- Assignment statements, if/else, goto, submit, etc.
- client-side Javascript
- Error / Event Handling
  - unexpected user input
  - mis-recognitions
  - network anomalies / system errors



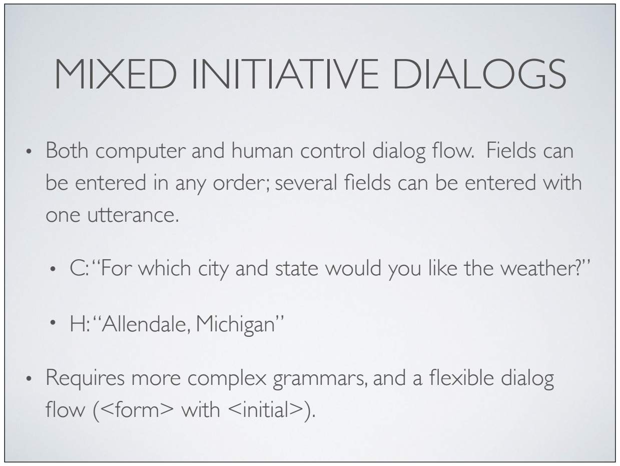
BASIC TELEPHONY CONTROL

- disconnect - terminate a call
- transfer - transfer a call
- telephony control is now factored out into Call Control XML (CCXML - another W3C specification)



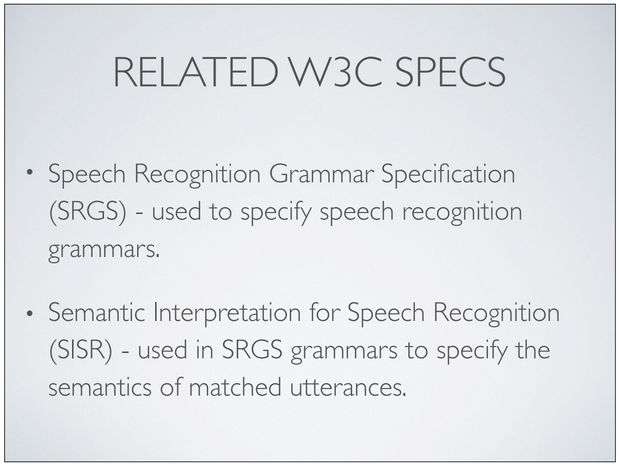
DIRECTED DIALOGS

- Computer controls the sequence of the dialog. Fields must be entered in order.
  - C: "Please say the state for which you want the weather."
  - H: "California"
  - C: "Please say the city for which you want the weather."
  - H: "Los Angeles"



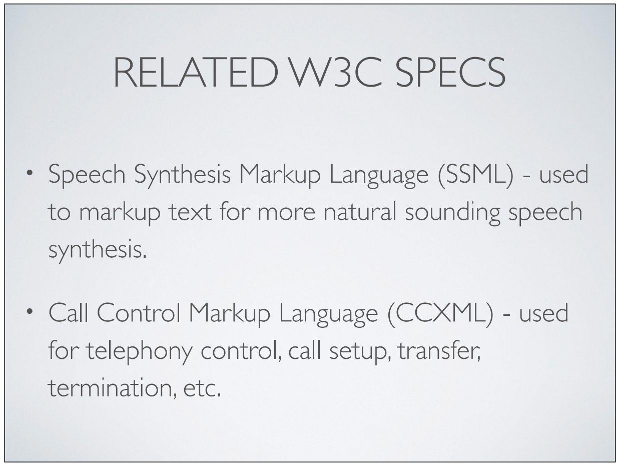
MIXED INITIATIVE DIALOGS

- Both computer and human control dialog flow. Fields can be entered in any order; several fields can be entered with one utterance.
  - C: "For which city and state would you like the weather?"
  - H: "Allendale, Michigan"
- Requires more complex grammars, and a flexible dialog flow (<form> with <initial>).



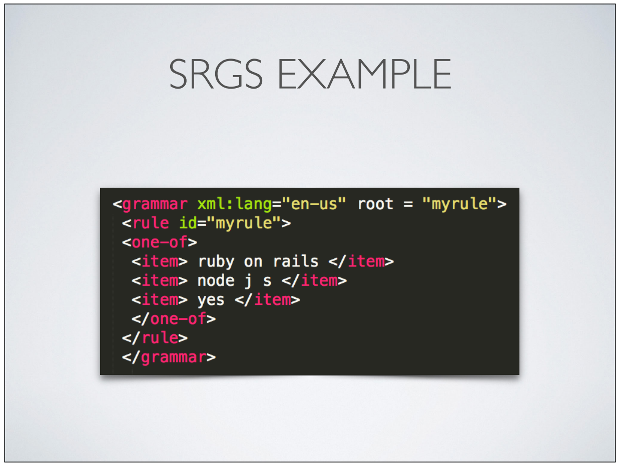
RELATED W3C SPECS

- Speech Recognition Grammar Specification (SRGS) - used to specify speech recognition grammars.
- Semantic Interpretation for Speech Recognition (SISR) - used in SRGS grammars to specify the semantics of matched utterances.



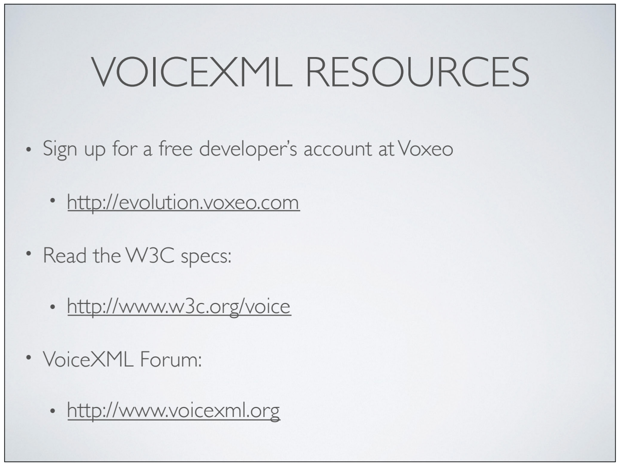
RELATED W3C SPECS

- Speech Synthesis Markup Language (SSML) - used to markup text for more natural sounding speech synthesis.
- Call Control Markup Language (CCXML) - used for telephony control, call setup, transfer, termination, etc.



SRGS EXAMPLE

```
<grammar xml:lang="en-us" root = "myrule">
 <rule id="myrule">
 <one-of>
  <item> ruby on rails </item>
  <item> node j s </item>
  <item> yes </item>
  </one-of>
 </rule>
 </grammar>
```



VOICEXML RESOURCES

- Sign up for a free developer's account at Voxeo
  - http://evolution.voxeo.com
- Read the W3C specs:
  - http://www.w3c.org/voice
- VoiceXML Forum:
  - http://www.voicexml.org

DEMO