



Creating Custom Applications for TeleVantage 3.5

With TeleVantage 3.5, an open, standards-based soft switch and suite of telephony applications, businesses can extend TeleVantage's capabilities to implement a complete communication platform by integrating such technologies as speech recognition, fax, interactive voice recognition (IVR), and desktop screen pops. In many cases, no programming customization is necessary to integrate your IVR or desktop screen-pop applications with TeleVantage.

The TeleVantage SDK enables you to integrate your telephony application with the phone system, sharing hardware resources and incorporating your business data into the TeleVantage Client desktop GUI. The TeleVantage SDK complies with industry-standard APIs including Microsoft COM and TAPI, allowing you to use all popular application development tools and integrate with your current tools and databases.

This document offers an overview of the TeleVantage 3.5 interfaces that allow for such customization capabilities.

Phone systems customizations come in many different types and can satisfy a variety of business needs.

- Typical interactive voice response (IVR) applications provide access to information via voice prompts and DTMF tones. Callers today have completed banking transactions, obtained prescription refills, and accomplished other tasks using key information exchanged via telephone systems.
- Screen-pop applications cause a desktop application to appear in response to an incoming phone call.
- Outbound dialing applications can help automate outbound calling campaigns.
- Other telephony technology applications include speech recognition, faxing, and text-to-speech software.

Topics covered here include:

- I. In-band signaling applications:** in-band signaling helps integrate existing applications with TeleVantage without modifying the original code.
- II. IVR Plug-in API:** IVR plug-ins allow hardware resource sharing and data exchange between telephony applications and TeleVantage.
- III. Device Status API:** the device status API provides users with a way to see real-time status information about trunks and stations.
- IV. Telephony Application Program Interface (TAPI):** TeleVantage provides a standard TAPI interface for desktop and call control applications through the TAPI Service Provider.
- V. More Information:** A list of additional Artisoft resources for details on how to customize TeleVantage.

In-band signaling applications

TeleVantage users can communicate with the TeleVantage server by entering digits or flash hook signals on the telephone keypad. These telephone commands can perform many common functions such as transferring and conferencing. A telephony application can communicate with TeleVantage in the same way, receiving calls as a station device (just like a telephone handset), processing the calls, and

transferring the call back into TeleVantage via the same flash hook commands. This kind of interaction between TeleVantage and another applications is called “in-band signaling.”

In-band signaling applications work with TeleVantage and do not require any code changes to, so in-band signaling is a good way to integrate an existing or off-the-shelf application with a TeleVantage system. In-band signaling applications can exist on a separate server, or as a running application within the same computer as TeleVantage.

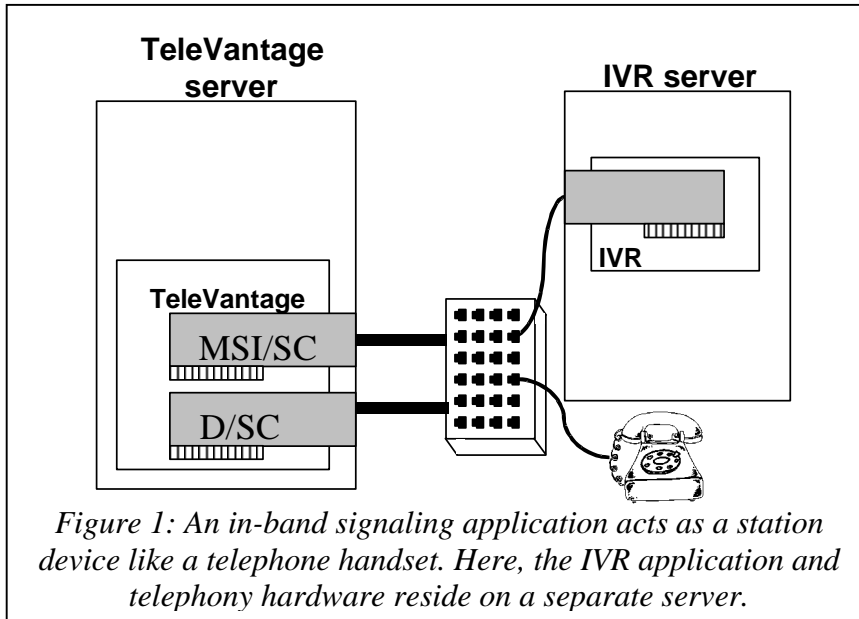
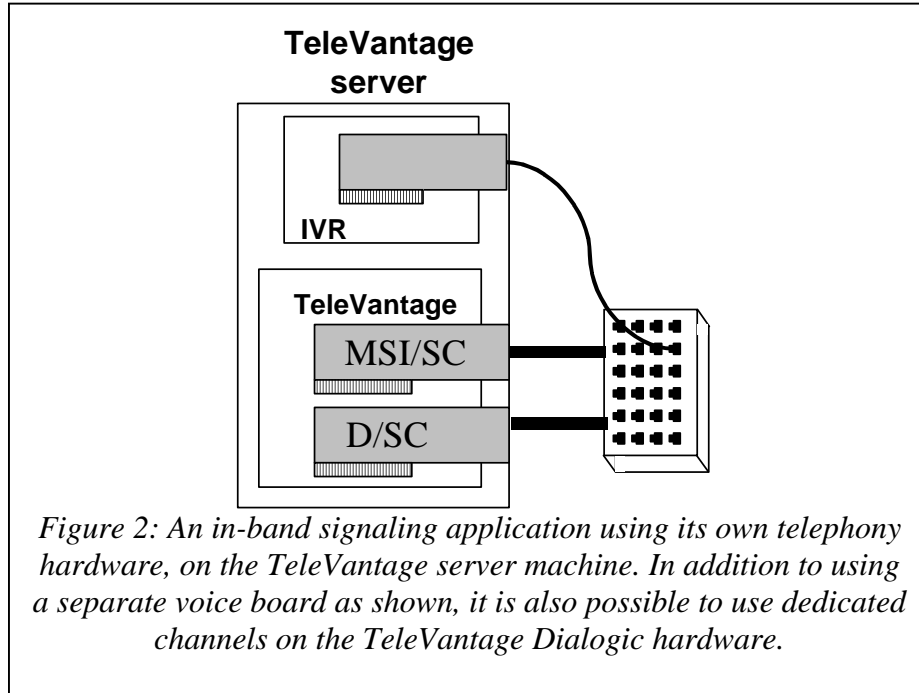


Figure 1, above, shows a separate server configuration; this configuration is preferable if the IVR application is particularly hardware-intensive, or if the telephony hardware (or any other system component) is not compatible with the Dialogic hardware TeleVantage uses. It is also possible to install and run your in-band signaling application on the same server as TeleVantage. Careful load testing should be performed to ensure that the IVR application does not affect TeleVantage performance (or vice versa).

In-band signaling applications on the TeleVantage server PC

Figure 2, on the next page, shows an in-band signaling application on the TeleVantage server machine. Although the PC and other hardware shared between the TeleVantage Server process and the IVR application, the telephony hardware resources (such as one or more channels on a Dialogic board) must be dedicated to the IVR application.

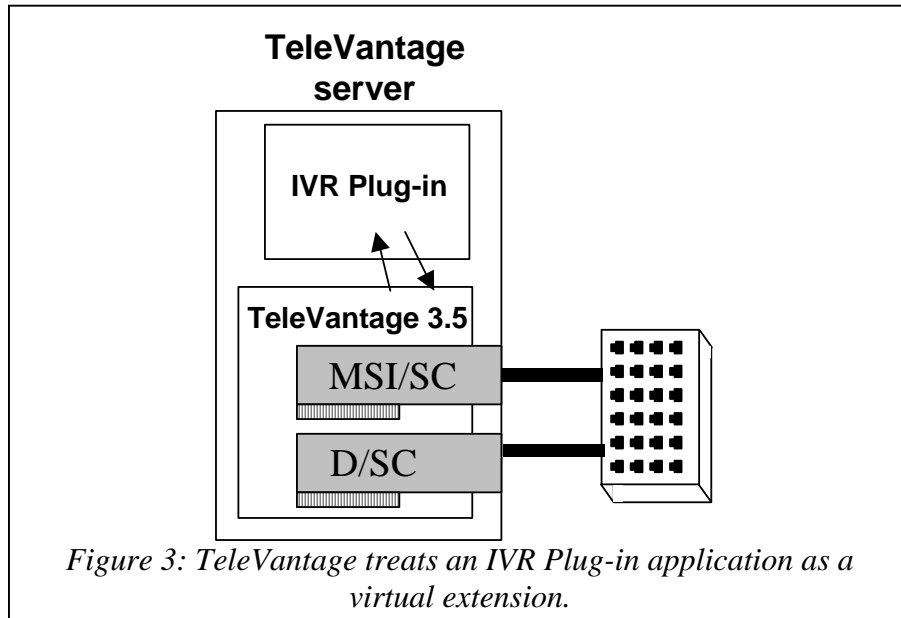


If Dialogic hardware is installed in the TeleVantage server machine, TeleVantage will automatically take control of all resources on each board. Special configuration settings must be used to isolate these resources for use by the IVR application. Although the IVR application executable may reside on the same server PC as the TeleVantage server process, in-band signaling applications are unable to share data with TeleVantage. Therefore, any interesting information about the call — including the caller's identification — cannot be passed between TeleVantage and the IVR.

In-band signaling application:	IVR Plug-in application:
<ul style="list-style-type: none"> • Additional telephony hardware (including voice and station resources) required 	<ul style="list-style-type: none"> • No additional telephony hardware needed.
<ul style="list-style-type: none"> • IVR application requires few, if any programming changes 	<ul style="list-style-type: none"> • Programming changes specific to TeleVantage required in IVR application.
<ul style="list-style-type: none"> • Data cannot be exchanged between IVR and TeleVantage. 	<ul style="list-style-type: none"> • Data can be passed between IVR and TeleVantage.

The IVR Plug-in API

TeleVantage 3.5 has an application programming interface, or API, which allows users to create IVR applications and integrate them with TeleVantage. When a custom application is written to the IVR plug-in API, they act as “virtual extensions” to TeleVantage. An IVR plug-in is a special type of extension on the TeleVantage Server that runs an IVR application object written to the IVR plug-in API when it is called.



The IVR plug-in API consists of several COM objects. An IVR plug-in application is written using any Microsoft COM-compliant development language such as Visual Basic or Visual C++ and compiled as an ActiveX executable. This executable must be registered in the TeleVantage Administrator and receives an extension; when a call is directed to the application (by an auto attendant, transfer from another user, DID, or any other means), TeleVantage creates the application object and passes it the Dialogic voice resource associated with the call.

IVR Plug-in Event	Description
CallOffering	Triggered when a call is offered to the IVR plug-in.
CallTerminated	Indicates that TeleVantage should terminate the IVR plug-in (generally, when the caller has hung up).
CallPlaced	Invoked when an IVR plug-in is receiving control of an outbound call placed by another plug-in (using the PlaceCall method).

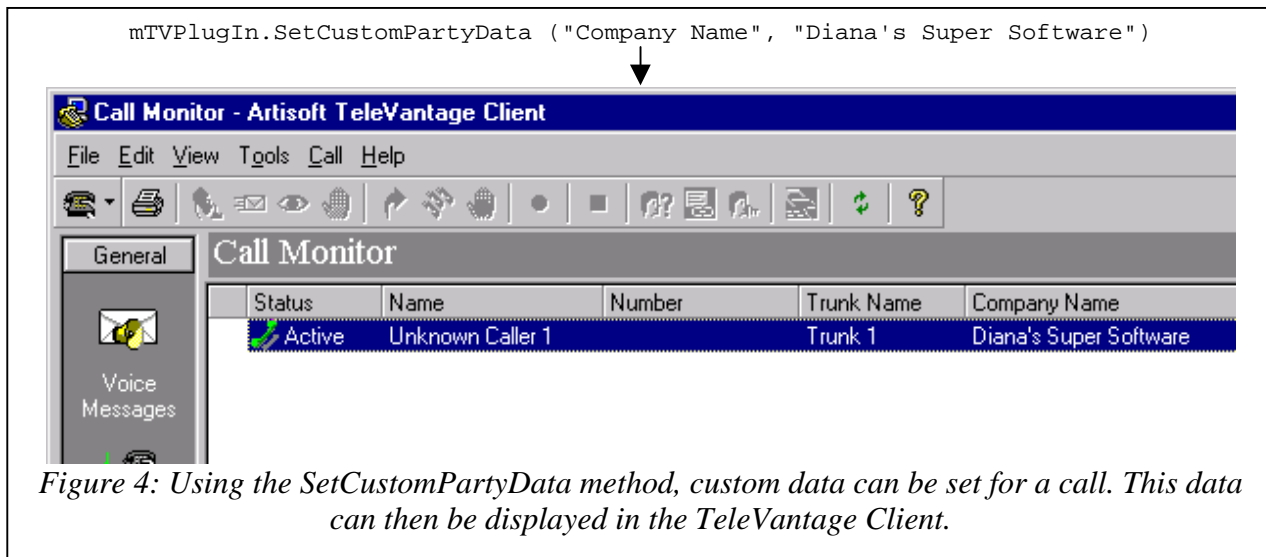
Once the IVR plug-in enabled application (or “IVR plug-in”) has received a call, the code can access the methods and properties of the IVRPlugIn2 object to interface with the TeleVantage Server. The IVRPlugIn2 object has methods for retrieving and setting data about the call (including CallerID, DID, and the trunk and station devices in use), administering the IVR plug-in (such as SetStatus, which allows you to write information to the TeleVantage Device Monitor), and completing the call and passing control back to TeleVantage to transfer or hang up (CallDone).

A few of the methods available in the IVRPlugIn2 object are shown in the table below.

IVR Plug-in Method	Description
CallDone	Returns control of the call to TeleVantage. Takes parameters to indicate how the call should be handled (transferred to an extension, hung up, etc.)
Get/SetPartyData	Used to retrieve and set data about the call (CallerID name and number, DID, device used).
Get/SetCustomPartyData	Used to retrieve and set custom business data about the call in name-value pairs (see next section for more detail).
GetXmitTimeslot	Provides access to the SCBus timeslot in use for the call (can be used to connect call to a fax device, speech recognition resource, etc.).
PlaceCall	Places an outbound call. Takes a parameter to indicate how the call should be handled once placed (for example, answered by a station user or another IVR plug-in).
SetStatus	Sends a string to be displayed in the TeleVantage Device Monitor, to monitor the activity of an IVR plug-in.

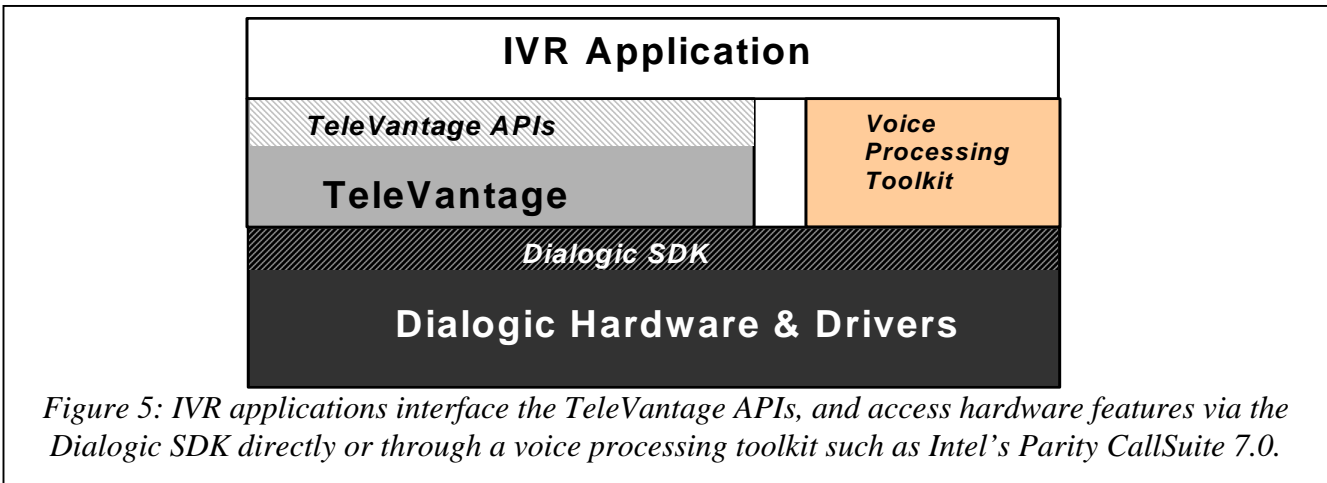
Associating data with a call

The IVRPlugIn2 object has methods to retrieve and set information about a call such as CallerID. It also has methods to associate other data — application-specific data, stored as name-value pairs — with a call. This is a good place to add data entered by the user (for example, the customer ID) or data retrieved from a customer database (e.g., caller credit status). This data can then be used by other IVR plug-ins or TAPI applications (see “Using TAPI with TeleVantage”). It may also be viewed in the TeleVantage Client and saved in the call log with the call.



Voice processing in an IVR Plug-in

The TeleVantage IVR plug-in interface passes control of a call to a custom application, and it can also access and set data about a call such as CallerID and DID information. For many applications, no additional software is required. However, if you are developing a custom IVR application that must perform interactive voice processing tasks, such as playing voice prompts and collecting DTMF tones, the application will need to access the Dialogic board resources to perform these functions. To perform any of those functions, one may write directly to the Dialogic SDK, but the API is intricate and the writer should be conversant in C or C++ is required. It is often preferable to use additional software – a voice processing toolkit — situated between the written application and the Dialogic drivers. One such toolkit is Intel’s Parity CallSuite 7.0.



An IVR plug-in receives a call by triggering the CallOffering event. One of the arguments to this event is a reference to the Dialogic voice resource in use for the call; this reference is used to let the voice processing software know what voice resource and channel to use. For Parity CallSuite, this is done using the VoiceBocx.MediaResource property.

Once the VoiceBocx.MediaResource property is set, other methods in the VoiceBocx control can be used to play and record voice prompts, retrieve DTMF digits, and other voice processing functions. The following Visual Basic code sample shows a full CallOffering event handler including the playing of a voice prompt by Parity CallSuite.

The IVR Plug-in API

```
Private Sub IVRPlugInNotify2_CallOffering(ByVal hParty As Integer, ByVal Line As Integer)

    ' This event is triggered when a call is offered by the telephony server
    ' It plays a greeting prompt, then transfers the call to the Operator.
    '
    ' Inputs:
    '     hParty           - a unique party id that is used by the application
    '                       to identify itself when calling back to TeleVantage
    '     Line            - Line number of the Dialogic resource associated
    '                       with the call

    On Error GoTo VoiceError:

    'This is the voice resource that the telephony server will use for the call.
    mVoiceBox.MediaChannel = Line - 1

    'Clearing the digit buffer. This is required before audio can be heard.
    mVoiceBox.ClearDigits

    ' The SetStatus method changes the call status the TeleVantage Device Monitor.
    mTVPlugIn.SetStatus hParty, "Call detected: " & Line

    ' Use the VoiceBocx control to play a file
    mVoiceBox.PlayFile "Hello.vox", 3

    ' Tell TV to transfer the call to the Operator
    mTVPlugIn.CallDone hParty, naTransferToExt, "0"

Exit Sub

VoiceError:

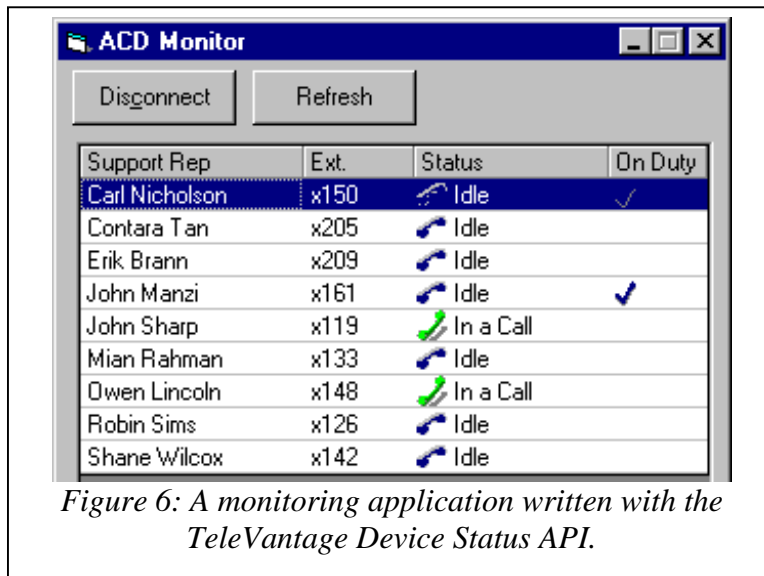
    On Error Resume Next
    ' [error handling code here]

    ' Tell TV to transfer the call to the Operator
    mTVPlugIn.CallDone hParty, naTransferToExt, "0"

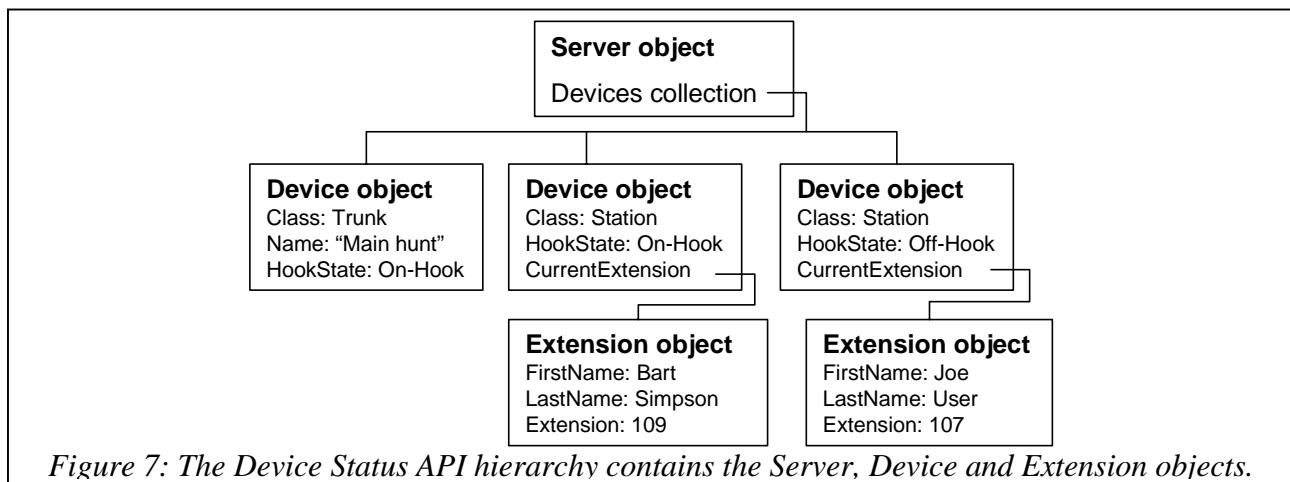
End Sub
```

The Device Status API

The Device Status API is a TeleVantage programming interface that can be incorporated into applications that require updated information about trunk and station devices on the TeleVantage server. This information can be used to write visual monitoring applications or utilities reporting the activity for trunks, stations, or users. This information is also important when writing outbound dialing applications; with the Device Status API, it is possible to know which agents are available to take a successfully placed outbound call.

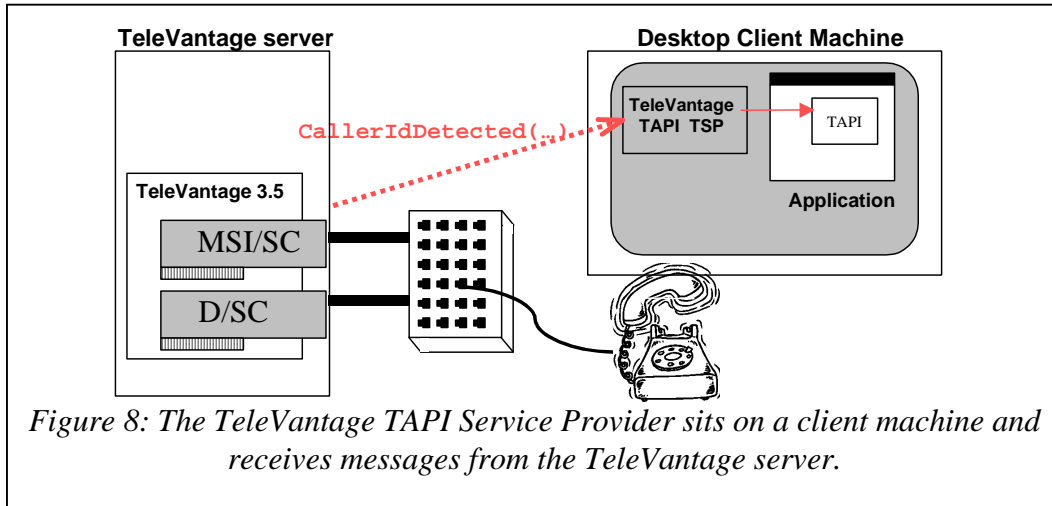


Device Status API consists of a set of hierarchically arranged objects representing the TeleVantage server and all of the devices on it. The server object has a Devices property that is a collection of Device objects. Each device object has a Class property (either Trunk or Station); another property represents the hook status of the device. All device objects of class station have properties representing the extensions associated with that station (DefaultExtension, CurrentExtension, and AssignedExtensions collection). Extension objects have properties containing the name and Do Not Disturb status of the extension. The Server object receives a DeviceStateChanged event every time the state of any device changes, making it possible to create an application that shows the real-time hook state of devices.



Using TAPI with TeleVantage

TeleVantage 3.5 also provides a TAPI 2.1-compliant Service Provider (TSP), which allows the creation of single-line call control applications on a remote machine (such as a client desktop machine). It is possible to use the TeleVantage TAPI Service Provider to interface to TAPI-enabled desktop applications (such as CRM applications); the TSP provides functions to place outbound calls and to receive screen pops from the applications.



TeleVantage 3.5 includes software to enable the integration with some popular TAPI-enabled customer relationship management (CRM) applications (including FrontRange GoldMine, Microsoft Outlook, and ACT!). Other TAPI-enabled off-the-shelf applications may require custom intermediary software; it is also possible to incorporate TAPI into a fully customized application to enable screen pops, outbound dialing, and other telephony functions into your application.

TeleVantage 3.5 TSP supported functions:
MakeCall
Dial
Drop
Hold
Unhold
Park
Supervised Transfer
Blind Transfer

It is also possible to access TeleVantage custom call data (as set by an IVR plug-in) in a TAPI application by using the TeleVantage TAPI Type library.

For more information

Administering TeleVantage, Chapter 13: Extending TeleVantage (*offers an overview of the TeleVantage 3.5 SDK*)

TeleVantage SDK Developer's Guide: available as an Adobe Acrobat file on the TeleVantage CD (included when the SDK is installed). The developer's guide is a full programmer's reference to the TeleVantage 3.5 SDK, and also contains information on installing, configuring, and debugging custom applications

Administering TeleVantage, Appendix A: TeleVantage Configuration Settings (includes information helpful to in-band signaling application installation)

Artisoft, Inc. provides educational training and support services for creating custom applications for TeleVantage and a Professional Services team to assist in your custom programming requirements. For more information, visit www.artisoft.com.

About Artisoft

Artisoft is a global leader in open, standards-based telephone switching (PBX) systems that bring together voice and data for more powerful and productive communications. Artisoft's software switches deliver a superior feature set, greater flexibility and lower cost of ownership through its TeleVantage brand of soft PBXs, and its Professional Services, which work with growing businesses to design software solutions to fit their unique needs. Artisoft's products and services are delivered worldwide through a dedicated and growing channel of authorized resellers.

About TeleVantage

TeleVantage® combines the advantages of your two most essential business tools, your personal computer and phone, into one powerful and productive communications system. A true software-PBX, TeleVantage is based on open industry standards. It runs on a standard PC, uses industry-standard line cards from Dialogic®, an Intel Company, runs on the Microsoft Windows operating systems and works with virtually any standard analog or CLASS feature telephone. One of the key advantages of the TeleVantage soft switch PBX is that it provides an open platform that can interface with other best-of-breed technologies from customer relationship management to e-mail.

About the Open Communications Alliance (OCA)

The trend toward open, standards-based phone systems has opened up a range of choices, allowing today's businesses to pick and choose the hardware components and applications that best suit their budgets and requirements. Artisoft's Open Communications Alliance is made up of vendors that use the TeleVantage software development kit (SDK) and industry standards to develop products and communications applications and technologies with the TeleVantage open, software-based phone system as the focal point.