



VoiceObjects

Interact Naturally With Your Callers

A White Paper on Natural Dialog Management

www.VoiceObjects.com

Table of contents

Introduction	3
Dare to Become More Natural	3
Barge-in	4
Global commands	5
Random Prompting	6
Multilingualism	6
N-Best Result Handling	7
Take It Personally	7
Mixed-initiative dialogs	7
Implicit correction	9
Natural pronunciation	9
Caller-adaptive pronunciation	10
Make Your Dialogs Natural Dialogs	11

Introduction

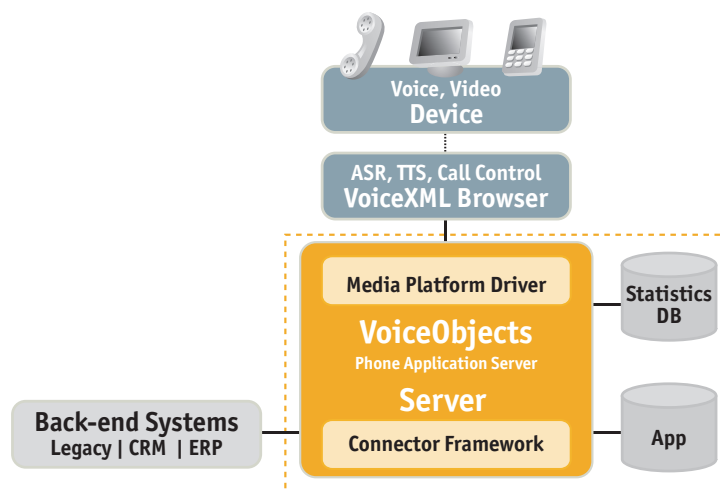
Interacting in a natural, human-like way on your phone portal is critical to your callers. Why? Because humans have spent their entire lives communicating using speech, developing and deploying common communication patterns without consciously thinking about them. When customers call your company to get information or perform specific tasks, they expect to be able to use the speech pattern they use every day: natural, spoken language to interact with each other; dialogs based on turn-taking, in which both parties can control the direction of the conversation. While callers should not be made to believe they are talking to a real person when calling your IVR, they should be able to use the service without having to make too many sacrifices in terms of talking to and understanding the system.

While the technology of current automated natural dialog systems does by no means match the communication skills of human beings, it can do much better than many systems deployed today. This paper introduces *Natural Dialog Management* capabilities of VoiceObjects, a variety of features supporting natural, human-like man-machine interaction. Learn how to easily enhance existing applications with features that can increase caller satisfaction, task completion rates, and thus revenue generated with your self-service voice applications.

Learn how to make your IVR interact naturally with your callers!

Dare to Become More Natural

Before delving into the various features of natural interaction, the following picture aims to clarify the roles of the different components required in a complete phone self-service environment, in particular the role of the media platform vs. that of a phone application server such as VoiceObjects Server:



The same way as voice applications simulate human communication patterns, this environment can be compared to the human body: Think of the speech recognizer (ASR) as the “ear” and speech synthesis (TTS) as the “mouth”. The media platform acts as some kind of “skull” that holds these pieces together. The phone application server, on the other hand, acts as the “brain” of the entire system, thus being crucial for managing the interaction with the caller, mediated through the (mobile) device. It instructs the media platform and ASR engine to “listen” to what the caller says and receives a raw string representing the utterance. The interpretation of the utterance in the context of the overall dialog is done by the server. It also instructs the media platform on how to talk to the caller, what to “say”. In addition, it keeps track of the ongoing dialog, reacting intelligently to the caller’s actions. Infostore, the logging component of VoiceObjects Server, finally acts as the “long-term memory”, remembering all interactions, so that the system performance can later be analyzed using VoiceObjects Analyzer.

The following sections describe how any voice application can be enhanced to interact more naturally with callers, exploiting the *Natural Dialog Management* capabilities of VoiceObjects.

Barge-in

Humans are able to speak and listen, at the same time. Listen for the dialog partner to interrupt, to “barge in” while the other one is talking. While it might be considered impolite to interrupt someone, it is a crucial means of human-human communication, required to be able to keep control of the dialog, to intervene, to speed up; in the end, it might just as well be impolite to not pay attention to the desires and behavior of your dialog partner if you keep on talking and talking.

The same paradigm can be carried over to the world of automated speech systems. The ability to listen to the caller while the system is speaking, or (viewed from the perspective of the caller) the ability to interrupt the system while it is speaking is called **Barge-in**. This is a property that can be enabled and disabled for each individual system prompt. Barge-in can also be enabled globally for all prompts, in which case it doesn’t need to be configured for each individual prompt.

In some cases, it might be required to turn barge-in off, so that the caller is forced to listen to a prompt, be it a commercial, legal, or any other form of announcement. It can even make sense to first allow barge-in, but after a while or a sequence of events switch it off. Keep in mind that the system detects barge-in by analyzing the sound that is transmitted over the phone: if it isn’t well tuned, a noisy environment can just as well cause the system to interrupt a prompt, even though the caller didn’t (intend to) speak. In such scenarios, after a series of so-called *No Match* events (the system could not understand what the caller said) barge-in can

dynamically be turned off, so that the overall performance of the system is enhanced. If the media platform supports it, barge-in can also be made sensitive to speech input only, ignoring any non-speech sounds. Or it can be configured such that it only allows interruptions if the caller speaks an utterance that makes sense at this specific point in the dialog flow. Setting this is also supported by VoiceObjects, through the *tuning parameter* **Barge-in Type**. Again, the type can be set globally, or individually depending on the dialog step.

Lastly, barge-in can be a feature of personalization, activated for frequent callers only, those that know the system and want to quickly reach their goals. For novice callers, unknown to the system so far, it can be switched off so that they are forced to learn about the basics of the system before they can interact with it more quickly and more intuitively.

Global commands

Voice applications can be designed as a directed call flow or based on menus to inform the user about the functionalities represented by the system. In the latter case, **global commands** play an important role to enable the caller to control the system and to easily navigate in the given menu structure. VoiceObjects supports any flavor of global commands, be it for pure navigation purposes (“go back”, “repeat that please”, “skip this question”), asking for help (“explain”, “help”, “What can I say?”), or activating any other kind of out-of-the-flow action (“talk to an operator”, “main menu”, “go to news”).

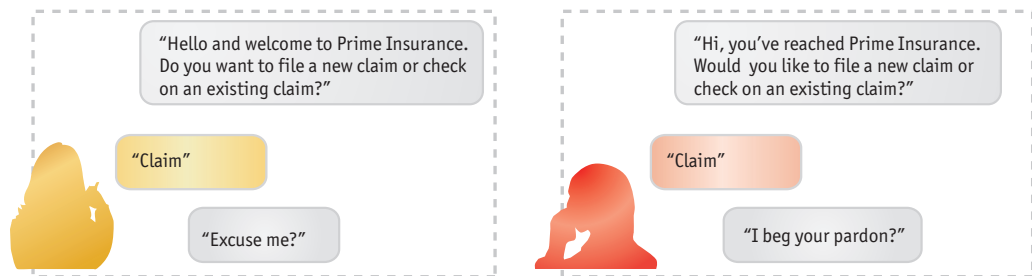
Global commands can, due to the object-oriented nature of VoiceObjects, be defined globally throughout the entire application. This saves development time as these commands do not have to be configured for each individual dialog step. They can, however, be deactivated wherever required, e.g. to avoid conflicts in recognition. Imagine a banking portal that asks for the recipient for money transfer, the caller saying “Transair”, which gets misrecognized by the system as “transfer”, so that the sub-dialog starts from scratch. This can easily cause frustration; why would a caller say “transfer” while being in the middle of it?

To avoid misrecognitions, VoiceObjects also supports **confirmation** steps before performing a task upon recognition of a global command. This can also be based on the confidence values returned by the ASR engine, i.e. the likelihood that a specific command was understood. If callers say “buy” in a trading application (even though this might not be a valid command at that point), you wouldn’t want to hang up on them just because you understood “bye”. Being able to make sense out of the callers’ utterance, or to ask back and then potentially return to the last dialog step is essential in these cases.

Random Prompting

Human language, by means of their syntactic and semantic rule system, allows for expressing one and the same thing in a multitude of different ways. In other words, humans can mean the same thing but use completely different words. This is an important factor of naturalness in language, which can be mimicked by voice applications. If used, it helps to do away with the monotonous and robot-like systems unfortunately still deployed widely today.

VoiceObjects provides a feature called **random prompting**. By recording different variations of the same prompt, VoiceObjects Server can be made to select different variations randomly at call time, resulting in a more natural call experience. Variations should be provided for prompts that can potentially occur more than once in a dialog. This is typically the case for main menu prompts, prompts asking the caller to repeat their last utterance (in case of *No Match* or *No Input* events), or even welcome prompts being played at the beginning of the call – callers might call into your system more than once, maybe even several times a week.



Multilingualism

Countries with more than one official language face the challenge of providing services in all required languages, which multiplies the required effort for any kind of self-service, not only automated IVR. In voice applications, multilingualism can easily be achieved by providing prompt recordings for all required languages, and potentially deploying TTS and ASR systems running in these locales. VoiceObjects allows designers and developers to manage the prompt definitions using the *Storyboard Manager* tool, which also has in-built functionality to prepare an application for a new language. With this, the task is simplified to providing translations of the prompts, while the application development effort is reduced to a minimum.

The required language can either be selected during the call (typically at the beginning, or switched while the call is ongoing), or it could be made configurable on a Web site accompanying the IVR service offering, taking personalization one step further.

N-Best Result Handling

Speech recognition is a statistical endeavor. ASR engines return likelihoods of recognition hypotheses, and the most likely result is usually taken to be what the caller has said. In situations of conflict, where there is more than one possible result due to phonetic similarity, a voice application should be capable of resolving the conflict much like humans do: namely by asking back.



A voice application should have access to the “n-best” hypotheses of the ASR engine together with recognition confidence values, so that it can decide whether to take the hypothesis with highest confidence as the result, whether to take one with lower confidence because it knows that it might be more likely due to knowledge on past caller interaction, or whether to start a clarification dialog. In the end, this is what humans do: analyze what they have heard and draw conclusions on what the dialog partner meant, taking into consideration knowledge of language, the communication situation, the domain, and potentially of the dialog partner themselves. If a caller is already identified and the CRM system knows that they have asked for Austin in the past, the likelihood of Austin could be rated higher than that of Boston, even though the ASR confidence values suggest a different interpretation.

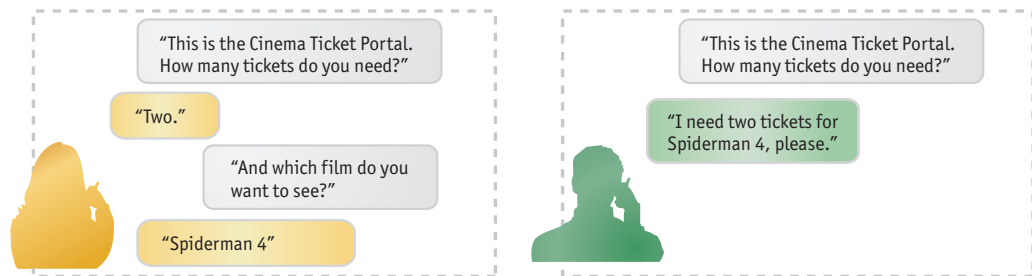
Take It Personally

While the features described so far can be considered basic and must-haves for any voice application, there is a variety of more advanced functionalities that can increase the naturalness even more, help the application to interact more intelligently with callers and treat them individually, even adapting to them in ways of style, vocabulary and content. The next sections describe these features of personalization in some more detail.

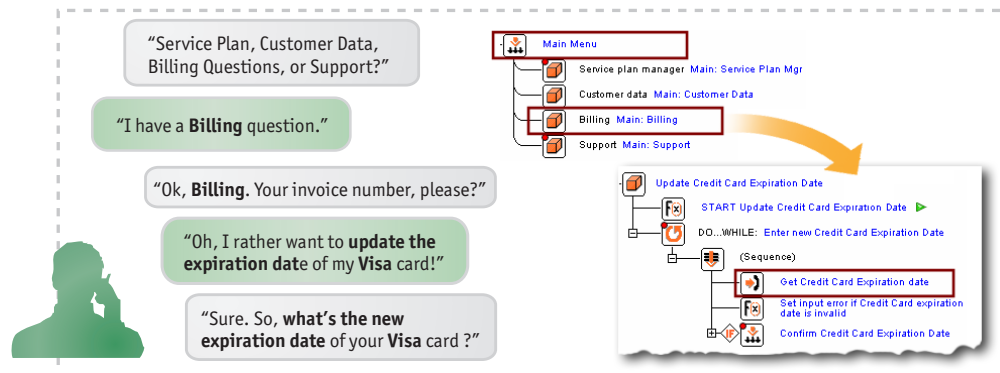
Mixed-initiative dialogs

As mentioned before, there are several types of dialogs that can be distinguished. Among them are **directed dialogs** that follow a predefined call flow definition with the system usually asking a variety of questions to collect information and little interactivity on the part of the caller.

These dialogs make sense for applications such as voice surveys. With information or ticketing systems for traveling, entertainment, or government, directed dialogs can help inexperienced callers to reach their goals, but frequent callers might prefer more interactivity and so-called **mixed-initiative dialogs**. These allow experienced users to by-pass certain steps and interact with the system in a quicker and more efficient manner. While directed dialogs usually ask for one item of information at a time, mixed-initiative dialogs could either follow a “How can I help you?”-approach, or start as a directed flow but allow responding with more information than was asked for. VoiceObjects also refers to this as **multi-slot recognition**:



While frequent callers can benefit from multi-slot dialogs of this kind, mixed-initiative can be taken one step further to also allow branching out to completely different areas of the application. This means that callers do not respond to the current question, but instead ask an entirely different question themselves which already includes one or more pieces of information:

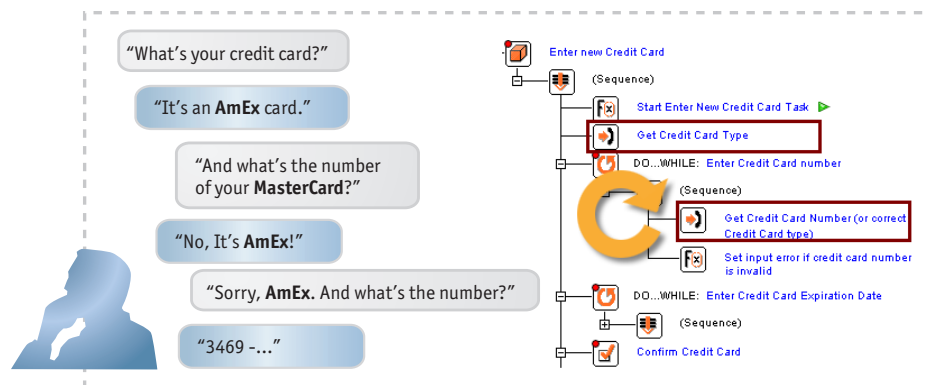


While being in the sub-module for Billing, the caller decided in this sample call flow to switch topic and “jump to” another module by responding to the billing question with information not asked for, using free and natural speech. VoiceObjects Server can detect this and lead the caller to the desired sub-module, skipping any questions on information already provided.

Implicit correction

Misrecognition is human. It occurs among humans just like it does between a caller and an automated voice application. The human communication system has developed effective means to compensate for recognition problems, and clarification dialogs help to reduce misunderstandings where required. If a voice application responded with a simple “Excuse me?” instead of “I am sorry, I didn’t understand you. Please repeat your last utterance”, callers wouldn’t even realize that there was a recognition problem. They are used to simply repeat what they said, with different or simpler words or maybe clearer pronunciation, which results in better recognition both with humans and with automated IVR systems. Becoming more natural can be that simple.

Another way of reducing misunderstandings is to enable the caller to “barge-in” with corrections whenever misrecognition was detected. Instead of letting them actively confirm each and every input collected, a voice application could implicitly confirm an input and in turn allow the caller to **implicitly correct any misrecognition**. The following example illustrates this:



As you can see on the right-hand side of the picture, callers can correct the credit card type while being asked for the number in the next step, and VoiceObjects allows them to loop within this dialog step until the ASR engines correctly understands and the dialog can continue.

Natural pronunciation

Among the criteria for evaluating synthetic speech are traits like **naturalness**, **pleasantness**, and **intelligibility**. While intelligibility is no longer a problem in modern speech synthesis systems, naturalness is still a topic of research and one of the big challenges in automated speech systems. With naturalness often comes pleasantness in the sense that as the system sounds more like a human, it is more pleasant to listen to it. Natural pronunciation is usually a challenge for applications that use pre-recorded speech to read back sequences of digits, such as in phone numbers, PINs, or account numbers. Instead of reading back each digit with the

same tone and pitch, the system can deploy algorithms that apply a natural melody to the entire number. As a result, the digits are grouped the way humans would group them and the system's voice goes up before breaks and goes down at the end. This kind of naturalness can also yield better intelligibility, as callers are not distracted or even irritated by unexpected, robot-like intonation.

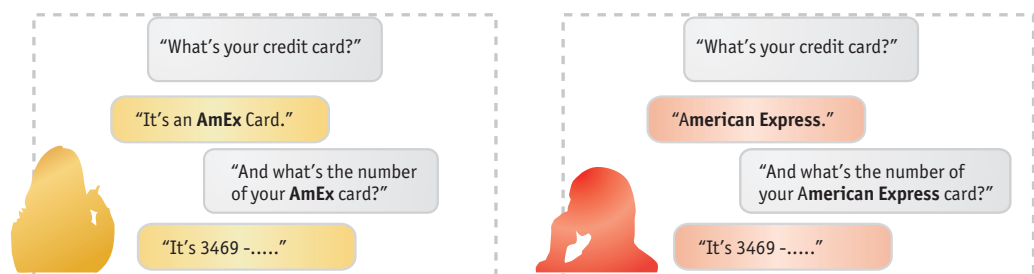
Caller-adaptive pronunciation

It is an important human trait to be able to adapt oneself to situations in various different ways. One aspect of this is adapting one's speaking style to that of the dialog partner. You wouldn't talk to your boss the same way as you talk to a friend.

As language is such a multi-faceted system, adapting language to a specific communication situation can encompass areas such as vocabulary, syntax, speed, style, and accent. As voice applications use speech to interact with callers, making them **caller-adaptive** in these areas can again improve the naturalness of the overall experience and thus drive acceptance and positive attitude of callers towards the system. As an example, think about how you would speak the last digits of your social security number. If it was "5724", would you say "five seven two four" or "fifty-seven twenty-four"? If your phone number was "6105622900", would you say "six one zero five six two two nine zero zero", or "six one oh fifty-six twenty-two nine hundred" or "six one oh, five six twenty-two, nine hundred"? If somebody was to repeat that number to you, you would experience that it is more difficult to understand your own number if your dialog partner uses a different way of pronouncing it.

Using VoiceObjects, voice applications can be taught to listen to the caller's speaking style and adapt accordingly, again improving intelligibility. This can even help to avoid confusion with synonyms that the caller isn't aware of. If with DSL and Internet you mean the same thing in dialogs on your telecommunications self-service portal, you might confuse callers that are not aware of this. If they call in and talk about problems with their Internet and the system responds with "I understand you have problems with your DSL, is that correct?", you are most likely to get a "No!".

Or check out the following example:



Make Your Dialogs Natural Dialogs

VoiceObjects believes that natural interaction is essential in modern speech applications. This paper has shown how *Natural Dialog Management* powered by VoiceObjects can help to

- **Increase caller acceptance ...**
... by deploying mechanisms of human language that everybody is accustomed to, making the call experience more natural and more enjoyable
- **Increase caller success rates ...**
... by reducing the number of misrecognitions through context-sensitive error handling strategies and increasing intelligibility of the system
- **Reduce call durations ...**
... by making each call a personalized experience using intelligent, adaptive dialogs that take caller preferences and history into account and lead the caller more quickly to the desired goal

While only VoiceObjects provides out-of-the-box support for all natural dialog features mentioned in this paper, there is no need to despair if you don't run your services on VoiceObjects yet. Learn how VoiceObjects has tools to migrate legacy IVR systems and enrich your existing applications with the full power of Natural Dialog Management. Go to www.VoiceObjects.com to find out more.



Corporate Headquarters:

VoiceObjects Inc. 1875 South Grant Street, Suite 720 San Mateo CA 94402 Phone: (650) 288-0299 Fax: (650) 525-9414

EMEA Headquarters:

Germany Friedrich-Ebert-Strasse 51429 Bergisch Gladbach Phone: +49 (2204) 845-100 Fax: +49 (2204) 845-101

www.VoiceObjects.com

VoiceObjects is a registered trademark of VoiceObjects, Inc. Any other trademarks, trade names or service marks mentioned in this document belong to their respective owners.

The material presented herein is based upon information that we consider reliable, but we do not represent that it is error-free and complete. VoiceObjects does not make any representation or grant any warranty with respect to such material, and the distribution of such material shall not subject VoiceObjects to any liability.