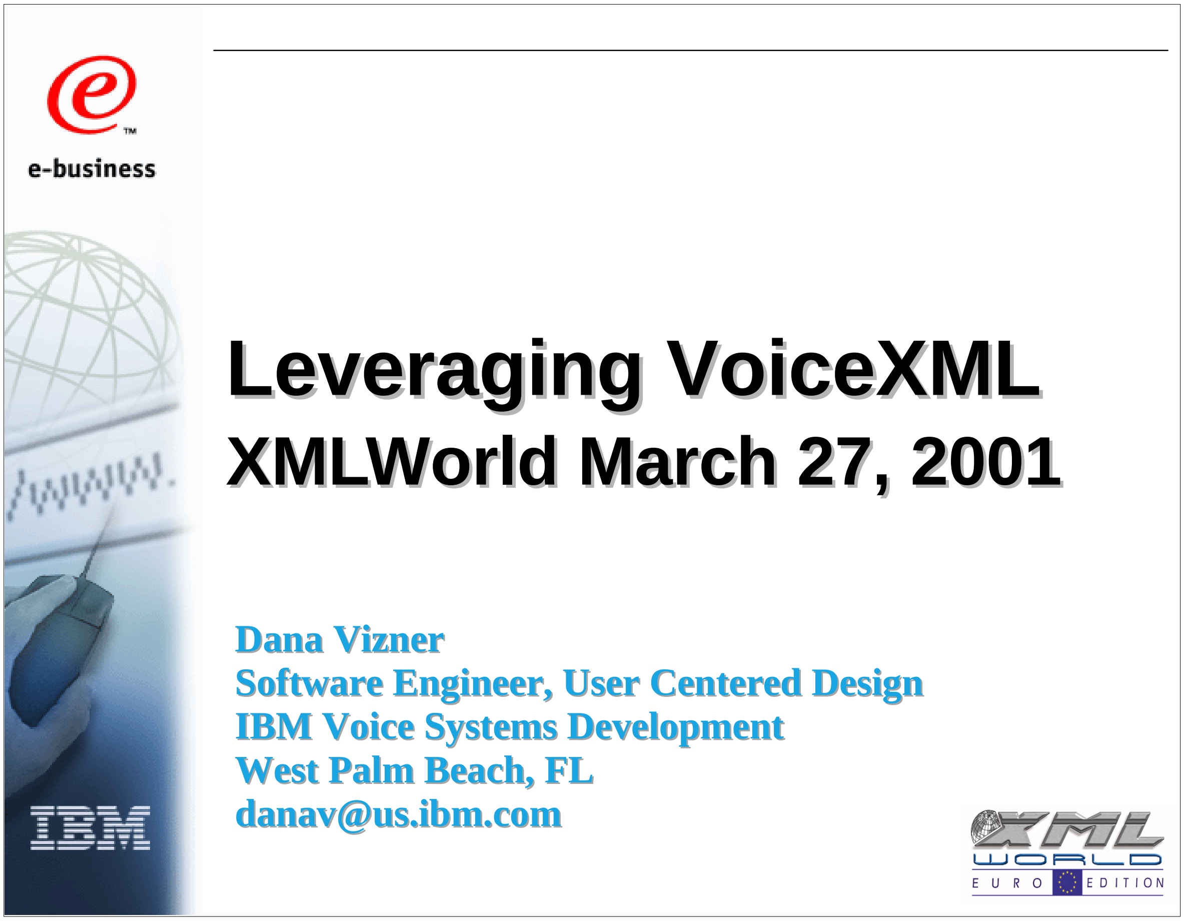


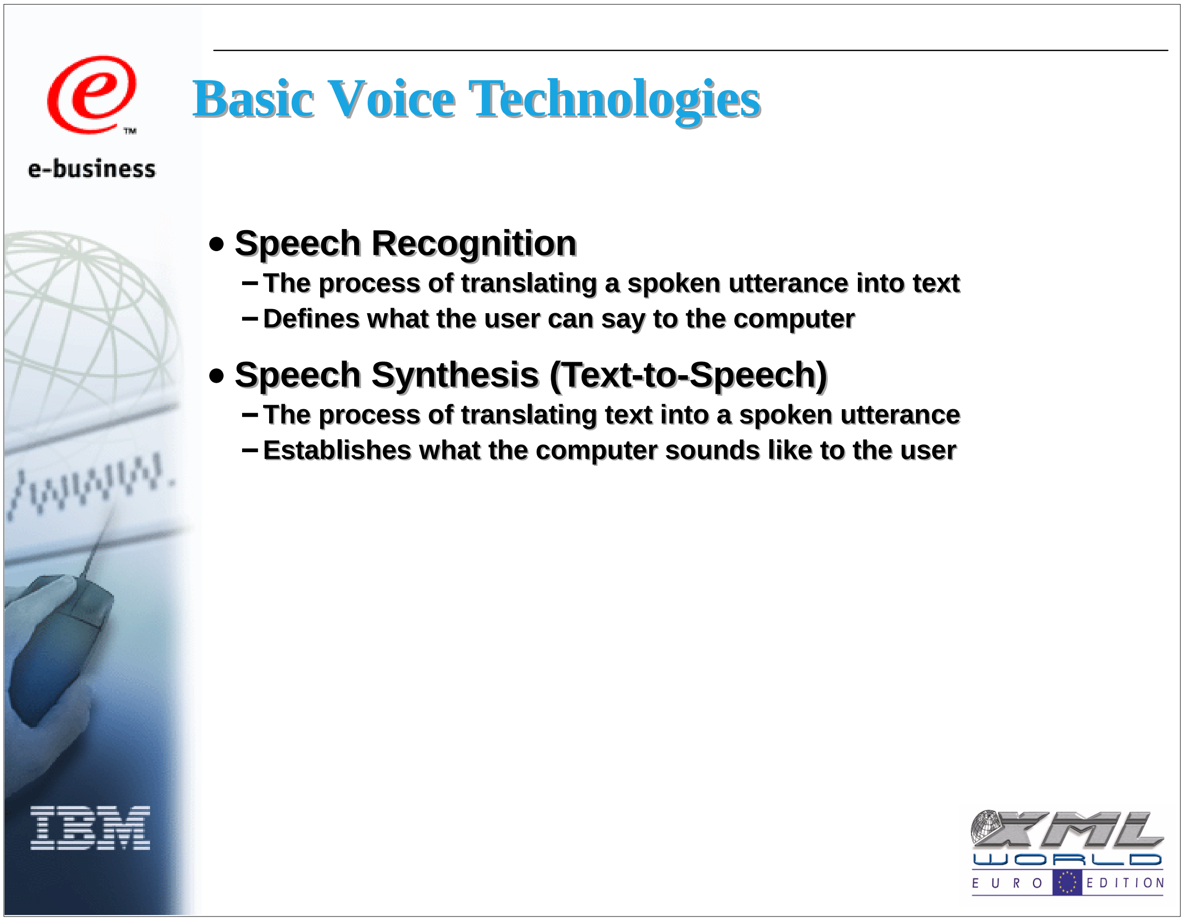
Leveraging VoiceXML

XMLWorld March 27, 2001

Dana Vizner
Software Engineer, User Centered Design
IBM Voice Systems Development
West Palm Beach, FL
danav@us.ibm.com

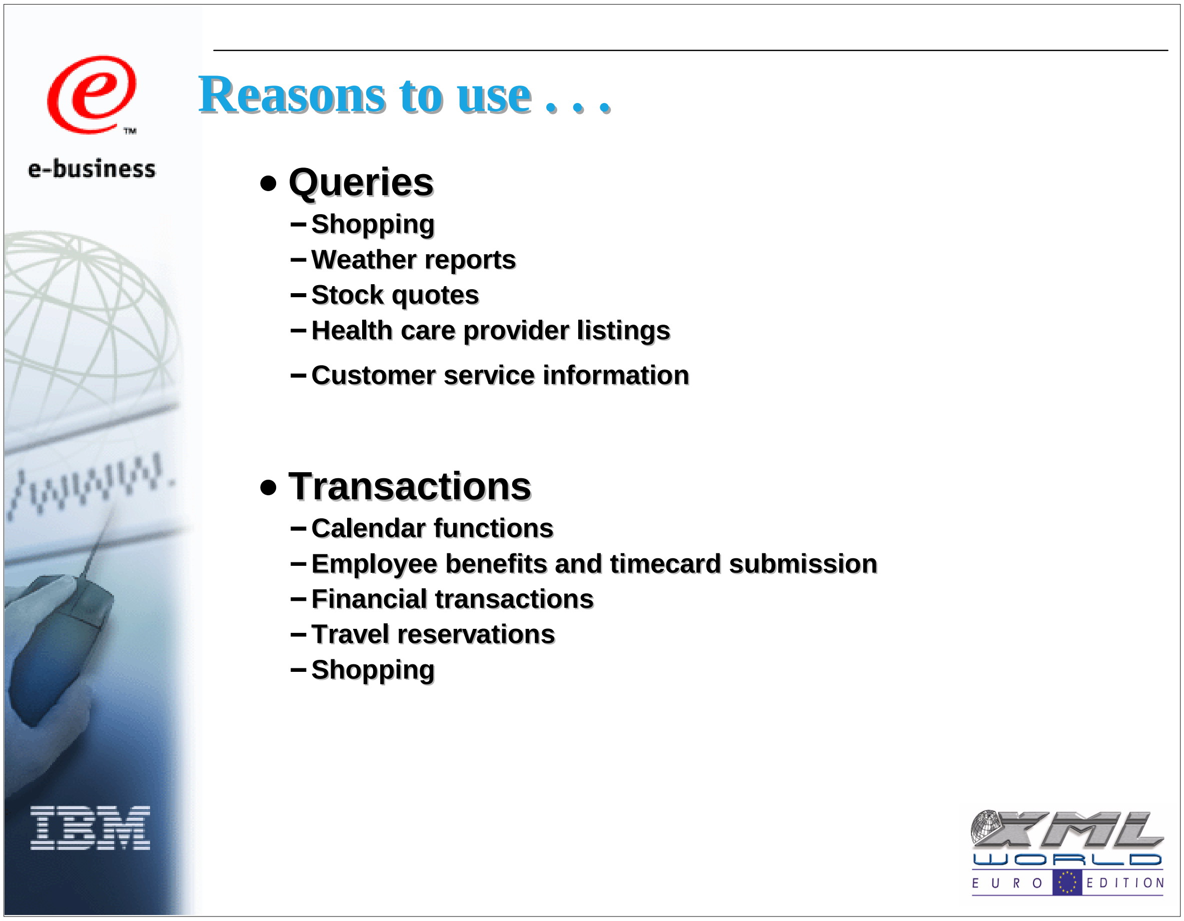
Session Objectives

- Introduction to Voice Technologies
- What is a Voice Server?
- Developing VoiceXML Applications
- Speech User Interface Considerations


e-business
IBM
XML WORLD
EURO EDITION

Basic Voice Technologies

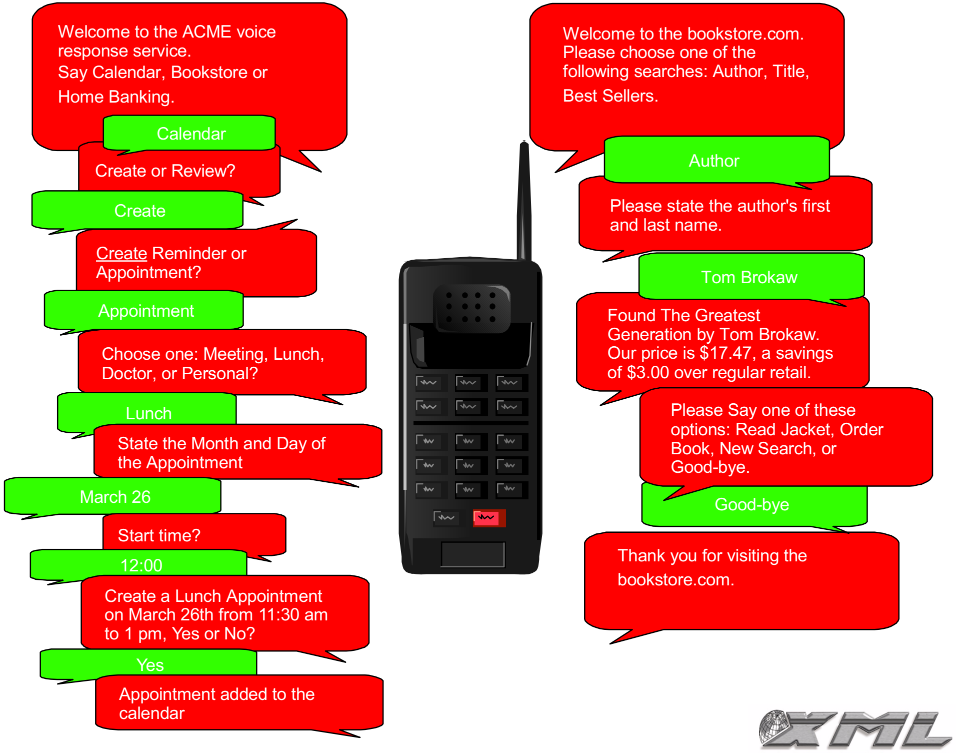
- Speech Recognition
  - The process of translating a spoken utterance into text
  - Defines what the user can say to the computer
- Speech Synthesis (Text-to-Speech)
  - The process of translating text into a spoken utterance
  - Establishes what the computer sounds like to the user


e-business
IBM
XML WORLD
EURO EDITION

Reasons to use . . .

- Queries
  - Shopping
  - Weather reports
  - Stock quotes
  - Health care provider listings
  - Customer service information
- Transactions
  - Calendar functions
  - Employee benefits and timecard submission
  - Financial transactions
  - Travel reservations
  - Shopping

Surfing Voice Web Sites



Welcome to the ACME voice response service.
Say Calendar, Bookstore or Home Banking.

Calendar

Create or Review?

Create

Create Reminder or Appointment?

Appointment

Choose one: Meeting, Lunch, Doctor, or Personal?

Lunch

State the Month and Day of the Appointment

March 26

Start time?

12:00

Create a Lunch Appointment on March 26th from 11:30 am to 1 pm, Yes or No?

Yes

Appointment added to the calendar

Welcome to the bookstore.com. Please choose one of the following searches: Author, Title, Best Sellers.

Author

Please state the author's first and last name.

Tom Brokaw

Found The Greatest Generation by Tom Brokaw. Our price is $17.47, a savings of $3.00 over regular retail.

Please Say one of these options: Read Jacket, Order Book, New Search, or Good-bye.

Good-bye

Thank you for visiting the bookstore.com.

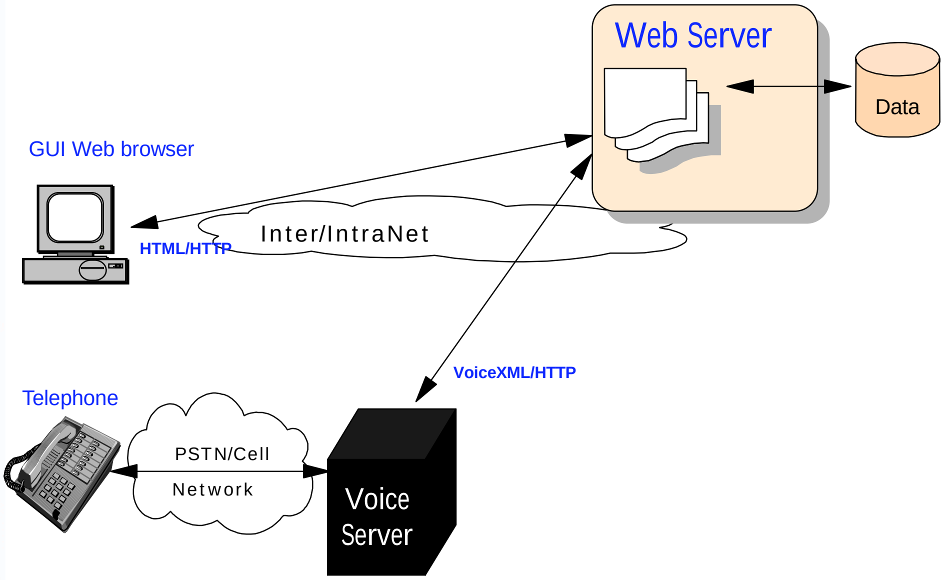
What is a Voice Server?

- Conversational Interface for Web Applications
- A speech analog to GUI browsing
- Fit into the standard web server architecture
  - Minimizes need of web developers to learn speech
  - Uses existing back-end data processing
- Telephony Deployment environment
  - IBM WebSphere Voice Server with ViaVoice™ Technology (VoIP Solution)
- Desktop Development environment
  - IBM WebSphere Voice Server Software Development Kit

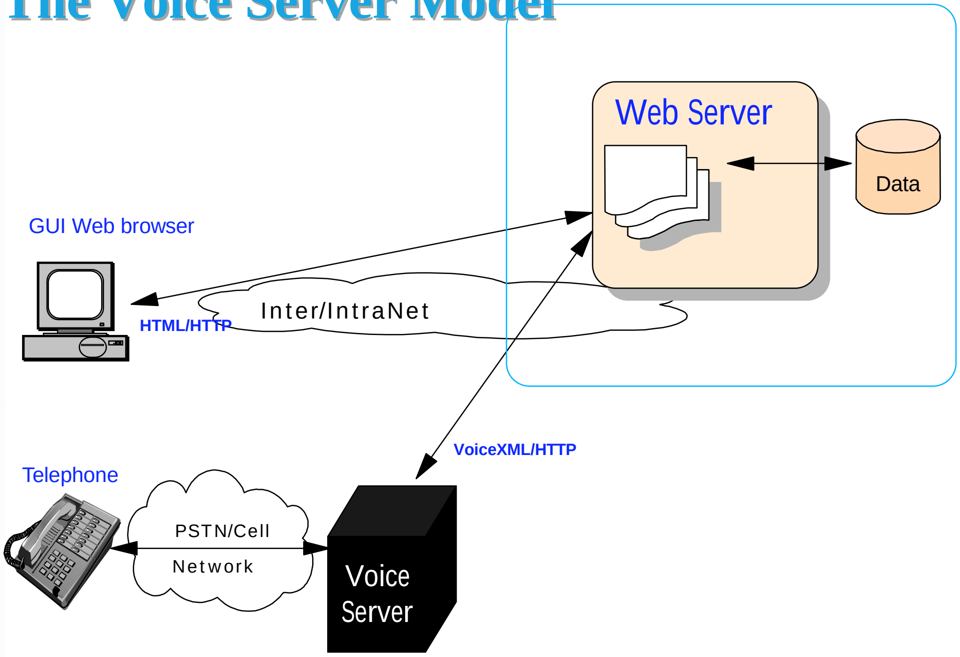
Voice Server (cont.)

- It is
  - Talking to web sites
- It is NOT
  - Reading HTML pages to users
  - A talking enabled Netscape or IE

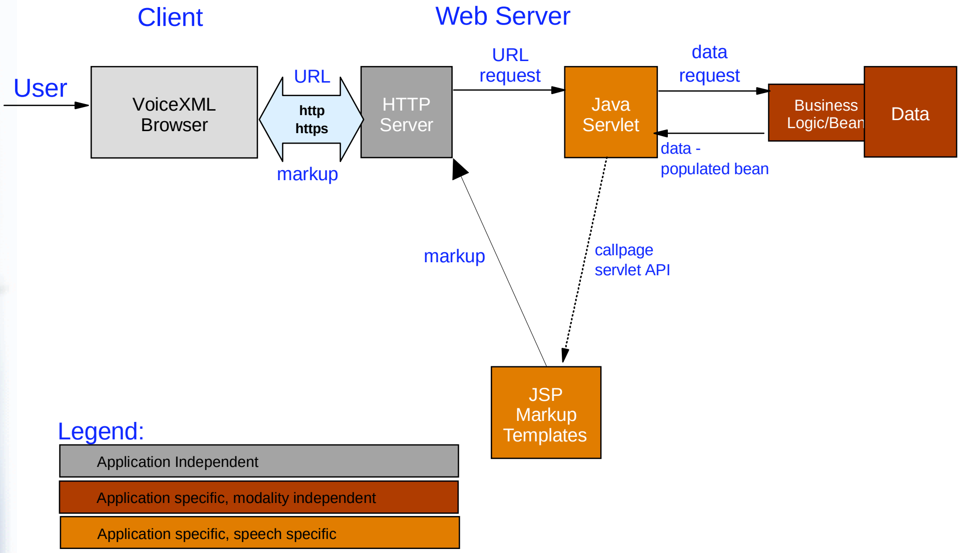
The Voice Server Model



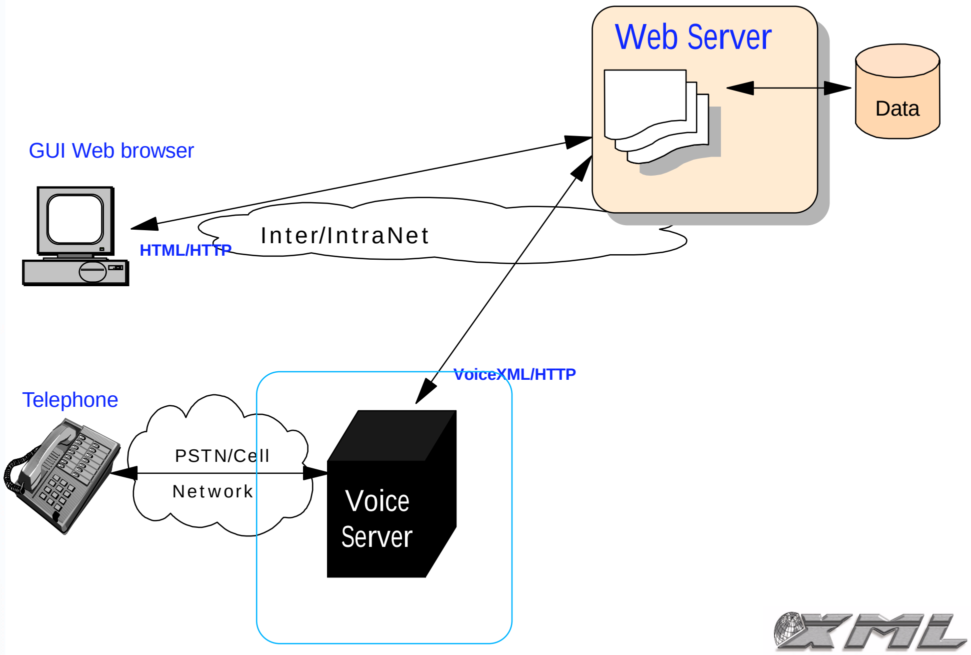
The Voice Server Model



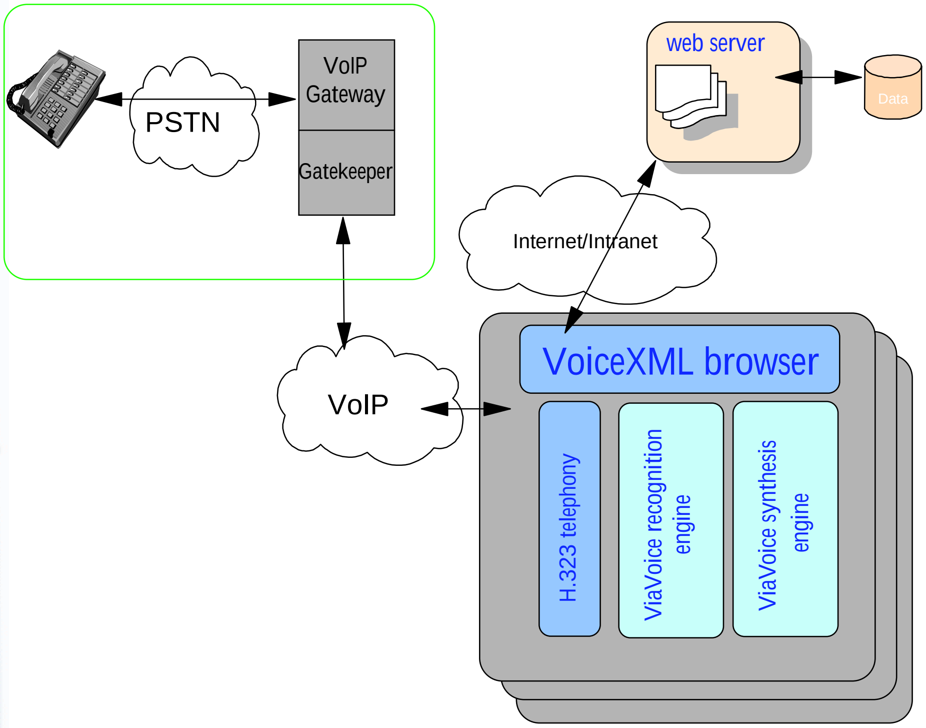
Web Server Connection

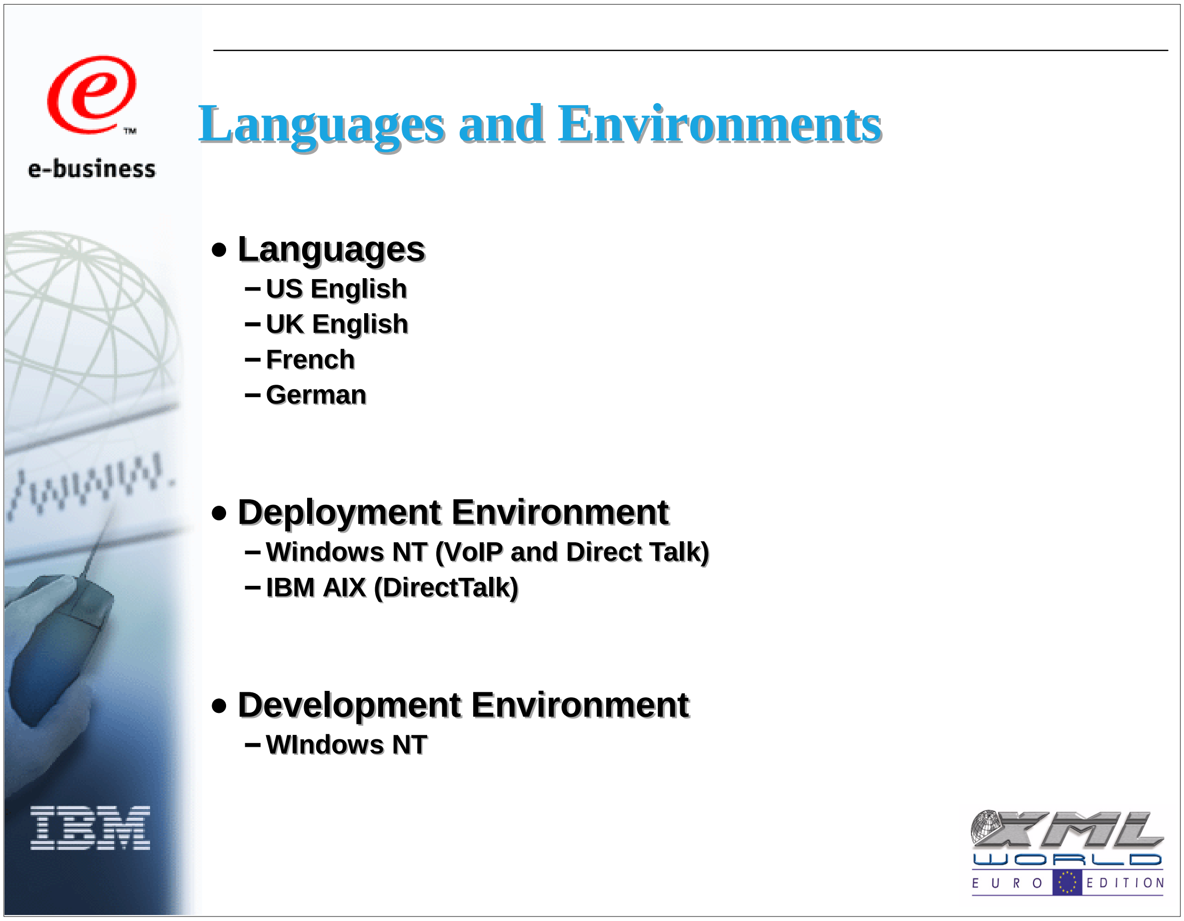


The Voice Server Model



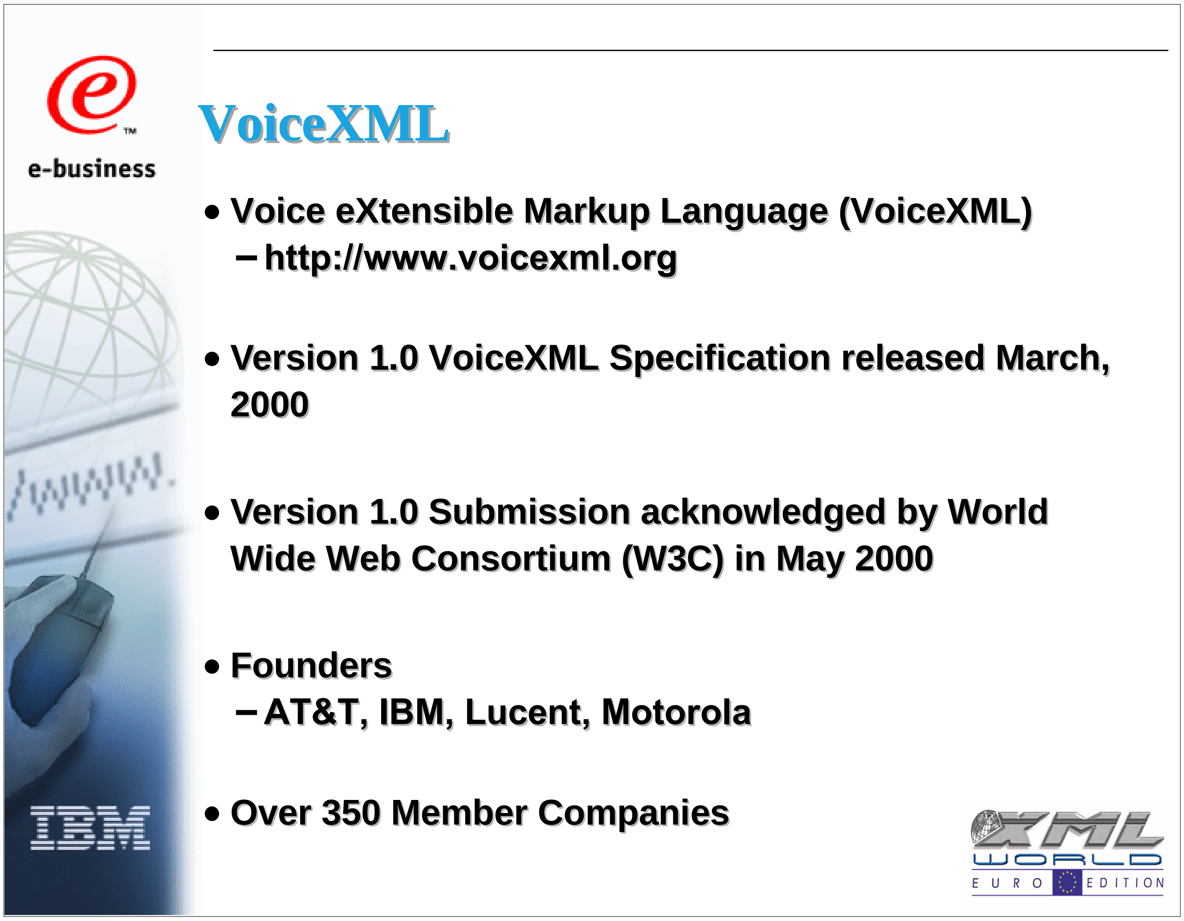
Deployment Runtime





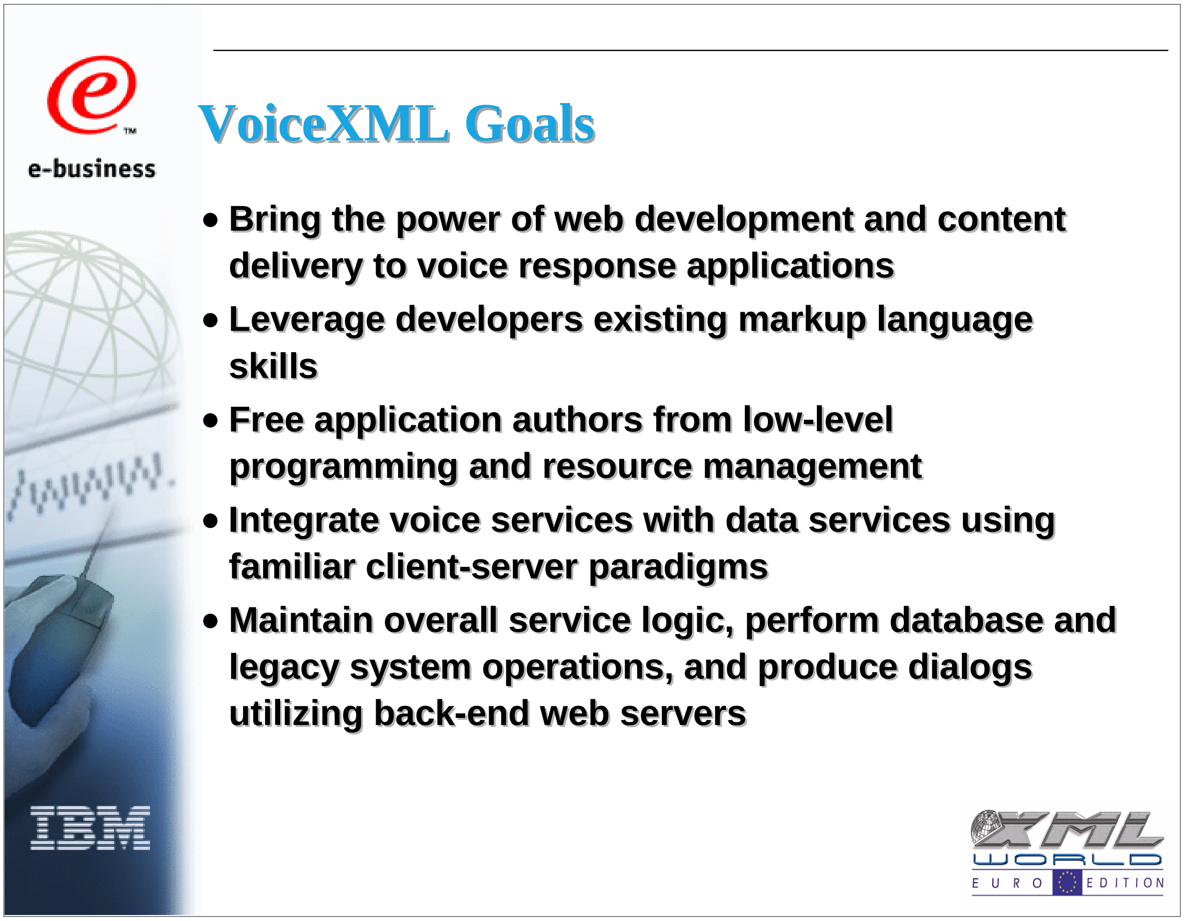
Languages and Environments

- Languages
  - US English
  - UK English
  - French
  - German
- Deployment Environment
  - Windows NT (VoIP and Direct Talk)
  - IBM AIX (DirectTalk)
- Development Environment
  - WIndows NT



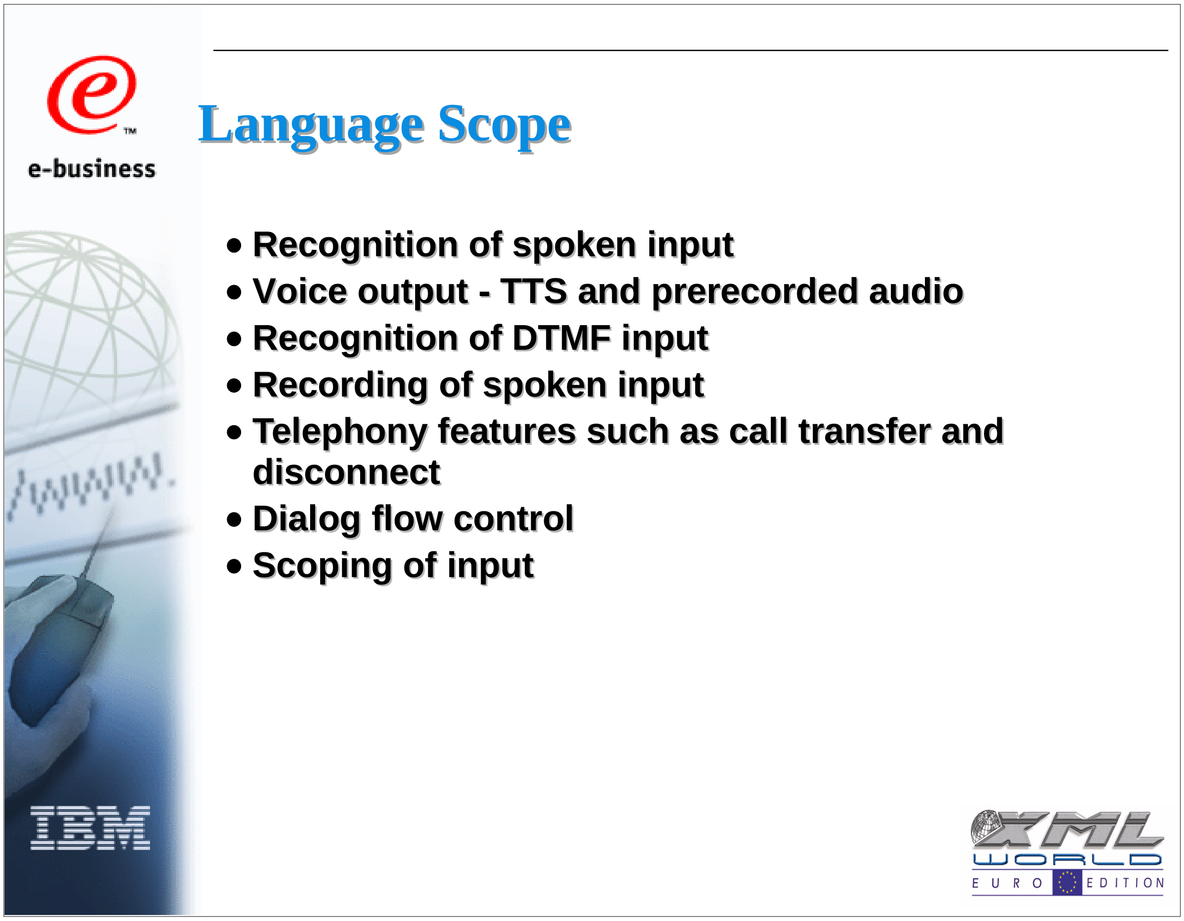
VoiceXML

- Voice eXtensible Markup Language (VoiceXML)
  - http://www.voicexml.org
- Version 1.0 VoiceXML Specification released March, 2000
- Version 1.0 Submission acknowledged by World Wide Web Consortium (W3C) in May 2000
- Founders
  - AT&T, IBM, Lucent, Motorola
- Over 350 Member Companies


e-business
IBM
XML WORLD EURO EDITION

VoiceXML Goals

- Bring the power of web development and content delivery to voice response applications
- Leverage developers existing markup language skills
- Free application authors from low-level programming and resource management
- Integrate voice services with data services using familiar client-server paradigms
- Maintain overall service logic, perform database and legacy system operations, and produce dialogs utilizing back-end web servers

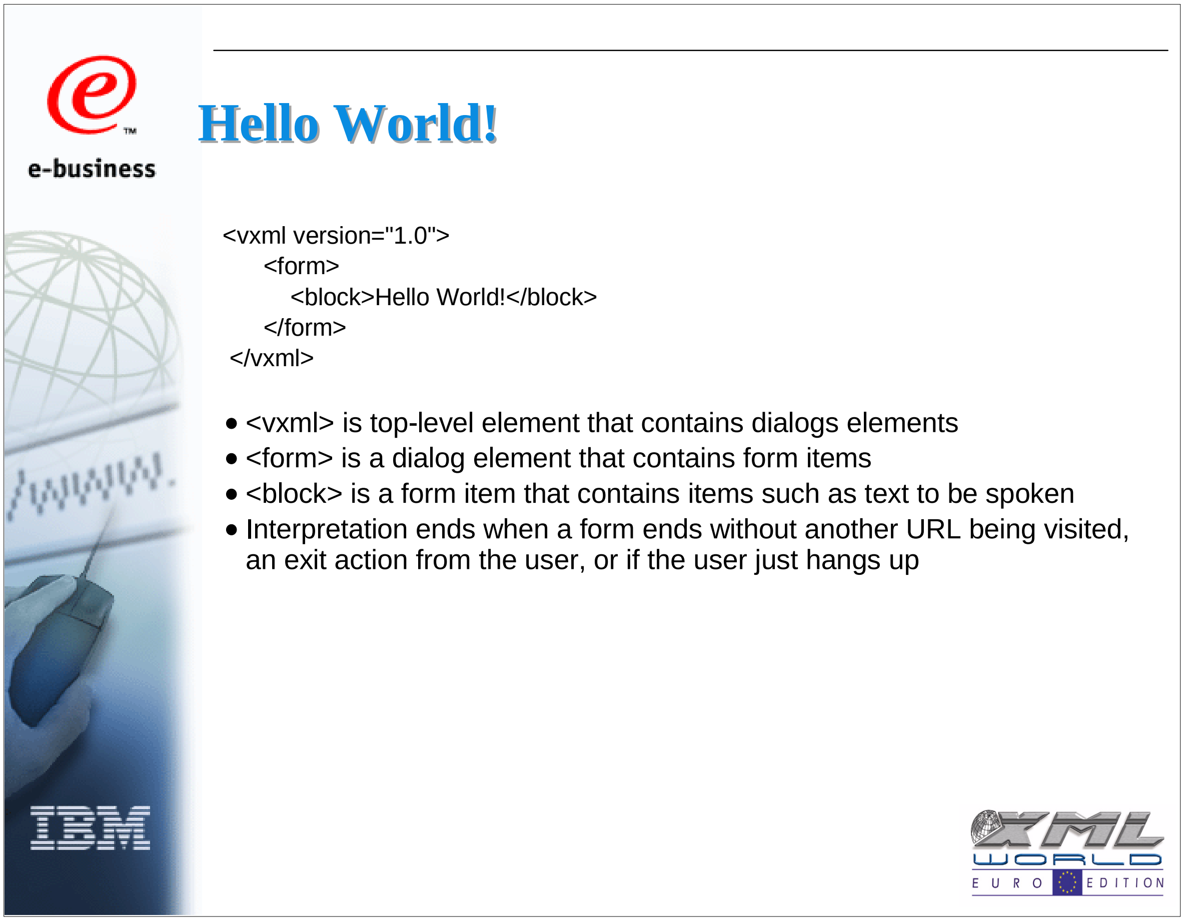

e-business
IBM
XML WORLD EURO EDITION

Language Scope

- Recognition of spoken input
- Voice output - TTS and prerecorded audio
- Recognition of DTMF input
- Recording of spoken input
- Telephony features such as call transfer and disconnect
- Dialog flow control
- Scoping of input

Language Features

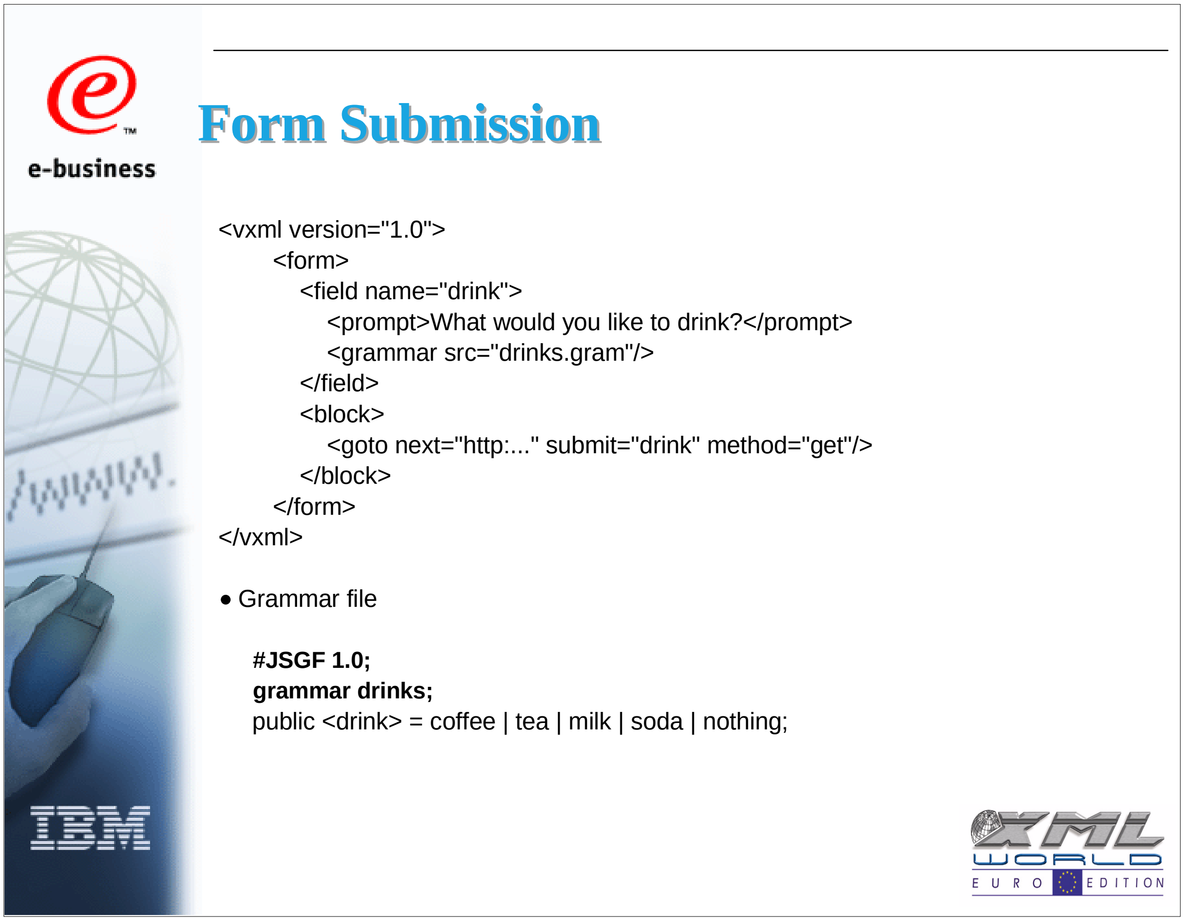
- document structure
  - <vxml> <meta>
- forms
  - <form> <field> <initial> <block> <filled>
- menus
  - <menu> <choice> <link> <enumerate>
- recognition control
  - <dtmf> <grammar> <property>
- control flow
  - <goto> <submit> <subdialog> <param>
- scripting
  - <throw> <exit> <return> <script> <if> <elseif> <else> <var> <assign> <clear>
- exception events
  - <catch> <error> <help> <noinput> <nomatch>
- telephony
  - <disconnect> <transfer>
- specialized input
  - <record> <transcribe> <object>
- audio and tts output
  - <break> <div> <emp> <pros> <sayas> <value> <prompt> <reprompt> <audio>



Hello World!

```
<vxml version="1.0">
     <form>
        <block>Hello World!</block>
     </form>
 </vxml>
```

- <vxml> is top-level element that contains dialogs elements
- <form> is a dialog element that contains form items
- <block> is a form item that contains items such as text to be spoken
- Interpretation ends when a form ends without another URL being visited, an exit action from the user, or if the user just hangs up

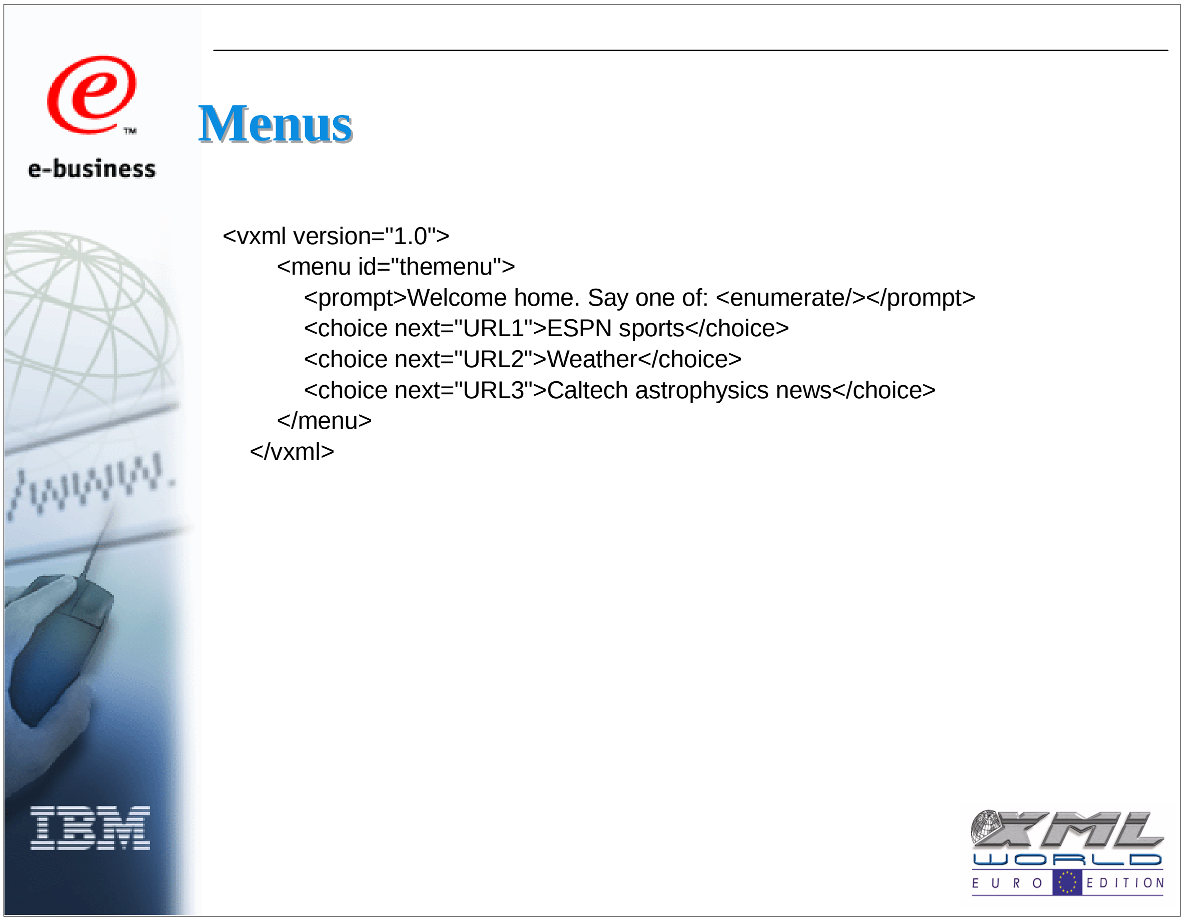


Form Submission

```
<vxml version="1.0">
        <form>
            <field name="drink">
                <prompt>What would you like to drink?</prompt>
                <grammar src="drinks.gram"/>
            </field>
            <block>
                <goto next="http:..." submit="drink" method="get"/>
            </block>
        </form>
</vxml>
```

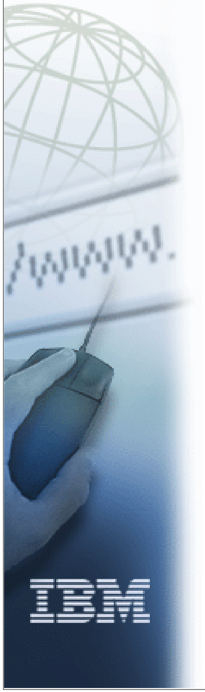
- Grammar file

```
#JSGF 1.0;
grammar drinks;
public <drink> = coffee | tea | milk | soda | nothing;
```



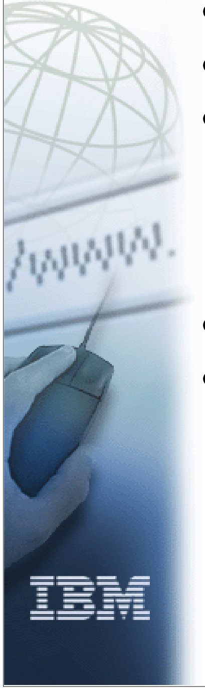
Menus

```
<vxml version="1.0">
        <menu id="themenu">
            <prompt>Welcome home. Say one of: <enumerate/></prompt>
            <choice next="URL1">ESPN sports</choice>
            <choice next="URL2">Weather</choice>
            <choice next="URL3">Caltech astrophysics news</choice>
        </menu>
    </vxml>
```



Building the Applications

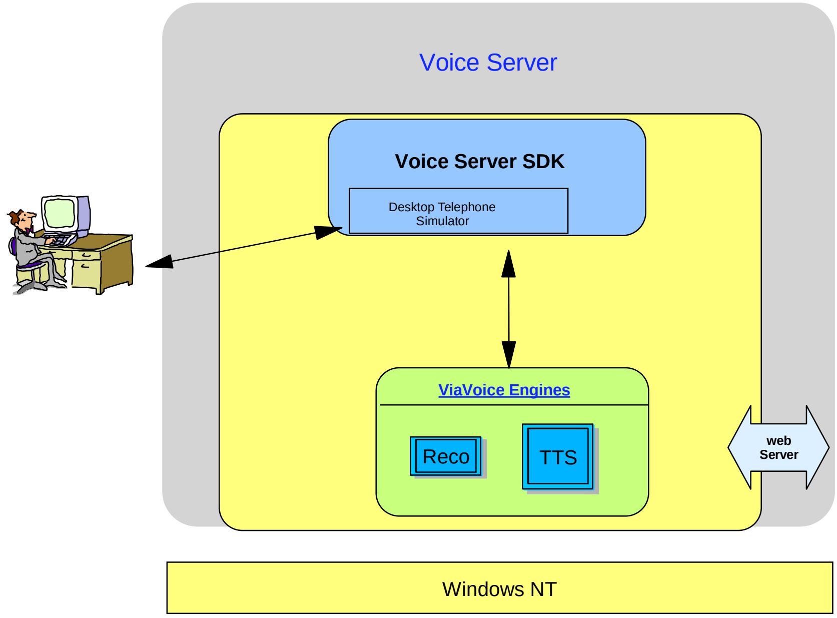
- Assumptions
  - Web developers know little about speech reco technologies
  - Speech (and IVR) developers often know little about the web
- Goals
  - Make speech app development as easy as GUI web development
  - Leverage existing web app logic using the same web programming model
  - Bring web developers into the voice space
  - Bring voice/IVR developers into the web space
  - With as little pain as possible!



IBM WebSphere Voice Server SDK

- Desktop test environment
- Supplies links for integration into WebSphere Studio
- Includes Programmer's Guide
  - Speech User Interface Guidelines
  - Hints, Tips, and Best Practices
  - VoiceXML Language Details
  - IBM Extension Details
- VoiceXML 1.0 Specification
- VoiceXML Audio Sample

Development Runtime



Voice Server

Voice Server SDK

Desktop Telephone Simulator

ViaVoice Engines

Reco

TTS

web Server

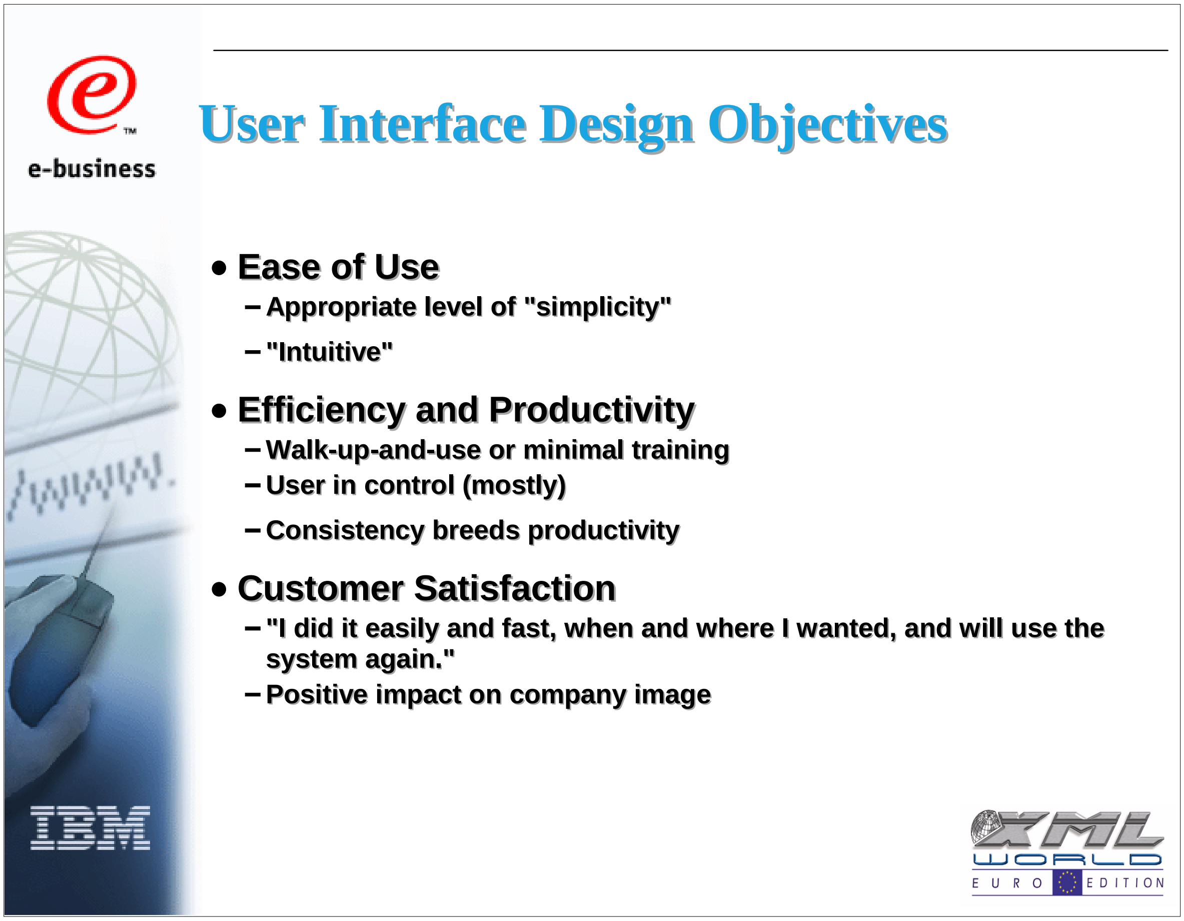
Windows NT

WebSphere Studio

- VoiceXML Editor
- Grammar Editor Connection
- Preview VoiceXML pages with the Voice Server SDK from navigation tree
- Publish pages to WebSphere Application Server
- VoiceXML Samples

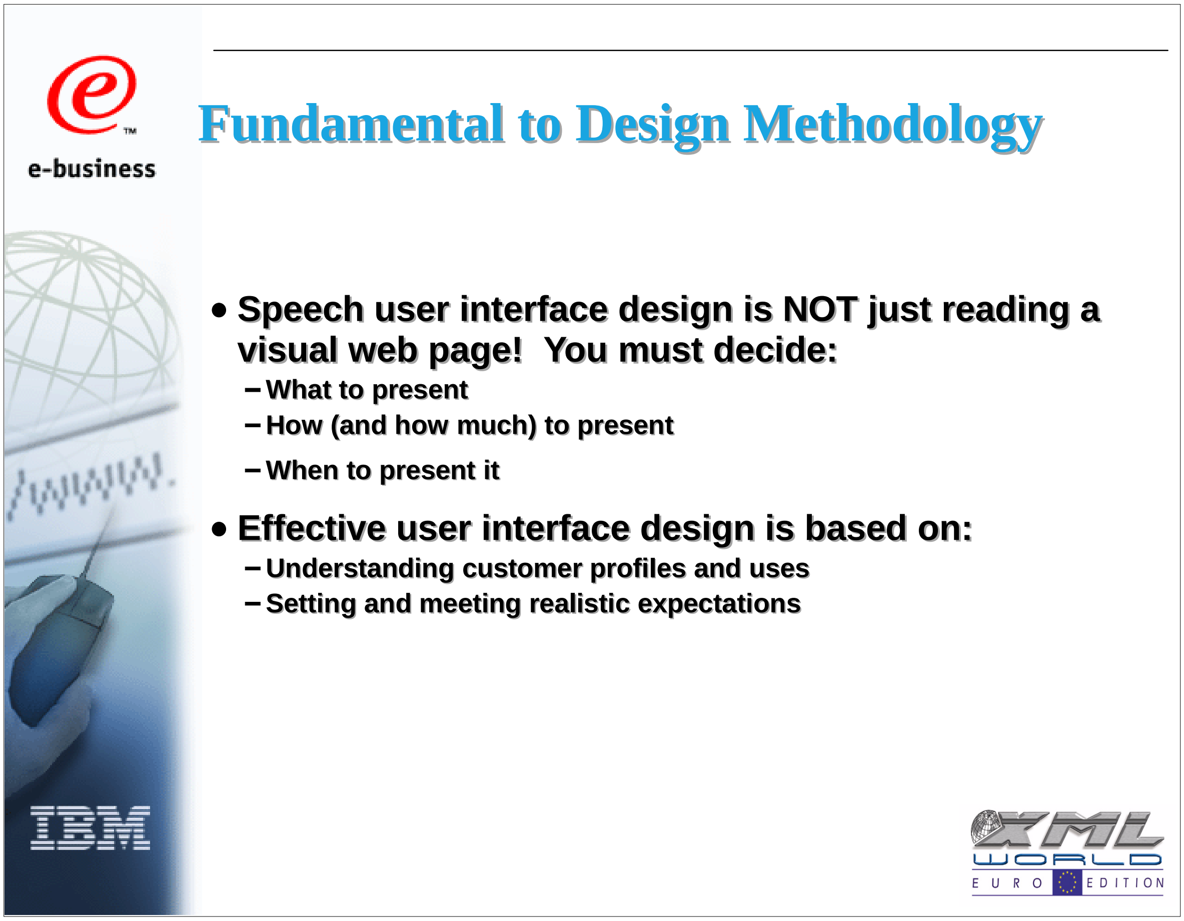
Tools currently in development at IBM Voice Systems

- Integrated Development Environment(IDE) for Voice Application Development
  - Uses IBM Middleware tooling framework (Eclipse)
  - Provides tools for:
    - VoiceXML development
    - Grammar development
      - What the speech recognition engine will recognize
    - Pronunciation development
      - How the Text-to-Speech engine pronounces words
- Re-usable VoiceXML Components
- VoiceXML Samples



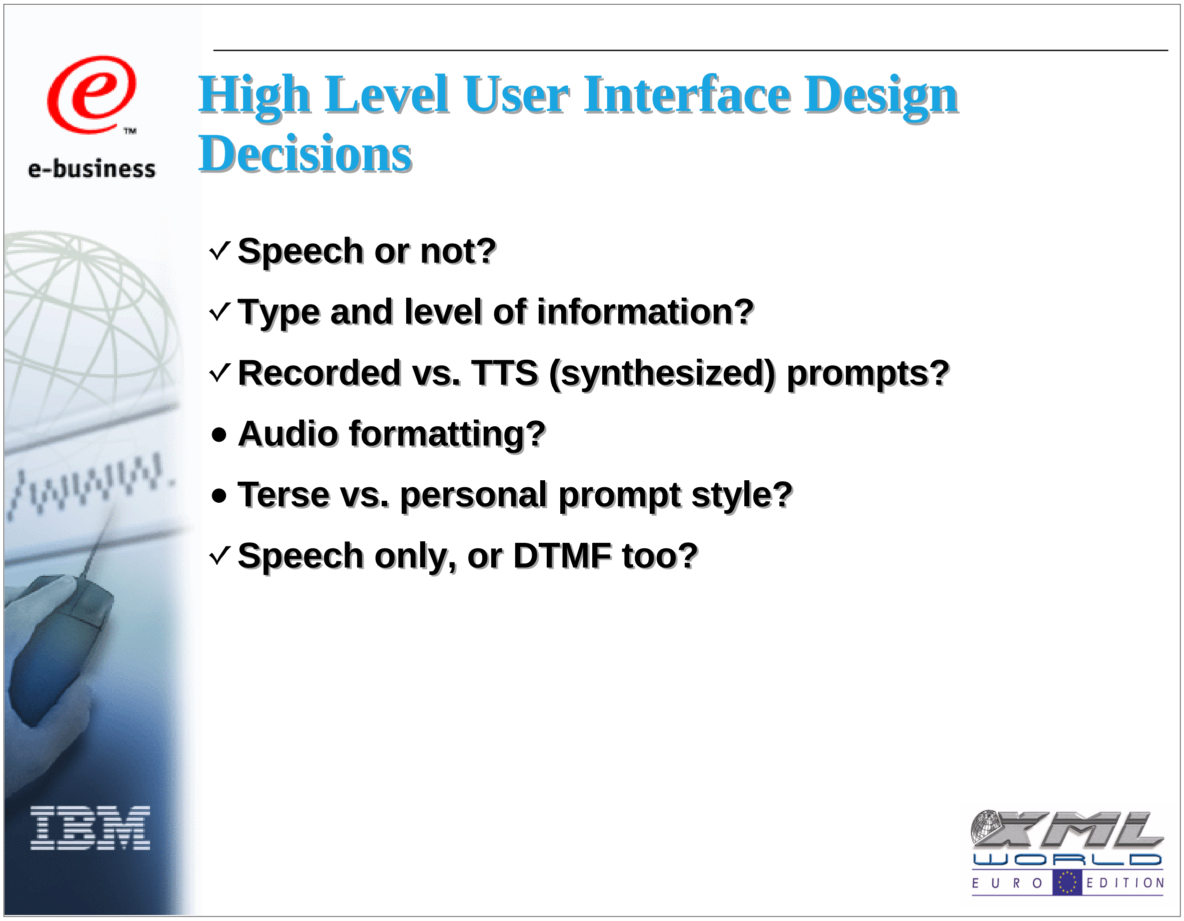
User Interface Design Objectives

- Ease of Use
  - Appropriate level of "simplicity"
  - "Intuitive"
- Efficiency and Productivity
  - Walk-up-and-use or minimal training
  - User in control (mostly)
  - Consistency breeds productivity
- Customer Satisfaction
  - "I did it easily and fast, when and where I wanted, and will use the system again."
  - Positive impact on company image



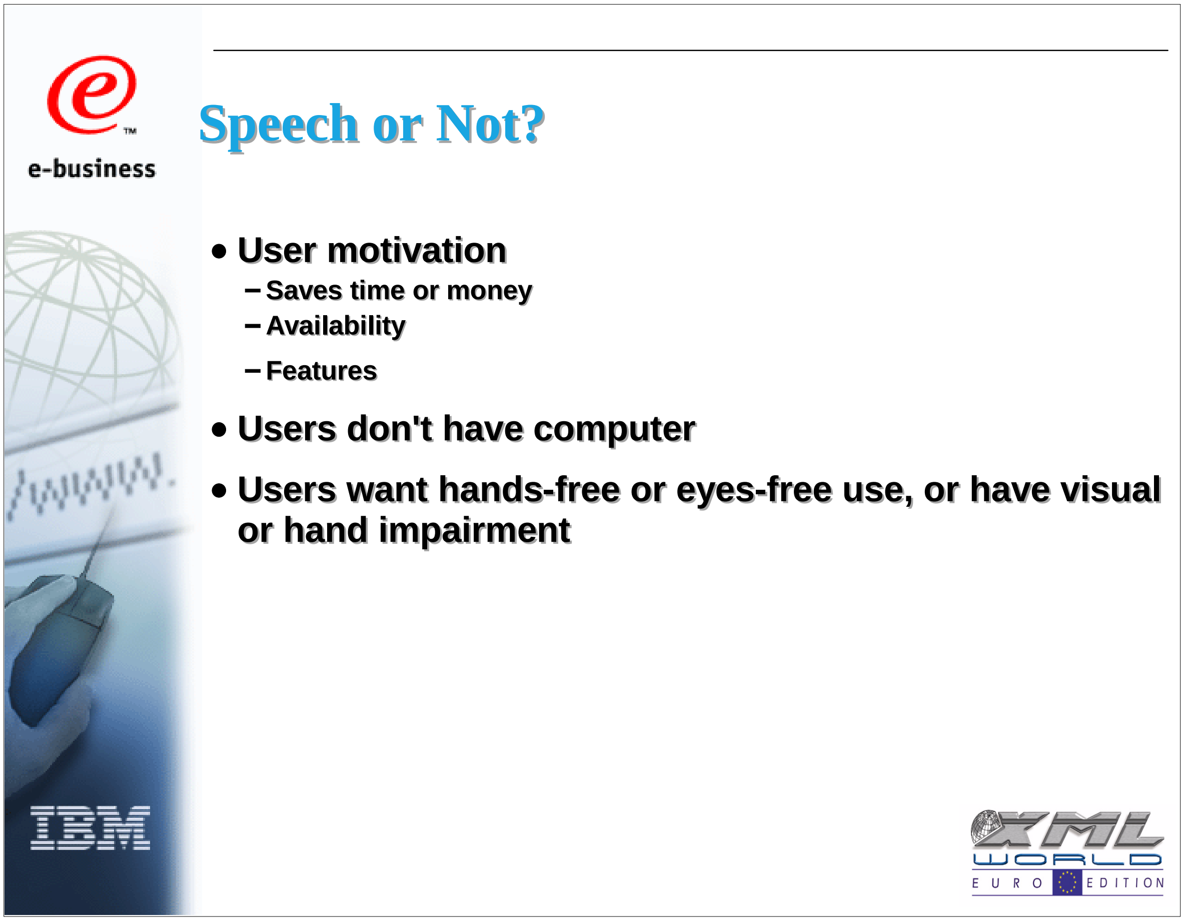
Fundamental to Design Methodology

- Speech user interface design is NOT just reading a visual web page! You must decide:
  - What to present
  - How (and how much) to present
  - When to present it
- Effective user interface design is based on:
  - Understanding customer profiles and uses
  - Setting and meeting realistic expectations



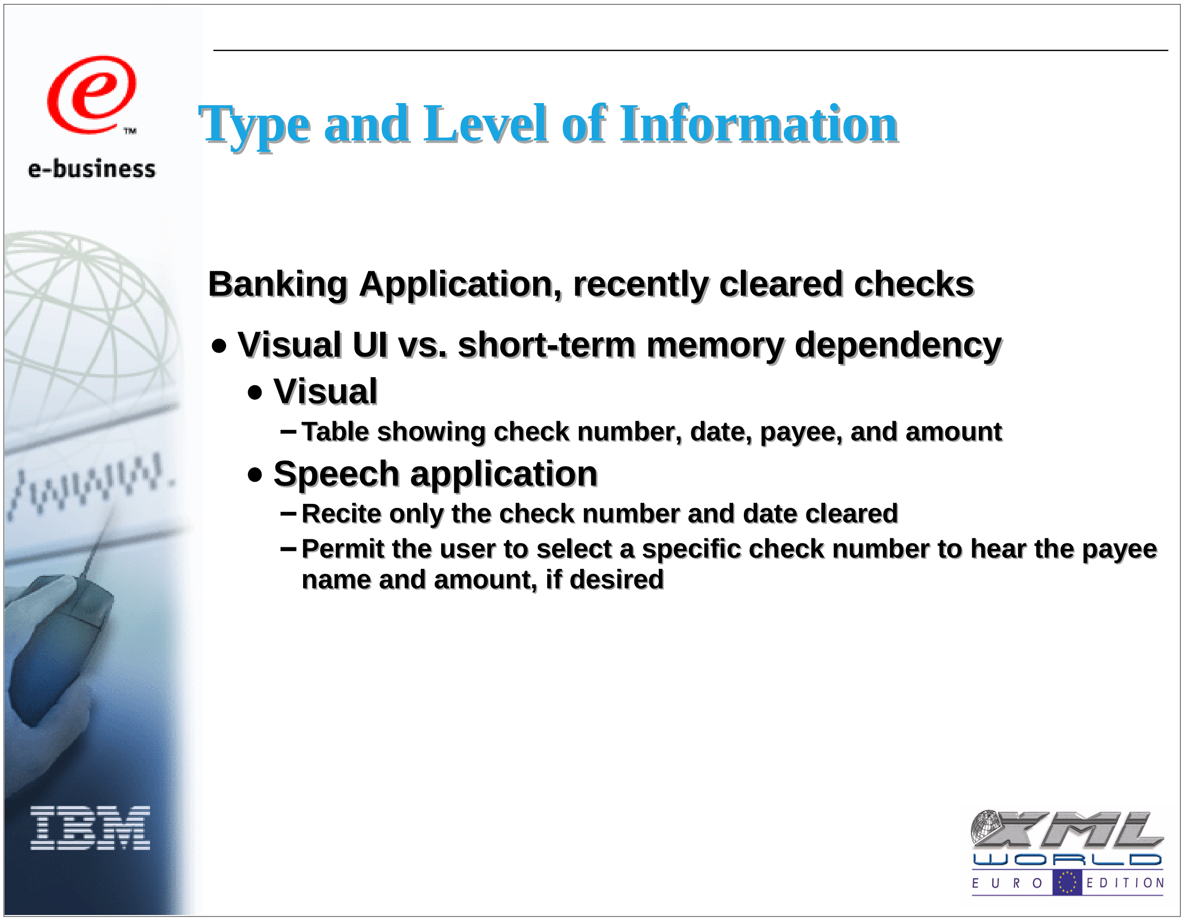
High Level User Interface Design Decisions

- ✓ Speech or not?
- ✓ Type and level of information?
- ✓ Recorded vs. TTS (synthesized) prompts?
- Audio formatting?
- Terse vs. personal prompt style?
- ✓ Speech only, or DTMF too?



Speech or Not?

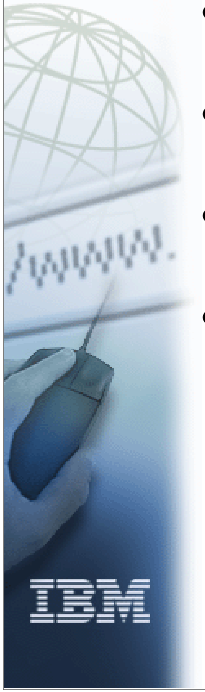
- User motivation
  - Saves time or money
  - Availability
  - Features
- Users don't have computer
- Users want hands-free or eyes-free use, or have visual or hand impairment



Type and Level of Information

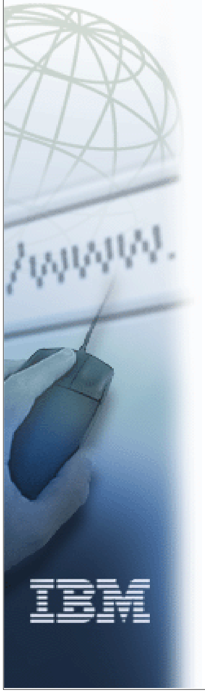
Banking Application, recently cleared checks

- Visual UI vs. short-term memory dependency
  - Visual
    - Table showing check number, date, payee, and amount
  - Speech application
    - Recite only the check number and date cleared
    - Permit the user to select a specific check number to hear the payee name and amount, if desired



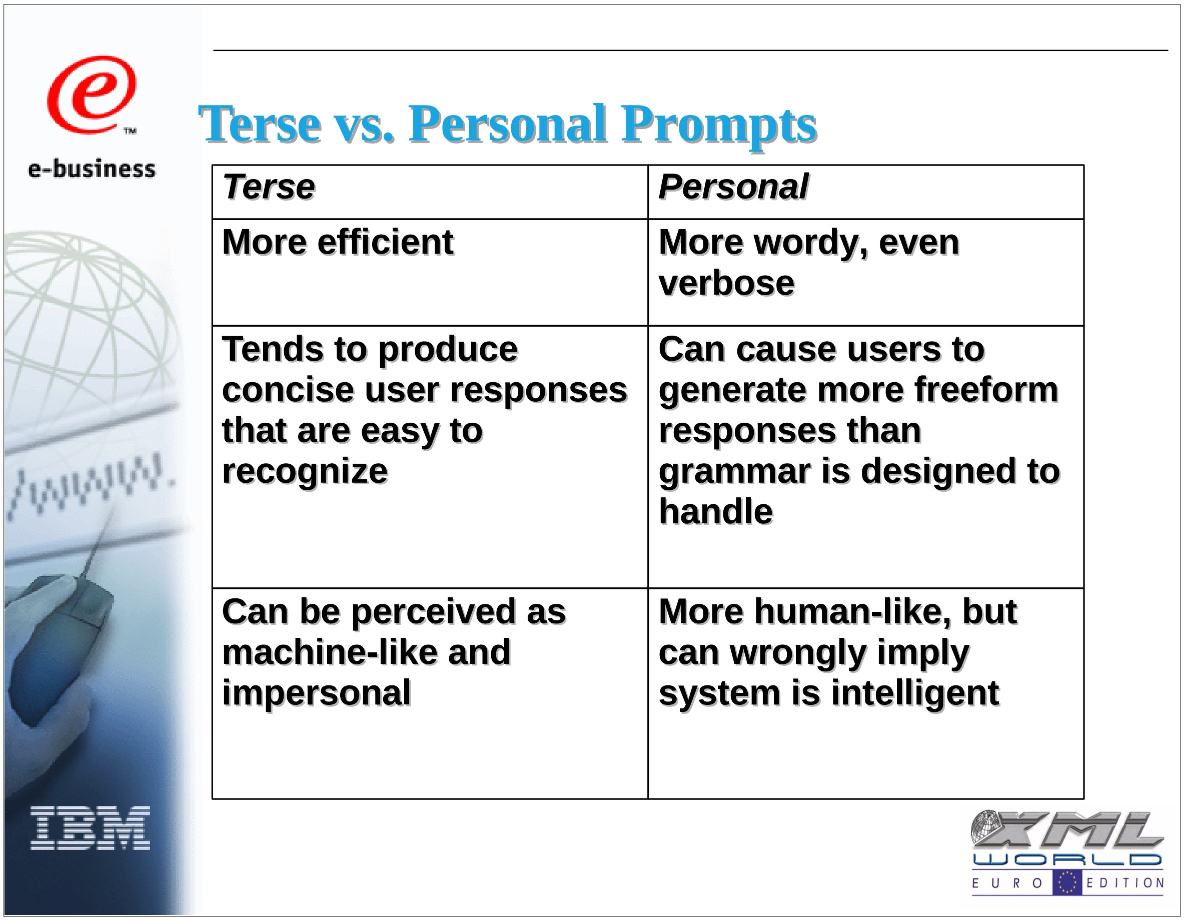
Recorded vs. Synthesized Prompts

- Use Text-to-Speech (synthesized) prompts during development and for unbounded data
- Use professionally recorded prompts for everything else
- Stick to one voice unless there's a clear design goal to use more
- Avoid using TTS and recorded voice for prompts in same product



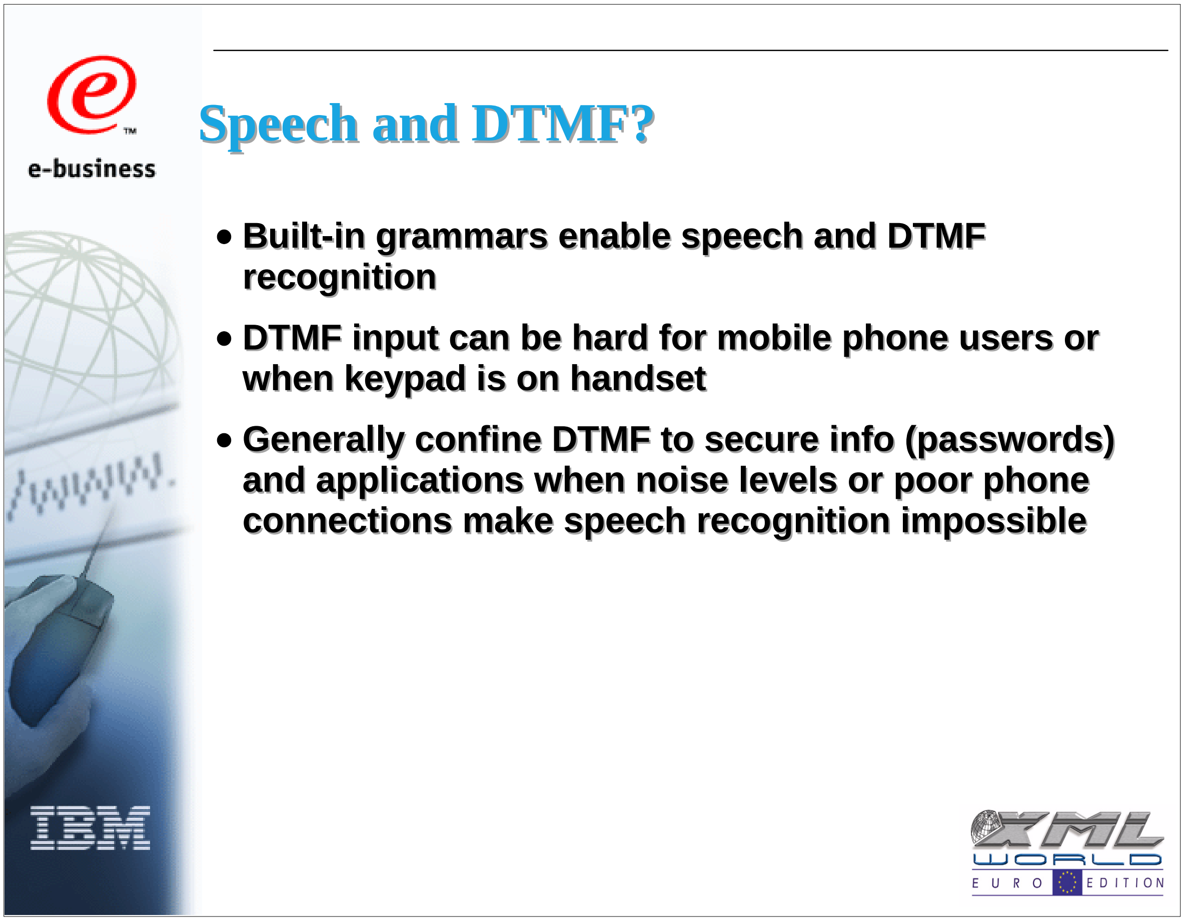
Audio Formatting

- Audio formatting refers to non-speech clues that accompany or overlay information
- Like indenting, capitalization, font size,style changes and color-coding in a visual interface
- Examples:
  - Turn-taking tone
  - Beep for bullet
  - Background music for secure transaction
  - Pitch or volume for emphasis



Terse vs. Personal Prompts

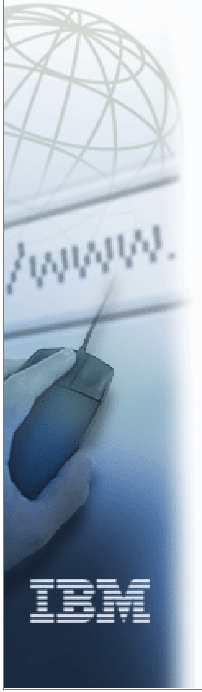
Terse	*Personal*
More efficient	More wordy, even verbose
Tends to produce concise user responses that are easy to recognize	Can cause users to generate more freeform responses than grammar is designed to handle
Can be perceived as machine-like and impersonal	More human-like, but can wrongly imply system is intelligent



Speech and DTMF?

- Built-in grammars enable speech and DTMF recognition
- DTMF input can be hard for mobile phone users or when keypad is on handset
- Generally confine DTMF to secure info (passwords) and applications when noise levels or poor phone connections make speech recognition impossible

e-business

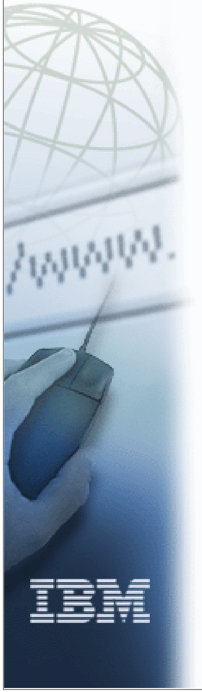


Final Thoughts....

- IBM WebSphere Voice Server and VoiceXML allow you to increase your potential customer base to anyone owning a telephone
- Reuse your already existing Web infrastructure, business logic and data... just add a new presentation layer
- Already existing markup language skills can easily migrate to the VoiceXML model
- Become experts in Speech User Interface design! VoiceXML and the IBM Development environment provide an easy way to prototype the User Interface and try it on real users.

XML WORLD EURO EDITION

e-business



Reference Material

- www.software.ibm.com/voice
- www.voicexmlforum.org
- www.alphaworks.ibm.com/tech
- www.ibm.com/software/webservers

XML WORLD EURO EDITION

Questions?