The Multi-Channel Service Problem: Challenges, Testing, and Solutions

Prepared for
College of Information Science & Technology, University of Nebraska at Omaha
• Introduction to West, and West Interactive
  – History, Technologies, Services

• Introduction to IVRS
  – Voice, Customer Interaction, Call Flow, Testing

• Routing calls to agents
  – Skills, Priority, CTI, Agent Selection, Routing, Testing

• Multi-channel services
  – Web and Mobile, Challenges, Context awareness, Testing

• Challenges, Open problems

• Q & A
• World leading provider of outsourced customer contact solutions and voice related services
• Specializes in: Inbound, Outbound, Interactive, Conferencing
• Headquartered in Omaha
• Over 40,000 employees in North America, Europe, and Asia
  – 140,000+ IVR ports
  – 35 Global Data Centers
• $2 Billion+ in revenue (2007)
• Started with our Inbound Group in 1986
  – Live Operators answering phones & processing calls
• Global leader for delivering Enterprise Contact Center transformation
• Solution Development/Deployment wing of West Corp
• Started in 1989
  – Computers answering phone & processing calls
• Products:
  – Hosted Contact Centers
  – IVR Development
  – Enterprise routing
  – Web, Mobile
  – Proactive Notifications
  – Multi-Channel Services
Platform, Infrastructure, and Automation Group

- **Scope**
  - Systems
  - Platforms
  - Software
  - Products
  - Services
    - used in common across an organization, regardless of project/client
- **Test and validate** client-specific and Client-agnostic solutions
- **Responsible for Test Automation development effort @ WIC**
Types of Testing

- **Component Testing**
  - Behavior of individual components

- **Integration Testing**
  - Behavior of inter-connected components

- **End to End (System) Testing**
  - Behavior of system overall

- **Performance Testing**
  - Load, Stress, Spike, Endurance, Capacity

- **Compatibility test**
  - Test backward compatibility with existing systems

- **Timing testing**
  - Evaluate response times and time to perform a function

- **Security testing**
  - Try to violate security requirements

- **Failure/Failover testing**
  - Test reliability, maintainability & availability across redundant sites

- **Recovery testing**
  - Test system’s response to presence of errors or loss of data

- **Human factors testing**
  - Test with end users.
THE VOICE CHANNEL
Interactive Voice Response Systems

- **IVR - Interactive Voice Response**
  - Technology which allows a computer to interact with humans via telephone
- **Fully automated**
  - Efficient approach for high-volume, common services: e.g., payments, inquiries, voicemails, voting
- **IVRs**
  - Present information to and request information from the caller
  - Output (greetings, announcements, prompts) played either as recordings or as Text-To-Speech (TTS)
  - Input (responses) taken as either DTMF (touchtone) or speech recognition
• More advanced IVRs use natural language parsing, via grammars, to direct flow
  – Ex: “What would you like to do today? You could say I want to pay my bill”
• IVRs may interact with other applications, databases, and services through API hits between prompts.
• IVR may send additional metadata for the call with the final transfer.
• IVR applications are written in markup languages like VoiceXML, CCXML, etc.
  – Applications are similar to web pages
  – Documents are navigated in order
• A system called a Voice Browser is responsible for producing the seemingly procedural nature to the caller
  – The Voice Browser is the caller’s interface to the IVR application, analogous to a Web browser viewing an HTML document.
  – The browser retrieves, and then renders, VXML documents from an internal server.
  – The browser then guides the caller through the document, directed by the flow control and structure of the application.
IVR: Sample Customer Interaction
• Each call in the IVR model is essentially a Finite State Machine
• Caller (FSM) walks through a directed graph of potential states
  – Can be thought of as a decision tree
  – Prompts correspond to States/Nodes
  – Decisions or Transfers correspond to Transitions/Edges.
• Voice is real-time and data-intensive
• Voice recognition is expensive (both money and CPU)
• FSM nature of calls makes Model-Based Testing attractive, but unique business constraints on each state make modeling extremely difficult
  – Ex: No Match, No Input, Retry logic, Failover to TTS
• Due to these challenges, much of IVR testing is done through expensive manual testing
Sample Customer Interaction

Flight Reservation:

• Customer calls into IVR to book a flight. Walks through menu and lands on agent
  – Authentication (Existing Customer)
  – Airport and Dates:
  – Flights & Times:
    • Populated with user profile info: Name, DoB, etc.
    • Seat selection, Upgrade option, Confirms Choice
  – Payment info
    • Credit Card #, Expiration Date, CVV
  – Confirmation #
Sample IVR Flow: Booking a Flight

Greeting → Authentication → Search for available flights → Select flight

Select Payment Method

Select Seat / Upgrade Options → Collect passenger information

Enter Payment Information → Receive Confirmation Number → Call Ends
ROUTING CALLS TO AGENTS
Call Routing

• Commonly used by call centers to direct customers to the appropriately skilled agents
  – Can be as simple as routing a call based on the caller’s ANI or DNIS
  – Can be based on complex routing logic, utilizing back-end systems and many possible destinations

• Process:
  – Routing engine examines Computer Telephony Integration (CTI) data attached to an interaction
  – Routing strategies apply rules to the interaction
  – Based on routing logic, routing engine will deliver the interaction to a route point

ANI: Automatic Number Identification
DNIS: Dialed Number Identification Service
Computer Telephony Integration (CTI)

- Umbrella term for technology that drives PBXs & ACDs
- Originally developed to support “screen pops” for agents in call centers
- Allows data to be attached to a voice interaction
  - Data stored as key/value pairs
  - Attached or manipulated by various systems over the course of an interaction
- CTI data can be queried by routing strategies
Routing strategies

- Rules are applied to interactions based on routing logic
- Back end systems can be utilized
  - Databases, web services, etc. can be queried
- Routing strategy determines how and where the interaction is delivered
- Agent Selection:
  - Once routing logic has been applied to an interaction, the interaction is delivered to an agent
  - The interaction can be delivered to directly to an available agent, or queued
  - Queued interactions will target eligible agents and follow their own queue logic
Testing routing

• Challenges:
  – Interactions need to enter the routing system with proper CTI data
  – Interactions are linked to agent desktop software
  – Many agent desktop products are difficult to automate
  – Synchronizing the entire call flow typically involves multiple major systems (IVR, routing, agent, etc)
Caller goes through IVR, KVPs attached:
Customer: Existing
Skill: Reservation

IVR

Routing strategies

Existing?

Check skill

DB hit, attach as KVPs

Hours of Operation?

Is customer VIP?

Set priority KVP

Queues

Agents

Reservations

Inquiry

Group Bookings

Play closed message and end call
High Level Routing Example

Routing Attached Data (From IVR)
- DNIS
- Skill
- Skill_Level
- Priority

Attached Data Valid

Route to any available Agent

Default Route if skill level is matched

Expand Target Skill level down

Target Timeout

Expand Target Skill level up

Set call Priority

Enter Queue

Target skill / level

Target Timeout

Calls in queue

Yes:

EWT Enabled

EWT < 2 min

EWT Message1

Yes:

EWT < 5 min

EWT Message2

Yes:

EWT < 10 min

EWT Message3

Yes:

Correct Tone Pressed

Route to Voice Mail

No:

Hold Music (90 seconds)

No:

Announcement (if available)

Hold Music (90 seconds)

No:

Voicemail

Yes:

VIM Message

Route to Voice Mail
MULTI-CHANNEL SERVICES
Sample Customer Interaction

Flight Reservation:

• Customer checks for flight availability using Mobile App
  – Search parameters: From, To, Start Date, Return Date
  – Selects flight

• In order to proceed with booking, Customer logs into website
  – Populated with user profile info: Name, DoB, etc.
  – Seat selection
  – Upgrade option

• Customer needs approval for Added fee. Calls into IVR later
  – Payment using Credit card on file
  – Notification sent to Mobile Phone and Email
Breakdown of the interactions

Airline Services

- Book Flight
- Check In
- Flight Status

States
- Airports & Dates
- Flights & Times
- Confirm Choice
- Payment Info

Tasks
- From
- To
- Depart
- Return
- Depart
- Return
- Yes/No
- CC Type
- Number
- Date
Multi Channel Interaction (Without Context Awareness)

Voice Application
- Voice Channel
- User Authentication
- Menu Selection
- Bill Pay Option
- Payment Type
- Credit Card Info
- Payment Processing
- Confirmation
- End

Mobile Application
- Mobile Channel
- User Authentication
- Menu Selection
- Bill Pay Option
- Payment Type
- Credit Card Info
- Payment Processing
- Confirmation
- End

WIC Core Systems

Bill Pay Application
Multi Channel “Context Aware” Interactions

Voice Application
- Voice Channel
- User Authentication
- Menu Selection
- Bill Pay Option
- Payment Type
- Credit Card Info
- Payment Processing
- Confirmation
- End

Mobile Application
- Mobile Channel
- User Authentication
- Menu Selection
- Bill Pay Option
- Payment Type
- Credit Card Info
- Payment Processing
- Confirmation
- End

Context Awareness Platform

Web Services Framework

WIC Core Systems

Bill Pay Application
Objective – Unified Customer Experience

Multi-Channel 
"Context Aware" 
Interactions

Voice
Web
Social Media
SMS
Mobile
Context-Aware
Appropriate Channel
Relevant Information
Empowering
Timely

Seamless Migration from Channel to Channel
Store and Present relevant information across contact channels allowing the new channel to pick-up where the previous channel ended thereby optimizing the communication with the Customer
Introduction to Context Awareness

• Context: “any information relevant to an interaction that can be used to characterize the situation of an entity. “
  – An entity: person, place, or object considered relevant to the interaction between a user and an application
  – Situation: Current state of the user with respect to the environment
  – Human behavior - a sequence of situations

• Metadata stored in the system about the entities
  – User profiles, history, preferences, other information

• Event – Condition – Action Rules
  – On Event If Condition Do Action

• Decision Support Systems
Requirements of the Context-aware Middleware

- Ability to associate relevant information
  - ECA approach is not the answer. Proactive, instead of Reactive
  - Alleviate Redundancy in requests

- Constant need to manage complexity and heterogeneity inherent in multi-channel services

- Common definition: Middleware is software that functions as
  - Conversion or translation layer
  - Consolidator and integrator

- Many middleware solutions but general purpose is:
  - Extending the scope of an application or applications over a network
Design parameters

- Channel Abstraction
- Uniform Menus across channels for same customer experience
- Common modules and APIs
  - Shared Services
  - Collaborative
  - Interoperable
  - Integrated
- Scalable, Enterprise-Level
- Leverage existing components
- Fault-Tolerant, Best Practices based
Architecture

Integrated Development

Applications
- Non voice applications
- Voice applications

Service Platform
- Flight Availability
- Flight Status
- Reservation
- Check In
- Service Orchestration
- Systems Connectivity

Core Systems
- Databases
- Intelligent Routing
- Client Interfaces
- Data warehouse

Infrastructure
- Consolidated Optimized Infrastructure
  High density pooled resources
  (Server, Storage and Network)

Consolidated Business Intelligence

Management & Monitoring
Challenges Involved

- Multichannel integration, resource constraints, mobility, heterogeneity, scalability
- Limitation of Event – Condition - Action rules for complex interactions
  - Complex events or actions have to be broken down
  - Potentially lead to multiple ECA rules being executed over a single trigger
- Acquiring context information and situation perception
  - Fusing heterogeneous context data and capturing user intent
  - Recognizing intent: Users can perform actions in several ways
  - Mitigation of ambiguous context information
  - High-level inference from low level data
- Fault-tolerant Middleware with learning and adaptation
• Inability to associate relevant information
  • Many middlewares designed with ECA approach
  • Redundancy in requests

• Lack of transparency in authentic information
  • User friendliness – user need not worry about technical details
  • Gap between user perception and actual information shared

• Configuration superseding action
  • Limiting user access to make the system robust
  • Privacy management loosely dependent on user interaction

• Granularity of the system
  • Privacy management based on peer bonding and organizational hierarchy
CHALLENGES AND RESEARCH TOPICS
Challenges and Research Topics

- **Proactivity and transparency**
  - Delays, resource utilization, unobtrusive services

- **Heterogeneity and interoperability**
  - Unevenness, incompatibility, h/w, s/w, communication channel

- **Seamless Channel migration**
  - Handoff - vertical/horizontal, data dissemination/acquisition, Routing

- **Context and Situation Awareness**
  - Smart systems, Middleware Development
  - Decision Support Systems, Rules Engines, Policy Engines

- **Authentication and security**
  - Privacy vs. services, cost, agents, availability