

THINKING OUTSIDE THE BOX
IN CUSTOMER SERVICE

I don't receive Health Benefits, You don't have to pay FICA, nor do I Take Days Off

Volume 7 Issue 77
1/31/2011

Inside this issue:

Reasons To Collect Data Beyond the Application

If you look up the definition of IVR in Wikipedia it states:

Interactive Voice Response (IVR) is a technology that allows a computer to interact with humans through the use of voice and [DTMF](#) keypad inputs.

In telecommunications, IVR allows customers to interact with a company's database via a telephone keypad or by speech recognition, after which they can service their own inquiries by following the IVR dialogue. IVR systems can respond with prerecorded or dynamically generated audio to further direct users on how to proceed. IVR applications can be used to control almost any function where the interface can be broken down into a series of simple interactions. IVR systems deployed in the network are sized to handle large call volumes.

This type of application is growing everywhere as companies are looking at the bottom-line and this form of technology can provide a significant cost savings over addressing customer's needs with live customer service representatives. A few years ago, if a company was using one of these auto attendant type applications a customer might take the position that this is impersonal and they always wanted a live representative. This same opinion was once voiced when a caller got voice mail instead of a live person. Over the years, voice mail became a standard in business

Building an Audit Trail	2
Collection of External Data	2
Using System Log for Debugging	3
Turning on Debugging	3
Example of Debug Log, script showing that section, & compiler	4

practice where if you don't get the live contact you were trying to reach, you at least want to have the ability to leave a voice mail. Voice mail is now a business function that is accepted by everyone. The same is now true with self-service IVR applications where a user wants the ability to gather and give information 24/7 and has accepted the convenience that this form of solution provides. By the very definition mentioned earlier, your IVR application is working with the data from your various system applications but there are other data needs beyond that.

This month's newsletter covers some additional data collection applications beyond your system application's data.





IBM Solution Connection
Integrates solution information
with IBM eServer, software and
TotalStorage technology



IBM Server Proven
Solutions with experience

Building an Audit Trail

One form of external data collection that most of our clients do beyond working with the live data of the system application is the building of an audit trail of how the system is being used. Some of the data that such a log may contain are:



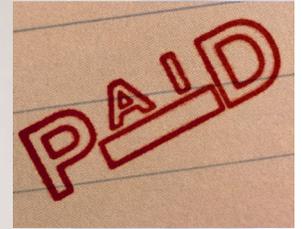
- ⇒ Who is using the system
- ⇒ If your IVR application has different type of users, define the breakout of the caller's types
- ⇒ If the IVR application has different foreign languages supported, the breakout for the language types
- ⇒ When was the call received
- ⇒ What part of the system are they using
- ⇒ Did the caller complete the function they required
- ⇒ What was the length of call in time
- ⇒ Log each transaction that occurs during the call
- ⇒ How many calls is the system taking
- ⇒ Did the user request a call transfer

Next you need to define where you want to store this data. It may be a simple comma delimited file that the customization script saves on the hard drive of the ODTVision VRU or on a LAN drive or it could even be a formatted record in a database file or table.

Although you may not have been thinking of the specifications of your audit trail when you began the design of your IVR application, it is an integral part of your self-service telephony applications. The data it provides will help you modify and improve the IVR applications by allowing you to tailor them to the actual needs of your users and how they are using the system.

Collecting External Data

There are many reasons to collect external data beyond the IVR application. Perhaps your IVR application is being used for payment processing. Now I don't have to tell you that nothing is more important within a firm than the collection of receipts. Even though the IVR application is collecting this data live to the payment processing side of your host system, you may also want to build a parallel data record of that transaction for reconciliation and to serve as a backup if the transaction is ever lost. Please keep in mind that if you do collect such data on the transactions, you must do it under the same security requirements that the actual transaction processing requires. This is paramount as you are required by law to have this data locked down and secure. It can also lead to major liability for your firm if that security on this collected data is ever breached. (Review last month's newsletter on what can happen).



Another reason for the collection of external data may be where you don't have an actual host system application for a feature you want to do through an IVR application. We are currently working with a client where most of the IVR applications are being done by utilizing existing AS400 applications. However, they want to add an application where their own agents using the system can request certain forms and supplies from corporate. With a simple menu driven IVR application they enter their agent number, enter the menu option for supplies and then enter the supply number, and the request is recorded to a comma delimited external file. Once that data

Vision Voice Vantage, Inc. is a certified ISV for IBM. Visit our web site on IBM.com at
<http://www-304.ibm.com/jet09002c/gsdod/solutiondetails.do?solutionId=25001&l>

is collected, it is a simple process to take the records to a fulfillment process and no host applications needed to be designed to accomplish this function.

The caller many have needs that go beyond the application and we need to track those needs. This can be anything from a simple request for a call back to information used to improve the customer service application.

Real Life Story

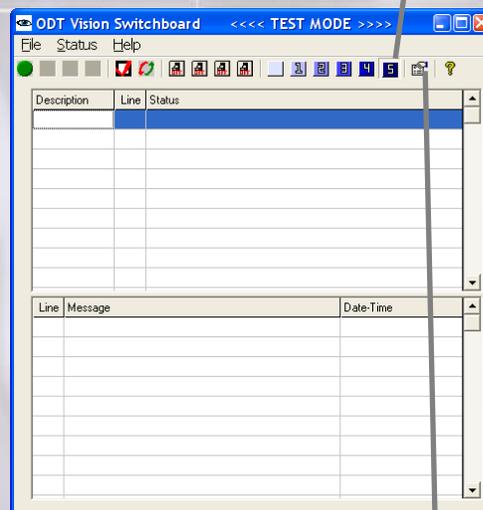
Sometimes you have to use what equipment and phone systems are already in place. We have a client that has an older phone system where the ODTVision VRU will be connected through an analog phone extension. Now in this example, we will do a blind transfer to a hunt group of live customer service representatives. We have no idea which customer service representative will pick up but that individual can see from CallerID which extension of the VRU the call is coming from. What the ODTVision script does is write to the AS400 screen which line they are on (matches a defined extension that shows as the CallerID), what account the current caller was working with, and a 3rd field that defines the reason for the transfer. When the live representative sees the CallerID, they will just click on the icon for which line was transferred and that will automatically populate their current screen with the correct account records including the reason for the call. (i.e. Account # 123456789 is overdue and they want to make arrangements for some form of payment method.)

Using the System Log for Debugging

An enhanced feature of the new ODTVision VRU system is that you can set the debug logging to level 5 and this builds a log entry within the System Log file as each line of the script is executed. This is turned on and used primarily when you are first building and/or testing your customization script or if you are experiencing problems where the IVR application isn't working the way you expect with the customization script.

Turning on level 5 for debugging

First on the Switchboard screen, click level 5 debug level



Now click the line properties icon to bring up the logging screen



PAGE 4

Vision Voice Vantage, Inc.
829 Bethel Road # 213
Columbus, Ohio 43214

Phone: 888-252-2555
Email sales@ODTVision.com



THINKING OUTSIDE THE BOX Get Your Own Demo Today

Contact us to get your own demonstration of the ODTVision Voice Response Unit. This demo application is a simplistic order entry and shipment status system which is running off a Microsoft Access database. The demo is in the test mode and you will be using the "Test Phone" feature of the ODTVISION VRU to simulate a phone call to the data. Manuals and case studies are also available on the web site.

Improving Customer Service Affordability

Get free project analysis regarding your telephony application or submit technical questions at:
TechSupport@ODTVision.com
Or Call: 614-985-3814

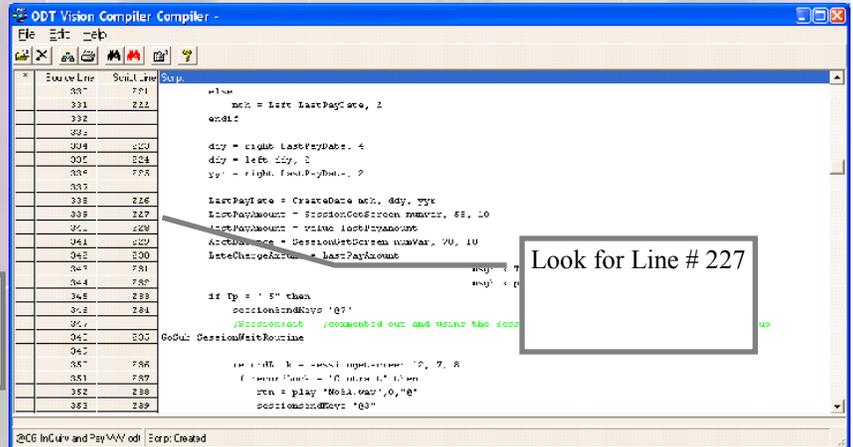
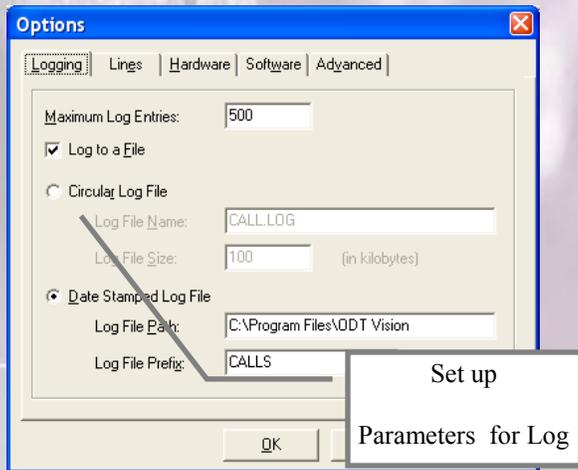
IBM Solution Connection..
Integrates solution information with IBM eServer, software and TotalStorage technology

<http://www-304.ibm.com/jct09002c/gsdod/solutiondetails.do?solutionId=25001&lc=en&stateCd=P&page=1>

http://search400.techtarget.com/productsOfTheYearWinner/0,296407,sid3_gci1157611_tax302575_ayr2005,00.html



Example of Compiler to Get Line Number and Example of Debug Log File Matching Section



Example of Script

LastPayDate = CreateDate mth, ddy, yyr
LastPayAmount = SessionGetScreen numvar, 58, 10
lastPayAmount = value lastPayAmount
AcctBalance = SessionGetScreen numVar, 70, 10
LateChargeAmount = LastPayAmount

12/27/2010 9:49:14 AM Line #1 Script Line #227 - add file to playlist: C:\Program Files\ODT Vision\Voice Files\Roxanne\isdueon.wav
12/27/2010 9:49:14 AM Line #1 Script Line #227 - Playing C:\Program Files\ODT Vision\Voice Files\Roxanne\isdueon.wav
12/27/2010 9:49:16 AM Line #1 Script Line #229 - GoTo line 232
12/27/2010 9:49:16 AM Line #1 Script Line #232 - GoTo line 234
12/27/2010 9:49:16 AM Line #1 Script Line #234 If 0 = 0
12/27/2010 9:49:16 AM Line #1 Script Line #235 - GoTo line 870

Through the use of level 5 for debugging, you can locate any possible errors in the logic of your script. ■