



ShoreTel Developer Network Interface Catalog

**ShoreTel, Inc.
960 Stewart Drive
Sunnyvale, CA 94085
July, 2009**

www.shoretel.com

1.	Introduction.....	1
1.1.	TAPI SDK.....	1
1.2.	COM SDK	1
1.3.	Web Service SDK.....	2
1.4.	Phone API (PAPI) SDK.....	2
1.5.	ScriptHlp SDK.....	2
2.	Telephony Application Programming Interface (TAPI).....	2
2.1.	TAPI Programming Options	3
2.1.1.	TAPI for Managing a Single Extension.....	3
2.1.2.	TAPI for Managing Multiple Extensions or Media Interactions	3
2.1.3.	TAPI Development Tools.....	4
2.1.4.	ShoreTel TAPI Extensions	4
3.	ShoreTel Multi-Line Control (COM OBJECT).....	5
4.	ShoreTel Web Service SDK	6
5.	Phone API (PAPI) SDK.....	9
6.	ScriptHlp SDK.....	10
7.	Developer Network Packages	10
7.1.	TAPI Software Developers Kit (SDK)	10
7.2.	COM Object and Desktop Samples	10
7.3.	COM Object Server Samples.....	12
7.4.	ShoreTel Web Service Software Development Kit (SDK).....	14
7.5.	ShoreTel Phone API Software Development Kit (SDK).....	14
7.6.	ScriptHlp SDK.....	15

1. Introduction

The ShoreTel system can be extended to provide greater capabilities for system administrators and end users. The programmatic interfaces available in the Developer's Network allow the creation of rich new applications and integrate ShoreTel's telephony features into businesses' central, back-office systems.

The ShoreTel Developer Network provides following five Software Development Kits (SDKs) for use by call control and monitoring applications to integrate with the ShoreTel phone system.

1.1. TAPI SDK

Using the Telephony Application Programming Interface (TAPI), advanced application programmers can develop 1st and 3rd party call control applications that work with the ShoreTel system and enable new capabilities and business integration. Applications developed on top of the TAPI interface can monitor and/or control phones, lines, trunks and calls. In addition, applications can interact with callers using the API's built in media support.

TAPI SDK provides base information and several samples for developers who wish to program directly to the ShoreTel TAPI 2.x interface. TAPI is Microsoft's Telephony Application Programming Interface and is a core computer telephony protocol supported by ShoreTel on both client and server PCs. The TAPI SDK also provides include files needed to build the COM object in the COM SDK. In addition, it provides documentation about the core TAPI features available in the ShoreTel TAPI interface. This set of documents and samples will be helpful even for those developers who principally use the COM Object for development.

1.2. COM SDK

Developers using a COM aware language can also elect to do development using the interfaces provided by ShoreTel's multi-line control com object. This enables applications developed in languages such as Visual Basic or C# .NET to interact with and control the telephony capabilities of the ShoreTel system. The multi-line control simplifies development while providing a broad set of telephony capabilities.

COM SDK is aimed at both client- and server-type call control and monitoring applications. Client applications are typically single line non-wave applications such as screen pop, integration with CRM applications, click to call features, call loggers, and time accounting systems. Server applications include multi-line and/or wave play and record applications such as interactive voice response, automatic call distribution, and voice mail.

1.3. Web Service SDK

The purpose of the ShoreTel Web Service (STWS) SDK is to enable ShoreTel partners and customers to rapidly build ShoreTel integrated, advanced software applications *using programming languages of choice on platforms of choice*. STWS SDK is intended to extend the reach of ShoreTel's current telephony SDK to non-Windows and non-Microsoft COM based applications. STWS SDK has been designed to grow with ShoreTel.

1.4. Phone API (PAPI) SDK

The ShoreTel PAPI SDK enables application developers to deliver new applications to the ShoreTel phone family. PAPI allows applications to control specific user interface elements (display, audio, alerts, keys) on the phone to deliver applications to phone users. Specific capabilities differ by model of phone. Following are a few examples of applications that may be developed using PAPI:

- School attendance
- Time Cards
- Emergency Alerts
- ACD Statistics
- Account code entry on incoming call
- Hospitality Services (wake up call, etc)

1.5. ScriptHlp SDK

The ScriptHlp SDK enables server scripts and applications to assign a ShoreTel user to his/her home phone, change a user's Call Handling Mode, or change the group to which a user is assigned.

2. Telephony Application Programming Interface (TAPI)

TAPI provides advanced programming flexibility for applications to monitor and/or manage lines/extensions, phones, and calls in the ShoreTel system. The ShoreTel system supports two (2) options for using the TAPI interface.

A single line interface is installed on every ShoreTel user's desktop via the Call Manager application. With the single line interface, applications can monitor and control the individual users' telephone in concert with other desktop applications or activities.

A second, multi-line interface is available through the installation of the Distributed Voice Services software. The Voice Services software provides an interface that allows applications to monitor and/or control all the lines/extensions, calls, phones, and trunks in the system as well as interact with callers through the associated media interface.



Access to the TAPI interface documentation, samples, and developer tools is provided through membership in the ShoreTel Developer Network. For information on joining the network, refer to www.shoretel.com or send an email to devnet@shoretel.com.

2.1. TAPI Programming Options

2.1.1. TAPI for Managing a Single Extension

The ShoreTel system provides a Microsoft Windows TAPI interface at every user's desktop computer. This provides an interface for applications to monitor and manage the user's telephony activities. When new calls arrive, TAPI provides notifications that applications can use to automate user or agent activities such as automatically opening matching customer records or recording the call in a log. Through this interface, applications can also implement call management actions such as launching a call from an external application, or invoking call monitoring or recording.

The TAPI interface provided on the end-user's desktop is Microsoft Windows TAPI 2.1 compliant and supports both line and phone devices as well as support for ShoreTel extensions such as call monitoring, recording, and barge.

Note: The end user must purchase and install the ShoreTel Call Manager to enable the TAPI interface for advanced applications on their desktop computer.

2.1.2. TAPI for Managing Multiple Extensions or Media Interactions

The ShoreTel Distributed Voice Services Server software provides an application hosting platform for advanced, call processing applications. This software provides a TAPI interface that enables monitoring and management of not just a single line or extension, but of multiple lines in the system. Additionally, trunk monitoring operations are supported.

The TAPI interface provided on the Distributed Voice Services Server supports the TAPI Wave interface allowing applications to interact with callers. Route Points are configured to provide applications an interface to make and answer calls as well as interact with the caller through the Route Point's TAPI Wave interface. Route Ports are dial-able numbers and therefore enable the application to have an extension in the system. Applications interfacing through the Route Point can both play files for the caller to hear as well as listen for the callers DTMF or audio responses¹.

¹ The ShoreTel system supports a maximum of 255 media streams per server.



These TAPI interfaces installed on the Distributed Voice Services Software enables highly advanced applications to be created for the ShoreTel platform that implement both 1st and 3rd party call control functions such as Interactive Voice Response (IVR) Systems, Call Recording, or Do Not Call Restrictions.

Note: The end user must purchase and install the ShoreTel Distributed Voice Services Server Software to access the server side TAPI interface. Additionally, when used for hosting advanced applications, the system must be dedicated to the 3rd party application and can not be used to host mailboxes or menus.

2.1.3. TAPI Development Tools

The TAPI SDK includes additional tools designed to assist developers in creating new applications and integrating with other business applications. Tools are provided to view the status of the TAPI interface, play wave files through the TAPI Wave Interface, and also enable/disable debug options. The SDK includes source code for all of the test tools to exemplify the use of the interfaces.

2.1.4. ShoreTel TAPI Extensions

ShoreTel TAPI supports numerous extensions to the standard Microsoft TAPI 2.1 interface to enable richer applications and more complete end user solutions.

Far End Answer Call State	Provides an indication to that the external call (trunk call) has been answered at the far end of the call. (Only supported with the call is placed through a PRI or SIP Trunk).
Globally Unique Identifiers	Provides an identifier that is unique throughout a distributed ShoreTel system and remains unique in high call volume systems.
Get DNIS Information	Provides the application both the DNIS information from the phone network as well as the “friendly DNIS” or the textual description in ShoreWare Director.
Call Recording	Redirects a copy of the media to a specific route point that is – typically – terminated by an application designed to record the media ² .

² When an application invokes call recording, the recording state will be shown to the end user who’s call is being recorded in the user’s call manager application.

Silent Monitor/Barge	Initiates a monitoring session or barges into a call on the indicated extension.
Set /Get Call Properties	Provides a way for an application to set and access call properties through TAPI. Via these mechanisms, applications can pass application specific information to other TAPI applications inside the system.
Set Call Coverage	Provides a mechanism to transfer calls to destinations while at the same time specifying the desired call coverage. For example, transferring a call to another destination but instructing the system to ignore the user's current call forward always configuration.
Disable Music On Hold	Provide a mechanism to place a call on hold without having the caller hear music on hold.
Send Call to Mailbox	Send the call to a specific system mailbox.
Get Server Time	Returns the current time and time zone information from ShoreTel server.
Set ANI Name and Number	Provides a mechanism for applications to set not just the ANI number for a call but, to also specify the name.

3. ShoreTel Multi-Line Control (COM OBJECT)

ShoreTel provides desktop application developers an enhanced interface that simplifies integrating telephony into back-office systems or creating custom capabilities. The object provides an easy-to-use, object based programming interface that can be embedded in other applications simplifying the job of application development.

The multi-line control uses the services of TAPI. The object can be installed and used on either the client TAPI interface to provide single line, 1st party control or on the server TAPI interface to provide multi-line, 1st and 3rd party monitoring and control. Additionally, a media interface is provided to support caller interactions.



The multi-line control can be used directly by COM capable languages including Visual Basic, scripts such as VBScript and Jscript as well as C# for .NET programming. The object features:

- A COM interface enabling simplified application development
- Enumerations of all TAPI and ShoreTel values for the application
- Caching all dynamic call property data to simplify TAPI usage
- Media interface and digit interaction support (DTMF interactions)
- Methods for setting and retrieving line and call events and settings
- A COM interface exposing Phone Events for detecting digit presses either before a call is made or during an active call

The multi-line control interface, like the TAPI interface, enables application developers to create desktop-based end-user applications that automate the activities of the user by dialing the phone for them, automatically updating other applications when calls arrive, logging telephony events, or automatically recording specific calls.

The interface also allows application developers to create advanced, multi-line applications that log events from multiple devices, provide 3rd party control of remote extensions, or automatically respond to telephony activities throughout the system.

The control is provided both as a binary that can be leveraged by the developer inside a custom program and as C++ source code that can be modified for the developers specific needs in conjunction with the TAPI toolkit.

4. ShoreTel Web Service SDK

The purpose of the ShoreTel Web Service (STWS) SDK is to enable ShoreTel partners and customers to rapidly build ShoreTel integrated, advanced software applications *using programming languages of choice on platforms of choice*. STWS SDK is currently a wrapper around existing ShoreTel COM control and is intended to extend the reach of ShoreTel's current telephony SDK to non-Windows and non-Microsoft COM based applications. The ShoreTel Web Service (STWS) SDK provides following benefits:

- Platform independent consumption

The STWS SDK WSDL provides a platform independent software interface that can be easily used from within almost any software application built using any programming language on any system platform. Consumption of STWS SDK WSDL has been successfully validated using following programming languages on following OS platforms:

- C# and Visual Basic.NET on Microsoft Windows
- Java on Microsoft Windows, Linux and MAC



- Perl on Microsoft Windows
- Ruby on Microsoft Windows

- Compliance with Industry Standards

The web service interface is conformant with **WS-I basic profile 1.1** (<http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html>) for maximum interoperability. It can be configured to use standard HTTPS/SSL for transport level security or message level security based on web service standards like WS-Security, WS-Trust and others.

- Tool support to generate client side proxies

Most popular programming languages on multiple platforms provide tools to generate client side proxies from web service WSDLs. The generated client side proxies obviate the need for application developers to build messages manually and to directly manage transport layer functions necessary to exchange the messages over the wire. This promotes ease of use of STWS SDK.

- Easy learning curve for ShoreTel COM SDK users

The interface exposed by the STWS SDK – method names, constants, error codes - directly map to those provided by the ShoreTel COM control. As a result, existing ShoreTel COM SDK developers should be able to easily master the STWS SDK.

- State Management

The STWS SDK manages “state” (such as lines opened, clients sharing a line, active call(s) on lines and call attributes) on behalf of software clients. This allows a client application to be stateless and eases the programming effort for such stateless clients.

- Flexible telephony event delivery to clients

The STWS SDK supports both “pull” and “push” models for telephony event delivery to clients. Client applications can choose to “pull” telephony events or choose to get them “pushed” in an out of band mode on a TCP connection. The reason for custom push model support is that web services today inherently do not support the asynchronous event delivery.

- Support for bandwidth sensitive clients

A client could choose the amount of information it wants to get and process. A large response payload of a web service method is often an issue for bandwidth-sensitive clients like mobile phones running on UMTS/GPRS networks where, not only the bandwidth, but also the price of the data exchanged might be necessary to constrain. Typically applications running on mobile phones are optimized so as to rely upon only minimal information to process.



- Simple click-to-call support

A common need for web based clients is to provide a way for an application user to dial a number by clicking a hyper link or button. The STWS SDK facilitates this by providing facility to make a one shot call.

Below is a matrix that compares the existing ShoreTel COM SDK and the new Web Service SDK along several different comparative aspects to give developers a sense of when it might be most advantageous to choose one over the other.

	ShoreTel COM SDK	ShoreTel Web Service SDK	Notes
<i>Client Platform</i>	Windows	Platform independent	
<i>Client Usage model</i>	In Process of Client Application	Cross Process and Cross machine	
<i>Development environment</i>	Visual Studio or any COM supporting development tool on Windows	Platform independent	
<i>Message Format and Binding</i>	Binary using standard COM ORPC(Object RPC)	SOAP/XML over HTTP(S) binding	
<i>Message Security</i>	Inherent	Optional using HTTPS and WS-Security	HTTPS binding could be used for transport level security. Alternatively, WS Security could be used for message level security.
<i>Support of client behind NAT and Firewalls</i>	N/A	Yes	Web Service access ports for requests and events need to be opened in the firewall.
<i>Telephony Events delivery</i>	In Process using standard COM event sinks	Pull model over HTTP(s) and/or push model over TCP connection	
<i>Support for all types of ShoreTel lines</i>	Yes	Only station type of ShoreTel lines are currently supported	
<i>Wave support</i>	Yes	No	
<i>Support for</i>	Yes	No	

<i>phone device events and methods</i>			
<i>Support for DTMF digit generation</i>	Yes	No	

5. Phone API (PAPI) SDK

The ShoreTel PAPI SDK enables application developers to deliver new applications to the ShoreTel phone family. PAPI allows applications to control specific user interface elements (display, audio, alerts, keys) on the phone to deliver applications to phone users. Specific capabilities differ by model of phone. Following are a few examples of applications that may be developed using PAPI:

- School attendance
- Time Cards
- Emergency Alerts
- ACD Statistics
- Account code entry on incoming call
- Hospitality Services (wake up call, etc)

Applications may be developed independently of the core ShoreTel solution. The only linkage is that there are several PAPI related parameters that need to be configured within ShoreWare Director that eventually find their way into the configuration files that the phones download on startup.

The applications themselves run on industry standard web servers such as Microsoft's IIS or Apache on Linux. The development process for developing applications in these environments is beyond the scope of this document.

PAPI supports two types of applications: Push and Pull. Push applications are initiated by the programming logic of the application residing on the web server. An emergency alert that appears on the LCD accompanied by an audio alert is an example of a push application. Pull applications are initiated by the phone user when they select an application from a list of available choices that are displayed when the user activates the phone's browser. The pull type of application model is similar to the model commonly used when browsing the web on a PC.

The native ShoreTel Phone API as described fully in the ShoreTel Phone API Development Guide was created to provide developers with the ability to create new applications targeted toward the ShoreTel phone family. The native Phone API gives developers control of certain phone functions such as the phone display, the soft keys, audible alerts, etc.



The native Phone API is available for use directly. Since its interface model is XML objects (strings) sent and received over HTTP, it is platform neutral as far as development is concerned. A developer can use C++ or C# to build the objects and can use IIS or Apache to send/receive the objects. However, the native Phone API must realistically include registration server and “object push server” functions. Developers would have to create these functions on their own. The tools for building native XML objects would also need to be developed.

The ShoreTel Phone API SDK provides a layer around the native ShoreTel Phone API thereby simplifying development of Phone API applications.

6. ScriptHlp SDK

The ScriptHlp SDK enables server scripts and applications to assign a ShoreTel user to his/her home phone, change a user’s Call Handling Mode, or change the group to which a user is assigned.

7. Developer Network Packages

The Developer Network provides access to documentation, tools, and sample code for the developer interfaces provided by the system. On the web site, developers will find multiple packages that encapsulate different components.

7.1. *TAPI Software Developers Kit (SDK)*

The TAPI SDK provides the documentation developers need to create both desktop as well as advanced, server based, multi-line applications. The SDK also includes the TAPI status program described above as well as other TAPI samples and tools.

7.2. *COM Object and Desktop Samples*

The COM Object and Desktop samples package includes the compiled multi-line control as well as the source code and documentation for the object. In addition, the package includes samples and development tools to assist in using the interface to create customized applications.

Make Call Test	Visual Basic 6.0 sample that launches a call to the specified extension using the STControl COM object.
Log Call Test	Visual Basic 6.0 sample that logs the caller ID and caller name for the last 10 calls received.

STControl Test	Test application that provides a graphical, call control application that illustrates the use of most of the functions supported by the COM object. The test application is provided in Visual Basic 6.0.
DDE Server	Test tool that logs and displays DDE events received from a sending application such as the STClientApp object.
ST Client App	Sample code that illustrates interactions required to connect telephony events to third party applications for screen pops or other activities.
STCall Handler	Sample code that illustrates monitoring call activity and triggering different call treatments based on the caller ID of the current call.

New Internet Explorer Click-to-Call Context Menu Desktop Application

A combination scripting sample and Internet Explorer add on which demonstrates adding support for a context menu feature to allow calling from Internet Explorer by selecting a number and then right clicking and selecting "ShoreTel Call" from the menu.

ShoreTel Client Application (STClientApp)

The Client Application sample illustrates the task of integrating the telephony events with a back-office application such as automatic customer look-ups for inbound calls (Screen Pops). The tool is designed to use the COM object and illustrates connecting programming events to telephony events. The sample program:

- Launches DDE events to external applications on call events
- Launches command line events to Windows™ on call events
- Can send multiple events for a call based on different call events
- Automates record look-ups (screen POP) in the GoldMine application

The STClientApp object is provided as both a compiled binary as well as complete Visual Basic 6.0 Source.

ShoreTel Call Handler Sample Application

The Call Handler application illustrates the use of call information to trigger different activities in the system. The application is configured with multiple caller ID strings with each configured to trigger different, programmatic activities including transfer to voice mail, transfer to another number or to simply play a wave file on the computer.

The call handler sample provides an example of using the multi-line control within a C#.NET development.

7.3. COM Object Server Samples

A sample package is provided that includes multiple tools and samples for development multi-line applications. The tools and samples are built using the COM object and illustrate using different parts of the interface for application development.

TAPI Browser	A development tool that illustrates the use of virtually every method supported by the multi-line control.
Junk Fax Screener	Watches for calls on one or more monitored extensions and automatically disconnects calls when the caller ID matches a configured junk-fax list.
Account Capture	Simply IVR application that illustrates how to interact with callers and collect input DTMF digits.
Account Player	Illustrates how to pass digits collected by one application (the Account Capture Sample), through the TAPI services, to a second application.
Prompt Recorder	An application that interacts with callers to either play a pre-recorded prompt illustrating the ability to play media, or to allow the caller to record a new prompt illustrating the record functionality.
Dial and Transfer	C++ sample that provides a dedicated interface to make a call between 2 passed phone numbers.
Dial & Transfer Test	Visual Basic 6.0 sample that illustrates using the dial and transfer sample.
Web Dial Sample	Sample implementation of HTML and ASP that illustrates using the Dial and Transfer Object from a web page.
Make Call Test	Visual Basic 6.0 sample that launches a call to the specified extension using the COM object.
Log Call Test	Visual Basic 6.0 sample that logs the caller ID and caller name for the last 10 calls received.



ST Call Logger

One of the pair of applications which compose the Call Quality Monitor Application Package together with the “ST Call Test” Utility below. Sample server code that monitors all active calls. When it detects a configured keystroke sequence it generates an event including call details to the Windows Event log. The log information can then be used to correlate to a bad call.

ST Call Test

One of the pair of applications which compose the Call Quality Monitor Application Package together with the “ST Call Logger” Utility above. Connects to the ST Call Logger and uses it to place a call from a station to a specified number. Useful for remote testing of audio paths for a specific call.

New Pre-Recorded Prompts for Server Applications

A set of common professionally recorded prompts are provided for building voice response type applications.

ShoreTel Dial and Transfer Object (STDialXfer)

The ShoreTel Dial and Transfer Object illustrates how to telephony enable a web site. The object can be used in ASP and other COM capable environments to enable applications to create a call between 2 extensions in the ShoreTel system.

The application is sent 2 numbers. It launches a call to the first number (Dials) and when that party answers, connects the call to the second extension (Transfer). An example would be enabling a live phone directory in a corporate intranet portal where clicking on a name in the directory launches a call to the person selected. Optionally, calling to a mailbox instead of an extension is also illustrated.

COM Object Salesforce.COM Sample

The Salesforce sample is an application that illustrates the use of the SDK to automate routine user interactions with a customer relationship management package. The application illustrates:

- Searching of the customer relationship management system to find a matching customer record based on caller and then automatically opening the application to the customer’s record.
- Providing an interface inside a web browser to automate out-bound dialing of a selected customer contact with a simple, click-to-dial interface.

7.4. ShoreTel Web Service Software Development Kit (SDK)

The ShoreTel Web Service SDK provides the documentation and samples to enable application developers to implement ShoreTel Unified Communications functionality into their own applications using platform independent web services technology.

The package also includes samples using different programming language to illustrate platform independent consumption of the STWS SDK from commonly used development tools.

There are following STWS SDK client samples:

- **ShoreTel Web SDK Browser**
This is the largest and most comprehensive of the samples. It provides a very useful tool as well as demonstrating the use of virtually every method and event supported by the STWS SDK.
- **ShoreTel Perl Script Sample**
This set of Perl script samples illustrates making a call and monitoring call status using the STWS SDK.
- **ShoreTel Ruby Script Sample**
This is a Ruby language sample illustrates making a call and monitoring call status using the STWS SDK.
- **ShoreTel User Active Call Handling Mode Management and Monitoring Sample**
This is a C# sample that demonstrates doing login as a ShoreTel user and managing as well as monitoring the user's active Call Handling Mode (CHM).

7.5. ShoreTel Phone API Software Development Kit (SDK)

The ShoreTel Phone API SDK provides a layer around the native ShoreTel Phone API thereby simplifying development of Phone API applications. The Phone API SDK includes a Windows service, which acts as the registration and object push server providing an interface for developers to keep track of the Phone API capable phones on the ShoreTel system as well as pushing Phone API objects to them.

A class library is also included in the Phone API SDK to simplify the creation of the native Phone API XML objects needed to control the phones. In addition, sample applications illustrate the use of the SDK and provide code that can be "cut and pasted"



to jump start development of new applications. To use the Phone API SDK, software development must take place using Microsoft's .NET development environment.

7.6. ScriptHlp SDK

ScriptHlp SDK enables server scripts and applications to assign a ShoreTel user to his/her home phone, change a user's Call Handling Mode, or change the group to which a user is assigned. The SDK includes sample code, the "STScriptHlpTest sample", to show how to use the COM object from a C# .NET object. The concepts would be similar in other programming languages including Visual Basic and C++, for example.