All-Mode Application Architecture Meets the Need

April 23, 2010
SpeechCycle is the leading provider of Rich Phone Applications (RPAs)

- Rapidly growing award-winning innovators in self-service transformation
  - Deloitte’s 67th fastest growing company in North America (2700%)
  - VC funded software company founded in 2001; based in New York City

- Experts in speech science, software development and voice platform design

- Processing millions of automated phone transactions for leading service providers
  - We triple call automation rates with our Rich Phone Application (RPA) platform
  - We automate calls that are typically handled by live agents (6-8 AHT)
  - We provide full suite of applications for service providers that are deeply integrated with OSS/BSS systems
  - In 2009, SpeechCycle virtual agents processed 100 million voice self-service transactions
  - We cut tens of millions of dollars of cost out of service provider operations

*RPAs are a high-growth category of software solutions that orchestrate enterprise and web systems with natural language dialogs to enable high service provider ROI through unmatched call automation rates*
Amdocs Virtual Agent RPAs vs. Conventional IVR

Rich Phone Applications (RPAs) deliver superior customer experience that drives higher levels of customer adoption over conventional IVR-based applications.

Traditional IVRs
- Traditional IVRs take a long time to build and provide only surface-level enterprise integration
- IVR flows are a snapshot in time that often get out of synch with other interaction channels. Keeping them in synch takes a significant amount of money and resources
- The hand-off between IVRs and live agents is inefficient and is a significant source of customer frustration

Virtual Agent RPAs
- Virtual Agent RPAs fully extends the value of existing CRM software to the dominant customer touch point.
- Virtual Agent RPAs seamlessly transition calls to live agents -- every Virtual Agent interaction becomes an integral part of the customer profile
- Asynchronous transaction execution optimizes the customer experience by shortening the time to resolution
- Personalization ensures highly relevant and in-context caller experience
- Self-improving dialog management through continuous updates based upon user-generated input -- not ship and forget model!
Making customer service more effective

• The Expectation
  • People are beginning to expect cross-modal integration.
  • To provide the most effective customer service, Speech > Web > Chat > etc. should all be aware of each other in real time.

• The Problem
  • Speech is “different.”
  • Existing speech development methodology does not extend gracefully to other modes.

• What we’ll cover
  • Lessons Learned
    • Mobile Troubleshooting
    • Extending Modes
    • Problems with Time
    • Reorganizing our IP
    • Reorganizing our development process
  • A Use Case
Smart Phone Distribution

October, 2009
Source: ComScore
Dumb phones rule

Smart Phone Penetration

October, 2009
Source: ComScore
Mobile Troubleshooting

• **Background: Technical troubleshooting**
  • *Internet*
  • *Video*
  • *Telephony*

• **Wireless problems**
  • Lost Internet
  • Slow & Intermittent
  • Unblocking websites
  • Password reset
  • Email problems
  • No signal
  • Spotty signal

• **Troubleshooting Steps**
  • First-time setup
  • Moving equipment
  • Antenna troubleshooting
  • Connection manager software
  • Error message analysis
  • Password/Credentials reset
  • International roaming

• **Unique Challenges to Wireless**
  • Storms
  • RF interference
  • GSM interference
  • Parties
  • Cave dwellers
Felt dirty, but it had to be done

- **Calling back from another phone**
- **Session bookmarks**
  - ID caller, then jump to the last step taken
  - Potential for continuing on the web
- **What about adding even more modes or channels?**
- **Would not scale because:**
  - Hard to build
  - Hard to maintain
  - Hard to extend
  - Almost impossible to explain
<table>
<thead>
<tr>
<th>Mode</th>
<th>What now?</th>
<th>Time-Limited</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speech</td>
<td>Some VXML app</td>
<td>Yes</td>
</tr>
<tr>
<td>Agent</td>
<td>A butt in a seat</td>
<td>Yes*</td>
</tr>
<tr>
<td>Web</td>
<td>The web</td>
<td>No</td>
</tr>
<tr>
<td>Mobile Web</td>
<td>Your smart-phone’s browser</td>
<td>No</td>
</tr>
<tr>
<td>Web IM</td>
<td>Instant messenger</td>
<td>No</td>
</tr>
<tr>
<td>Mobile IM</td>
<td>IM client on phone</td>
<td>No</td>
</tr>
<tr>
<td>SMS</td>
<td>Mobile keyboard or T9</td>
<td>No</td>
</tr>
<tr>
<td>Live Chat</td>
<td>Web-based chat window with human</td>
<td>No</td>
</tr>
<tr>
<td>Bot Chat</td>
<td>Web-based chat window with robot</td>
<td>No</td>
</tr>
</tbody>
</table>
1. No documentation capabilities
2. Options list limitations
3. Information dominance
4. No indefinite pauses
5. Reliable go back/undo
6. Timeouts carry significance

- VXML handles most of this
- Reporting is usually based on speech issues
Development process

• The old, “good” way
  1. Business analysis
  2. User research
  3. Sunny-day dialogue design
  4. High-Fi prototype
  5. Usability testing
  6. Pilot
  7. Tuning
  8. Production

• But now: 😊
  • Exception handling is baked into your app for speech
  • Inheritance has it’s fingers all the way into the pie
  • Any reusable modules are still crafted for speech only
  • Structures to handle time-dependencies are baked in

• If we’re to build multi-channel apps from the beginning, we have to:
  1. Identify, define and name and individual business function that can execute correctly for any mode
  2. Abstract mode-specific functionality away from the common business flows
  3. Establish a unified reporting framework that is mode agnostic
Atomic Interaction Providers. Also, bubble gum.
Abstraction

Exception Handling

Defaults Inheritance

Business Logic

Reporting Framework
Exception Handling

SPEECH
- No Matches
- No Inputs
- Agent reqs
- Rpt, Help, etc.

Robo-Chat
- No Matches
- No Inputs

404
- Etc.
- Etc...

Exception Handling
### Reporting Categories

<table>
<thead>
<tr>
<th>Speech Category</th>
<th>Agent Category</th>
<th>Visual Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caller completed services and hung up</td>
<td>Caller completed services and hung up</td>
<td>Completed service</td>
</tr>
<tr>
<td>Caller completed service and requested more services</td>
<td>Caller completed service and requested more services</td>
<td>Completed service and requested contact back</td>
</tr>
<tr>
<td>Caller hung up during call</td>
<td>Caller hung up during call</td>
<td>Ended session before completion</td>
</tr>
<tr>
<td>Caller transferred to an agent</td>
<td>Caller transferred to Tier-N agent</td>
<td>N/A</td>
</tr>
<tr>
<td>Technical problem</td>
<td>N/A</td>
<td>Technical problem</td>
</tr>
<tr>
<td>Caller escalated because of speech error</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Classification problem</td>
<td>Classification problem</td>
<td>Classification problem</td>
</tr>
<tr>
<td>Zombie call</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>
A use case...

[WEB] Julie wants to buy a smartphone so she researches on the web...

...she thinks she's narrowed it down but has a question. Clicks the 'CHAT' button

[AGENT CHAT] Julie asks agent about 3G coverage the difference between data and text plans. Agent answers questions, then asks if he can call to complete the transaction.

[AGENT OUTBOUND CALL] Agent calls Julie and completes the sale

[OUTBOUND IVR] Initiates courtesy call after phone is provisioned. Offers help and Julie selects to have an email sent to her home address about setting up mobile email.

[OUTBOUND EMAIL] Contains instructions for how to setup email on her handset.

[OUTBOUND IVR] Later that evening, Julie gets an email with a link to a survey. She decides to take it, and...

[WEB] Clicks on the link, answers three short questions and goes to bed a happy girl.

[OUTBOUND IVR] Julie not seeing her emails in Outlook, so she calls support and gets a troubleshooting IVR. The IVR understands the problem and says there are 3 possible solutions. Would she like them sent directly to her smartphone?