

# Speech Recognition Redefines Self-Service

By Doug B. Brown

*Vice President in eLoyalty's Architecture and Call Center Practice Areas*

10.23.2003



OPTIMIZING CUSTOMER INTERACTIONS™

# Contents

INTRODUCTION	3
WHAT'S REALLY CHANGED	4
CHANGES FOR THE BUSINESS ORGANIZATION	7
CHANGES FOR THE INFORMATION TECHNOLOGY ORGANIZATION	9
CASE STUDIES	12
CONCLUSION	14

# Introduction

Traditional touch-tone Interactive Voice Response (IVR) platforms have been around and stable for many years, widely accepted by organizations as a cornerstone of self-service strategy. Speech recognition has also been around for many years, but until very recently had only found very limited and specialized use. But just over the last 18 months, adoption of speech applications has spread from virtual obscurity to the point where one in three large organizations are at some stage of implementation. Now every significant traditional IVR vendor, and a whole host of others — both well-known vendors and new vendors — are touting some version of the speech recognition story as the way to go. What has changed to make this technology suddenly more viable? Is it cost? Is it functionality? Is it scalability, performance, and/or reliability?

These questions each hold part of the answer, but there is much more. Make no mistake, the world in which the traditional IVR platform developed and matured is long gone — and the difference is as evident as the difference between day and night. Many of these differences lie behind the *sudden* viability of speech recognition technology. As different as today's world has become, so too are the opportunities for what businesses can do with speech technology, and how they can do it. This is not your father's IVR — it's better, simpler, easier, and cheaper, and the bottom line is that organizations need to understand what it is and how it is different so they can best take advantage of this exciting technology.

This white paper is not so much about what you can do with speech recognition as it is about what you *can't* do with traditional IVRs. It is about why and how your organization needs to change its approach. If your organization intends to fully achieve the benefits of speech, there is very little about your traditional IVR that carries over to speech recognition. This white paper will provide you with an understanding of what is new with this technology, how the market is changing, and give you some things to think about as you begin the process of planning, designing, implementing, deploying, and maintaining speech recognition as an integral part of your self-service strategy and infrastructure on a whole new playing field.

## What's Really Changed

The traditional IVR has reached a stage of widespread adoption and maturity with little now in the way of significant innovation. Each IVR vendor's platform is quite different in terms of specific tools and techniques, but they are all very similar in terms of what they can do. The processes for defining, designing, and documenting IVR applications do not differ much across organizations. The organization and staffing models different organizations use to manage IVRs look pretty much the same as well. Even the principles of a good IVR user interface have finally reached a degree of maturity.

So the traditional IVR is at the end of its maturity when it comes to what it can do and how well it can be made to do it — nine touch tone keys and hierarchical menus can only go so far — as such it is very good that speech recognition has come along. Can the "status quo" that exists in the IVR world today accommodate speech recognition, or are the opportunities afforded by speech recognition technology so different from traditional IVR that things are going to have to change? If things are going to have to change, is there at least hope of heading into familiar ground rather than uncharted territory? Let's start to explore and find out.

### Historical Paradigm

A traditional IVR tends to be a closed, turnkey, stand-alone type of system that contains in it all the capabilities needed to perform its basic purpose, with integration hooks to back-end business systems. The business technology environment that existed during the birth and evolution of the IVR is what drove this self-contained architecture. *Back then*, there were mainframes and UNIX servers, local area network standards barely existed, and X.25 was the predominant wide area network standard. Industry standards like HTTP, MQI, HTML, XML, SQL, and J2EE simply did not exist.

Back then, organizations didn't really have the option to deploy an IVR as one functional component in architecture composed of many other functional components. There simply wasn't an available marketplace of functional components and tools from reliable vendors. If something was required to get to the solution, it had to come from the IVR vendor. IVR vendors filled any gaps their customers needed with (usually proprietary) technological developments, manifested and packaged as a continuing stream of IVR product features. A symbiotic relationship between the IVR vendors and their call center customer base developed and persisted over the last 10 years even as the technology world at large went through the massive changes wrought by the Internet, the browser, and the Web server.

### New Paradigm

In contrast to the evolution of the traditional IVR, speech recognition has *reaped the advantages* of the technology world of today. The most tangible aspect of this is the Voice XML (VXML) standard. Voice XML is basically an agreed-upon, simple and straightforward way for an application to interact with a speech recognition platform. It defines a boundary line in your speech solution. On one side of the VXML interface are all the speech technology *internals* — such as sampling, digital signal processing, grammar recognition, text to speech, playing audio files, and interacting with the telephony cards. On the other side of the VXML interface is the application business logic — basically what to say, when to say it, what to listen for, and what it means.

It is perhaps easy to underestimate the impact of the VXML standard on the IVR market, and ultimately on your organization. It parallels the impact of HTML on the Internet. The Internet existed many years before most people had ever even heard of it — relegated to very few users and uses. Then when a standard interface (HTML) emerged, and a product that supported that interface emerged (Web browser), the use of the Internet suddenly exploded. Products rapidly evolved on both sides of the interface — browsers on one and Web servers on the other. Today we see the browser as a fairly standardized component. All the diversity is in the Web server end of the solution. The browser is the enabler and the Web servers and Application servers are where the action is.

The global impact and ubiquitous penetration of the Web was predominantly driven by the simplicity of the open HTML standard. The Web development paradigm brought vendor and network independence to distributed applications, and drastically reduced the cost and skills required to quickly deliver powerful solutions. The essence of this model is that powerful Web servers execute application logic (written on any platform from any vendor in any language) while connecting locally to enterprise data, ultimately delivering simple HTML markup over HTTP to a thin browser client, which renders an experience to the end-user.

The Voice XML standard closely parallels the HTML standard – even down to the concept of a browser. Customers already interact with your organization on the Web via a Web browser, and Voice XML enables them to interact with your organization on the phone via a voice browser. Voice XML has taken something that was difficult to use and risky to invest in and transformed it into an easy-to-use tool and a secure investment. The Voice XML standard defines exactly what a vendor is providing in a platform, and defines exactly how to use it – prerequisites to organizations being able to confidently invest in any technology.

The close parallel between Voice XML and HTML also suggests strong parallels between Web self-service and Voice self-service applications – which answers the question of whether speech technology heads into uncharted territory or into familiar territory. Because of Voice XML, any functionality that can be, or is being, provided to customers via the Web can also be provided via voice and the telephone. This is in sharp contrast to the traditional IVR, where the IVR itself had to have specific built-in features to support any functionality that was provided.

## Voice Browsing

In addition to the lessons that can be learned from the parallels between HTML and VXML, there are also some noteworthy differences, for example, the device used for “browsing” and where the browser technology is actually located. In the case of visual self-service, the device – typically a personal computer – contains the browser technology component. In the case of voice self-service, the device – a wired or wireless telephone – does *not* contain the browser technology component. Instead, when a customer accesses your organization’s telephone self-service, you are providing the browser – it is running on a Voice XML platform. The Voice XML platform is the browser for telephone self-service.

The importance of this difference is easy to overlook. In dealing with visual self-services your organization has to deal with the issue of browser vendor neutrality. There are pros and cons to this – you don’t have to pay for the technology the user needs on their computer, but it does cost you more to make sure that whatever they have will work with your Web self-services. Regardless, it is a fact of life on the Web, and your organization probably has invested, is investing, or will invest, in tools, technologies, and techniques to give you the visual Web browser vendor neutrality you need.

Voice self-service doesn’t necessarily face the same cross-vendor compatibility issues, because your organization essentially picks and procures the voice browser. However, this is an important decision to consider while defining your voice self-service architecture. There are benefits to having cross-browser compatibility in the voice half of your self-service too – all the benefits that come with more deployment and operational choices, options, and flexibility. Considering your organization is likely already dealing with cross-browser compatibility on the Web, extending that to cross-voice browser compatibility is a matter of leveraging that Web investment.

## Speech Recognition Redefines Self-Service

There is one other major benefit to adopting a voice browser neutral architecture — being more prepared to accommodate another type of device that your customers and prospects will soon be using in increasing numbers to access your self-service. That is the so-called multimode device — a small wireless handheld with built-in visual and voice browsers. With these devices your organization will have the opportunity to present a combined and coordinated voice and visual interface to the user. When the user talks, the device itself will interpret what he or she said, and when your application responds, the device itself will convert that response into a voice and visuals. With these devices there is no need for an intermediary Voice XML browser between your applications and the user's device. This is the arena of the other major Web-based standard in speech recognition — Speech Application Language Tags or SALT.

Although Microsoft is the major driver behind the SALT standard, it would still be reasonably safe to bet that there will be similar browser compatibility issues given the proliferation of devices and vendors that is bound to occur as adoption takes off. Plus, there will still be plenty of users who simply use their telephone or desktop computer to access your self-services. It is clear there really is a browser compatibility issue to deal with in voice self-service applications. The same best practices that have proven to be successful for dealing with this in Web applications can largely be applied directly to voice applications. Voice XML and SALT allow your voice applications to operate across different/multiple vendors' voice browser platforms, in the same way that Web applications are made to work across different Internet browsers. It is a matter of choice to be cross-vendor compatible or not, whereas in the traditional IVR market that choice has never existed.

### A New Voice

Another major difference between speech recognition and traditional IVR technology is actually not recognition of speech at all — it's the generation of speech from text. If speech recognition is the "ear" of the speech platform, then text to speech is the "voice." The VXML platforms have the ability to say any text while sounding virtually identical to human recorded voice. In fact, on some platforms your text to speech voice can be based on any person's voice. The ability to dynamically compose responses to your customers on the fly makes it easier to make interactions more natural and less rigid. The fact that you no longer need to hire voice talent and record and manage audio files for every single word you need to say to your customers eliminates a major focus and value proposition of traditional IVR development work.

The takeaway from this exploration of *open and standard* is that the Voice XML and particularly the voice browser have successfully achieved a sea change — a large historical shift — in voice self-service, not unlike the HTML and the Internet browser did years before. Just as the Internet market continued to significantly change and evolve even after the first browser vendors came to market, organizations should expect to see some sizable changes in the voice browser market of today — new and better platforms and tools coming from both expected and surprising sources.

So, does speech recognition technology fit snugly into the status quo of the traditional IVR market? No, it blows the doors off — better, cheaper, simpler, and easier. Does that mean your organization is in for a dangerous ride into uncharted territory? No. The Internet and the Web have already charted the territory. Although speech recognition is complex compared to traditional IVRs, speech recognition is not much more complex than a sophisticated Web self-service application. Success is a matter of finding the mix of transition steps that will smoothly transition your customers and prospects off your current IVR applications, and of building on the existing infrastructure you already have for Web self-service.

# Changes for the Business Organization

Businesses have invested heavily in providing Web self-service for their customers in anticipation of great benefits. Yet still today over 90% of all customer service transactions are performed via the telephone. It behooves businesses to find ways to leverage the self-service investments they've already made into the channel their customers naturally use. Speech recognition – in particular the Voice XML voice browser – makes this possible. Much has been written about the features and business benefits of speech recognition – higher self-service completion rates, shorter calls, more satisfied customers, speaker independence, text to speech, voice verification, and natural language. Rather than expounding on these features and benefits, suffice to say that they can be had with *any* vendor's Voice XML speech platform. Hence, this section and the next will instead drill right into two key changes organizations need to address to effectively use speech recognition.

The first key change the business organization has to face in getting what it wants out of speech platforms is that the organization's current IVR requirements and design methodologies will not "scale" to speech applications. Without a change, your speech application may be little more than a glorified IVR application where the customer can say "one" instead of pressing one. The problem with the current methodology most organizations use starts with the fact that it mixes requirements and design into a single effort that almost exclusively focuses on defining an ordered sequence of questions to ask the customer. There is absolutely no element that addresses the most important benefit of speech – listening to your customers in a dramatically new way. Speech technology offers so much more than the traditional IVR technology that when faced with this transition, many organizations are unprepared to step back and focus on what it is they want the application to do and what the application requires from the customer. Instead of building this foundation, organizations often jump to defining call flows.

To get the most out of speech recognition, the business organization needs to start by thinking about what has driven their customers to call them – what needs they have, what events have occurred. Then, think about how the different ways a customer would express his or her need. This process starts to define the requirements for the speech application. What the speech application requirements ultimately need to define is what to listen for from the customer – what words, how broad a vocabulary, and what those signals indicate the customer requires. Unlike traditional IVR technology, the order in which the customer may say things is not an issue with speech applications.

Although this approach is totally contrary to the usual approach for IVR applications, it should have a familiar ring to it for the call center organization. After all, getting to the root of why customers are calling is not an unfamiliar activity for many call center businesses. Getting to these speech application requirements may be more a matter of making some adjustments to some existing processes, than of buying into a completely new and separate process.

## Speech Recognition Redefines Self-Service

After the listening requirements, next is to think about how to speak to the customer. Much of the speaking will be about fulfilling the reason they have called – “Your order shipped last Tuesday,” or “I’ll need to transfer you to accounting, please hold.” Some of the speaking will be to ask them for more information in order to get to the point that fulfillment can begin – “Ok, I can help you with that. What is your account number and PIN?” Lastly, some speaking will be to clarify your understanding of what they want – “You would like the status of your order?” Focus the requirements effort on defining, recognizing, and handling these kinds of patterns:

- Under what conditions is what the customer wants clear?
- Is clarification called for?
- When should options be explained?

The result of the requirements process for speech applications should focus little on the dialog flow and sequence, and much on dialog states: inventorying the reasons why customers will call, how they’ll indicate them to you, and what the fulfillment actions are. It doesn’t help to diagram the 24 different flows a customer could follow to provide the three pieces of information it takes to understand and fulfill his or her need. Voice XML voice browsers aren’t designed to work this way, and in fact are remarkably good at naturally supporting many different paths (flows) to the same end point. Without these adjustments your requirements process is going to expend time and energy defining constrained dialogs, rather than focusing on applications that let the customers use their own order and pace to tell you what they want.

The second change business organizations have to face is simply letting the speech platform implementation migrate over to the Information Technology organization, where the rest of the self-service implementation responsibility resides. Speech applications demand a lot more out of developers than IVRs did, and it will be difficult for organizations to justify, secure, and maintain the necessary skills, tools, methods, and talent to implement speech recognition within a call center silo when many of these resources already exist in the IT organization.

# Changes for the Information Technology Organization

Like the business organization, there are two key changes Information Technology (IT) organizations need to address to effectively use speech recognition. The first change IT has to face is simply bringing the speech platform all the way into the fold of the Web self-service organization. With the traditional IVR, the technology resources are usually part of the call center business organization, rather than part of the IT organization. Given the self-contained nature of the IVR, there is no particular benefit to having it otherwise. But that's not the case for speech platforms — or more accurately it doesn't have to be the case.

To be sure, designing and implementing voice applications are different, and more challenging, than designing and implementing the Web applications. Dialog as an interface to information and services presents many unique design challenges that demand expertise and ongoing refinement. While even the worst visual interfaces are at least somewhat usable, all but the best voice interfaces can deeply frustrate customers and thereby be rendered useless. While speech recognition makes it possible to produce outstanding interfaces that quickly and efficiently deliver self-service access to information and services, achieving this is still complex and demands some specialized expertise and a deep commitment to quality. But none of these points makes the case for keeping speech in a separate organization — these are qualities and capabilities for which most IT organizations strive. In the more sophisticated self-service applications on the Web, you can actually find many of the capabilities you would need in a voice self-service application. Again, the demands of voice application development are not as foreign to the Web development organization as they might appear to a group used to working with the traditional IVR.

The second key change for IT is establishing an application architecture for the voice applications, and then choosing the application development tools. The other components required for successful implementation — project management, development life-cycle management, quality assurance, and support — work as well for voice as they do for Web. However, voice being different from visual begs the question of what different and/or new tools the organization should get. To answer this question requires a little understanding of the mechanics of a speech application that works with a Voice XML voice browser.

A great many of the functions that a speech application needs to perform are no different than what a Web application needs. This is due in very large part because Voice XML makes the speech platform a browser that communicates with your Web application over HTTP — no different than Internet Explorer (IE) or Netscape (NS). All the parts of a Web application — HTTP posts and gets, form variables, cookies, session management, session state, and page caching — are just as much a part of a voice application. Keeping track of what someone has in their shopping cart is no different than keeping track of what they have said over the course of a dialog.

The real difference between voice and visual is presentation. Regardless of anything else, the application architecture should reflect this fact. For example, a self-service shopping application may *tell* a shopper what's in their cart differently than it would *show* them, but it would obviously want to use the exact same shopping cart in either case — otherwise, there would have to be some benefit in buying or building a different shopping cart for voice.

So the question of whether or not a special voice application development tool is needed is limited to the presentation layer. In trying to answer this question, IT should consider whether and how their existing Web architecture and tools are already dealing with presentation differences. For instance, although a desktop computer and a handheld device are both visual interfaces, the form factors are so different that adjustments in the presentation must be made to enable customers to view and browse your Web site correctly. If you have a strategy for this (there are many possibilities) then explore how that strategy can be extended and used to deal with voice. If you don't have a strategy, then consider how you're going to deal with not only voice/visual differences, but also with different visual form factors too.

## Speech Recognition Redefines Self-Service

After considering your architecture and your strategy for managing different presentation media, it comes down to what specialized voice development *tools* might be called for. There are a range of tool categories to consider – turnkey speech applications, speech application templates, speech objects, speech development suites, and speech testing/tuning tools. Turnkeys, templates, and objects can be of great benefit, to the extent that you can find ones that are appropriate to your architecture. Speech testing and tuning tools are likewise beneficial because the tools you might have for testing visual interfaces aren't likely to be suitable for testing a voice user interface.

The toughest decision is around the application development suite – whether a new one is really needed, or the one(s) you already have is suitable for the job. The tool suite you already have for the Web most likely can do the job of developing an application that can provide Voice XML to the voice browser. The reason to consider a specialized voice development suite is if it makes the development effort easier. There is no right answer, just getting to the point where an informed choice can be made that fits your IT organization, and can provide what the business organization wants.

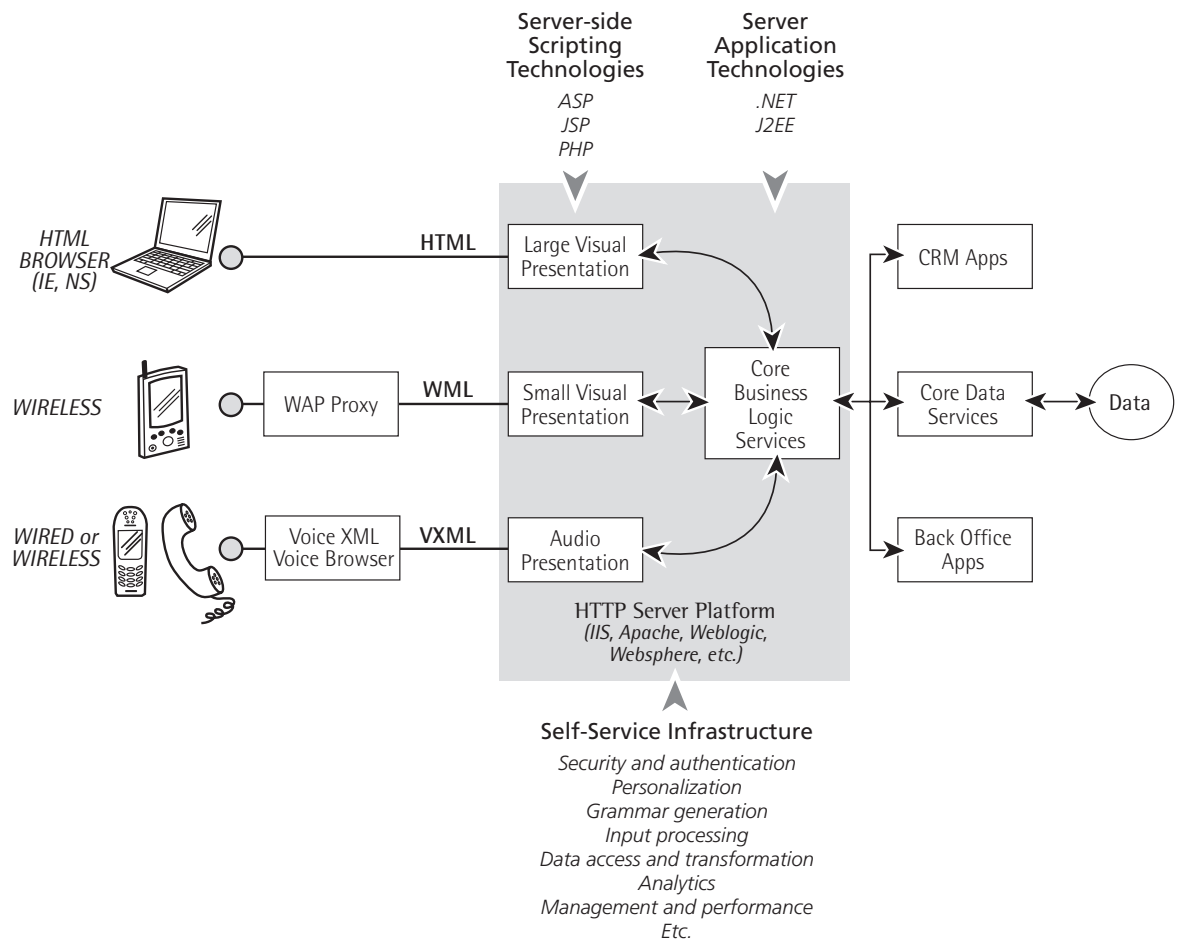
Most speech application development tools you're likely to come across today have a value proposition centered on a Graphical User Interface (GUI) that allows you to visualize the application. Visualization is also a common trait in many other types of application development tools. What You See Is What You Get (WYSIWYG) tools make adjustments to the presentation by literally moving the visual components around on the screen. Workflow tools draw flow charts of how work progresses through a process. Voice tools also draw the dialog as a flow chart, but depending on what your business customer is asking for, this may or may not be helpful.

If business wants to dictate the order and pace of the dialog, then using a flow chart as the application is useful. If business wants to let the customer dictate the order and pace, then using a flow chart for the application won't actually work, because when the customer is driving the order and pace of the dialog almost any order and pace is possible. For example, a customer could say "I want the status of order number 1234," or "I want order status," or "I want status." If an application is going to support this level of *listening* to the customer, then the important state is when it has the three things it needs from the customer to do fulfillment – the information they want (status) about a specific (identifier) type of business object (delivery). There is no point in drawing every possible sequence and combination for getting to this state – such a flow chart is useless at best; costly, error prone, and difficult to maintain at worst.

Without some familiarity with Voice XML voice browsers, supporting this level of listening flexibility in a speech application may appear very complex, but in reality it's not. The grammar capabilities of voice browsers can easily handle the listening task. The task of tracking and reacting to dialog progress (e.g., what they have said so far, what else is needed from them) resides in your application, and is easily handled with a state-oriented design. Conversely, if your application is designed around flow chart rather than around state, this type of flexibility is very difficult to implement. Implementation is easy with the right application architecture, and difficult with the wrong application architecture.

So, just because a visual (WYSIWYG) tool makes sense for visual presentation, it does not automatically mean a visual (flow charting) tool makes sense for voice presentation. This may seem like a contradiction – if visualization is good for one it should be good for the other. But these two types of tools are visualizing two completely different things. The typical voice tool is trying to visualize the order in which the customer can say things, whereas with a WYSIWYG tool you aren't trying to visualize the order in which the user looks at the items on the page.

To recap, the major influence on this choice of what tool to use for developing speech applications is whether or not business wants the customer to be able to set the order and pace of the dialog. Most businesses want this – it is one of the fundamental elements of the value proposition of speech. A final consideration is that even if the organization wants to let the customer set the order and pace, the customer might not want to do so. To prepare for this contingency, the application needs to be able to handle both conversation structures. A state-machine architecture can scale down to the simplicity of a flow chart, but a flow chart can't scale up.



## Case Studies

eLoyalty works with clients to carefully evaluate the business objectives that can be enabled through speech recognition, and how best to implement the customer-facing solution. In the following examples of our clients' experiences, we'll take a look at a couple of distinct possible starting points for launching into speech technology. Each illustrates a different set of reasons why the time was right for these organizations to begin its transition.

### Enhancing Existing Voice Self-Service – Healthcare Industry Experience

Our client, a multi-billion-dollar healthcare provider, had experienced significant growth through acquisition. These acquisitions enabled them to provide a rich service offering including:

- Network-based health and well-being services
- B2B transaction processing
- Consumer connectivity
- Technology support services to large employers and health plans

A side effect of this rapid growth was a self-service architecture that consisted of numerous IVR applications interfacing with siloed legacy systems. Customers were required to navigate a maze of complex menus to gain access to information. It became clear that their self-service capabilities were ineffective and did not provide value to their customers.

To address these shortcomings a phased implementation approach was developed. The first phase sought to reduce application complexity, establish consistency across applications, and efficiently and accurately ascertain the customer's need. Replacing disparate network routing, using a single voice actor for prompting, and defining a common user interface accomplished these goals. The second phase expanded their self-service capabilities to enable providers to inquire about the status of a claim and verify eligibility using advanced speech recognition.

The most notable outcomes of this initiative were:

**Increased automation rates.** Our client has experienced an increase in automation rates, resulting in a reduction in CSR-based call volumes.

**Expanded current self-service capabilities.** The advanced speech recognition capabilities have resulted in a high satisfaction rate among providers.

Through the adoption of a standards-based architecture, reducing application complexity, and defining a common user interface, our client can effectively meet the needs of its customers.

## Replacing Traditional IVR with Voice Self-Service – Transportation Industry Experience

Our client, a leader in the shipping industry, maintains a massive fleet of vessels and containers enabling the organization to service ports across the globe. The shipping industry is highly competitive and providers often differentiate themselves through value-added services and customer service. Our client's existing touch-tone IVR application platform was reaching end-of-life, was unstable, lacked functionality, and experienced significant downtime. All of these contributed to a poor customer experience and increased agent call volumes. An assessment of the organization's self-service infrastructure revealed the need to quickly replace its existing IVR platform, reduce customer service representative (CSR)-based call volumes, and define an architecture that leveraged existing Web infrastructure investments.

To address these challenges the client rapidly replaced its premise-based IVR with a hosted Voice XML application. This approach delegated systems management functions to the service provider, enabling the client to focus on application design and development. In addition, defining an architecture based on industry standards like Voice XML provides greater flexibility and deployment options. This includes converting the current solution to a premise-based offering.

The most notable outcomes of this initiative were:

**Achieved automation rates of 60%.** By replacing its antiquated system with a speech-enabled interface the client found customers preferred to use the automated system, for routine inquiries, rather than talking with a live agent.

**Enhanced usability through a flexible interface.** The system was designed with the user in mind. For novices, context help or shortcuts were provided to educate them on how to use the system more efficiently. For the expert, short phrases could be uttered to quickly bypass structured menus and quickly retrieve information.

**Standards-based architecture.** Adopting a standards-based architecture provided a strong foundation and flexibility to quickly add functionality and more complex transactions.

Defining a self-service architecture based on industry standards and hosting the initial deployment has enabled our client to achieve higher levels of customer satisfaction and differentiate itself in a highly competitive industry.

# Conclusion

Speech recognition is real and viable, and Web-based standards have made voice browsers a component of the Web architecture. The potent mix of hot technology and a crowded vendor roster guarantees a fast-changing speech market. Many "rules" that applied for traditional IVR are already on the way out. Look to the Web for the new "rule book." As a result of these changes, organizations should shift investments from traditional IVR to voice browser platforms, abandon the silo approach they've used with IVRs, and leverage the approach they're using with the Web.

eLoyalty can bring its expertise to assist you in the selection of best-of-breed speech-recognition vendors or the traditional IVR vendors. Our Speech Recognition Solutions Methodology includes both a Business Track and an IT Track, and our approach includes the following steps:

1. Speech-recognition requirements confirmation
2. User interface and Business design for customer directed dialog
3. IT application architecture and design leveraging Voice XML voice browsers
4. Vendor evaluation and selection (as required)
5. Application scripting (vendor specific as well as independent)
6. Integration with Web infrastructure for consistent back-end data sources
7. Integration with CTI and call routing
8. Development, testing, and implementation management
9. Personalization and customer segmentation
10. Application tuning

eLoyalty is uniquely positioned to help companies identify and harvest the value of this new, groundbreaking paradigm. We have assembled a powerful combination of customer experience design, Internet and CTI technology, cross channel integration, assessment and implementation methodologies, and deep subject matter expertise. We look forward to talking to you to bring these resources to bear to help you create a new paradigm of interacting with your customers through self-service.

## Keys to Speech Recognition in Self-Service

	CHANGES	BENEFITS
BUSINESS	<ul style="list-style-type: none"> <li>• Adopt new requirements and design processes to get what you want out of speech. Focus on customers' reasons for calling, not on the order and pace of the dialog.</li> <li>• Let the IT organization incorporate speech as a component of its self-service architecture.</li> </ul>	<ul style="list-style-type: none"> <li>• Increase usability over existing IVR applications</li> <li>• Provide more self-service capabilities</li> <li>• Increase self-service usage</li> <li>• Decrease call length</li> <li>• Improve security (voice verification)</li> <li>• Increase channel consistency</li> </ul>
INFORMATION TECHNOLOGY	<ul style="list-style-type: none"> <li>• Incorporate voice self-service as an integral part of your self-service architecture.</li> <li>• Leverage existing Web skill and infrastructure investments. Most of the existing Web infrastructure works for voice, including application servers, security and authorization, transaction management, monitoring and tracking, reporting, personalization, content management, and business rules.</li> </ul>	<ul style="list-style-type: none"> <li>• Reduce silos between voice and visual self-service. Integration is no longer a separate effort for speech. Speech is just a presentation layer. Pick tools and techniques accordingly. People may use a tool like Front Page to manage the details of presentation, but nobody uses a tool like Front Page to build a Web self-service application.</li> <li>• Roll out more functionality</li> <li>• Increase Architectural Consistency</li> <li>• Leverage different operating models – hosting, or premise, or both – to mitigate the risks associated with investing in voice self-service. Operating costs, internal skill sets, and implementation timelines are all factors that can be "optimized" at different phases. In an initial phase it may be more advisable to utilize an outsourced speech platform vendor initially in order to focus on developing the voice application and quickly transition some call volume from your existing IVR platform. In later phases, as adoption rates increase and your self-service strategy is refined, it may be better to purchase and deploy speech platforms in-house.</li> </ul>

### About the Author

#### DOUG B. BROWN

Douglas B. Brown is a Vice President in eLoyalty's Architecture and Call Center Practice Areas, specializing in enterprise CRM architecture, Internet architectures, speech recognition and mining, project management, system design, relational database systems, and telecommunications. With over 20 years of industry and consulting experience, his tenure includes work across numerous industries, including automotive and high tech manufacturing, media, publishing, banking, insurance, wireless telecommunications, travel, and consumer products.



OPTIMIZING CUSTOMER INTERACTIONS™

**CORPORATE HEADQUARTERS  
U.S.A.**

150 Field Drive  
Suite 250  
Lake Forest, IL 60045  
Phone: 877.2ELOYAL  
Fax: 847.582.7001

**CANADA**

eLoyalty (Canada) Corporation  
90 Allstate Parkway  
Suite 301  
Markham, Ontario L3R 6H3  
Phone: 905.944.5500  
Fax: 905.944.5501

**UNITED KINGDOM**

eLoyalty (UK) Limited  
Albany House  
Market Street  
Maidenhead  
Berkshire  
SL6 8BE  
United Kingdom  
Phone: 087 0735 9526  
Fax: 087 0735 9527

## About eLoyalty

*eLoyalty is a leading management consulting, systems integration, and managed services company focused on optimizing customer interactions. With professionals in offices throughout North America and Europe, eLoyalty's broad range of enterprise Customer Relationship Management (CRM)-related services and solutions include creating customer strategies; defining technical architectures; selecting, implementing, and integrating best-of-breed CRM software applications; and providing ongoing support for multivendor systems. The combination of eLoyalty's methodologies and technical expertise enables eLoyalty to deliver the tangible economic benefits of customer loyalty for its Global 2000 clients. For more information about eLoyalty, visit [www.loyalty.com](http://www.loyalty.com) or call 877.2ELOYAL.*