

## Necessity is the Mother of Invention

### Inside this issue:

What Happens When The Sub System Goes Up and Down	2
Elements to reboot the VRU	4
What are your needs Outside Elements of a normal IVR Application	4

This month's newsletter covers what is done when the client's IVR application requires a feature or process that we have not done before or that is not a part of the features and abilities that were built in our solution. This need generally surfaces in the design and implementation phase.

During this period there is a bi-directional communication process going on between us and the client. The client is trying to learn about our solution, how customization is done and how to build and maintain their future IVR applications. Our support personnel are trying to learn about the client's application, the client's environment including systems and data, and what logic and functions are needed by this application to perform all the required tasks.

Here is an example of a requirement in a client's application that was not anticipated. The July newsletter discussed ways to make your script friendlier and simpler for the user to move through their IVR routine. One of the points made in that newsletter was to use the value of the CallerID to identify the user and automatically going directly to that user's record thereby removing the steps where a user would normally identify themselves. Now as outlined in that newsletter, this requires



the company to create and keep a database where all CallerID's are tied to the actual users' records. Once such a database exists, your IVR application can locate that user's record by either supplying the CallerID value to locate the user's record or to search the CallerID database by that CallerID value to locate the user's account number and then using that account variable value to locate the proper record.

As stated, "**Necessity is the Mother of Invention.**" We were in the design phase with a client that planned to use this "CallerID location" method in their IVR application. When we started the discussion of where this CallerID



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



**IBM Server Proven**  
Solutions with experience

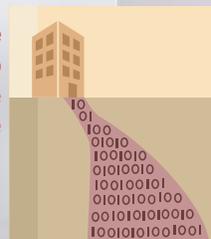
database is currently stored and how would the application connect to.... the client responded, "There is no CallerID database, we thought your system would supply that." Interesting

approach, we hadn't thought of that but no reason it couldn't be done. Within the ODTVision VRU is a container which is actually a Microsoft Access type database. This file is called the



"ODTVision.mdb." Any IVR application that requires ODBC connectivity to other platform database tables normally uses this container. Those linked data tables show up in this container as external Tables files but our customization script can locate records, retrieve information from those located records, write information back into those records or created entirely new record if that is needed. Keep in mind, the IVR application is interacting with the CallerID database:

- ⇒ Locate the record (Yes/No)
- ⇒ Are there multiple accounts using this CallerID, if so, go to the Manual record location section
- ⇒ If Record for CallerID is found, locate proper account & retrieve account number
- ⇒ If No CallerID supplied, go to Manual record location section
- ⇒ If CallerID value has no record assigned to it, locate the record though the manual process and then create a new record in the CallerID database with this assigned account number
- ⇒ If the CallerID value locates a record but the user requires a different account number than the one assigned to it in the CallerID database, then we need multiple accounts assigned to this value for the CallerID. The CallerID database record needs to be updated so that in the future the user will automatically go to the Manual record location section.



This is a great method to simplify the application for the caller and make it possible to always locate the correct record. You just build within the application the ability to locate the proper records by account number and perhaps a cross-reference second account identifying variable such as zip code if no CallerID record exist. If there are multiple accounts assigned to this CallerID, that needs to be addressed. If the user's CallerID value does not exist in the CallerID Database, your application needs to have the logic to build such a record. If no CallerID exist, that needs to be handled as well. It is recommended that the IVR application repeats back to the caller the value of the account once located but if there are multiple accounts tied to this CallerID the user will still need to manually input the record. In addition, the logic needs to update the database if it is now discovered that the same CallerID value has multiple records.

Back to the client that had no CallerID database. They will have users that will be calling in monthly to hear the status of their account and once informed of amount due, make a payment. The nature of their users dictated that we needed a very simple IVR logic as their users are not very sophisticated. The application dictated the ODTVision's customization script to build and maintain a internal rather than external database. This database is the same as if it had existed on another platform but is being built, maintained and even updated in the ODTVision unit by user calls.

## What Happens When the Sub-System Goes Up and Down?

Many of our client's IVR application are on an IBM midrange host where we are doing



PAGE 3

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 ODT VISION 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](mailto:TechSupport@ODTVision.com)  
Or Call: 888-252-2555



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



<http://www-304.ibm.com/partnerworld/gsd/solutiondetails.do?&solution=25001&lc=en>  
<http://search400.techtarget.com/review/Ohio-Data-Transfers-DTT1000-Voice-Response-Unit-60>

<http://www.ibmssystemsmag.com/ibmi/productnews/>

simple HLAPI screen scraping to an existing display screen application. The connection to the actual data is through this legacy green screen application through the HLAPI API interface through IBM Client Access. There is an IBM Client Access display session for each line of the system. After the initial connection of a new phone call, the script will always check to see if the System Available is high (signals that we have a connection to the subsystem) and most likely will drop through this section as the unit and software does have active communication with the IBM host. If the subsystem is down, the caller would hear that voice file that the host system is not available at the moment and to call back later.

Let's say the subsystem is taken down weekly for routine system maintenance. Since the IBM Client Access is a true IP connection, the break in connection will not come back unless we re-issue the display session. "Necessity is the Mother of Invention" and we have a problem to resolve. Either a computer operator can manually restart the client access sessions or we can automate this process.

### Here is how we do that

While the subsystem is down, the IVR application doesn't care, the unit continues to answer the calls so the users know they have called the proper phone number but at that point they would hear the voice file, "the host system is down for maintenance."

### Elements to reboot the VRU



Since this is a weekly maintenance routine where the host subsystem goes down automatically, the VRU need to bring down the IVR application and then provide for the system to come back on line automatically. This is very easy to do.



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  
IN CUSTOMER SERVICE

## 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](mailto:TechSupport@ODTVision.com)  
Or Call: 888-252-2555



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



**IBM Server Proven**  
Solutions with experience



<http://www.ibmssystemsmag.com/ibmi/productnews/productreviews/Product-Review--DTT1000-from-Ohio-Data-Transfer/>

First configure the lines of the ODTVision to start automatically in the setup configuration screen under the line

Second, edit a build-in a wait statement to allow the following to occur:

- ⇒ Wait of the Windows OS to load
- ⇒ Wait for IBM Client Access Sessions to start (they must be configured to auto log on the 1st level of IBM security)
- ⇒ Wait for Dialogic cards microcode to load



ured to auto start), and the ODTVision lines are loaded again, the next caller to each line will be signed back onto the host and you have successfully restarted they IVR application.

## What are your needs – even if they appear to be outside the elements of a normal IVR application?

From the scheduler feature of windows, scheduler, create a Windows Event to reboot the PC based on a day and time element where you know the sub-system will be back up. This element should match the day and time where the sub-system is available again.

Also add to the windows startup group IBM Client Access Sessions for each display session (make sure these displays are configured to automatically go to the sign-on display) and the ODTVision switchboard program (make sure each line is set auto-start).

Now that the Client Access session is restarted, the dialogic microcode is loaded (it should be config-

This month's newsletter covered just two situations where the needs of the client's IVR application seemed to be outside the capabilities of our software. In most cases that appear this way, with a little thought there is likely a workaround that allows us to address your application requirements. Please contact us with your thoughts if you believe you have some requirements that might seem .."Outside the box." ■