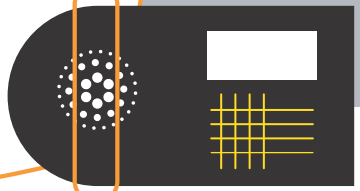


PRODUCT DOCUMENTATION

Contact Center

Applications Interface
Guide

RELEASE 5.1



Document and Software Copyrights

Copyright © 1998–2009 ShoreTel, Inc. All rights reserved. Printed in the United States of America. Contents of this publication may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without prior written authorization of ShoreTel, Inc.

ShoreTel Inc. reserves the right to make changes without notice to the specifications and materials contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the materials presented.

Trademarks

ShoreCare[®], ShoreWare[®], and ShoreGear[®] are registered trademarks of ShoreTel, Inc. in the United States and/or other countries. ShoreTel, ShorePhone, Office Anywhere, and ShoreTel Smart are trademarks of ShoreTel, Inc. in the United States and/or other countries. All other copyrights and trademarks herein are the property of their respective owners.

Patents

ShoreTel's products are covered by one or more of the following patents: United States Patent 6,996,059, United States Patent 7,003,091, and United States Patent 7,167,486. ShoreTel, Inc. All rights reserved.

Version Information

CRM Integration Guide
Contact Center 5.1
Part Number: 800-1260-01
Version: Integration_51_October 6, 2009

Company Contact Information

ShoreTel, Inc.
960 Stewart Drive
Sunnyvale, California 94085

(408) 331-3300
(408) 331-3333 fax

www.shoretel.com

Table of Contents

PREFACE	5
CHAPTER 1 USING CALL PROFILE INFORMATION	7
Overview	7
CRM Integration with Call Control Scripts	7
Call Profiles	7
Call Control Scripts	9
Integrating to CRM Databases	10
Call Routing Based on CRM Information.	10
Self-Service Based on CRM Information.	11
Extracting and Storing Call Profile Information in TAPI.	11
CHAPTER 2 OUTBOUND CALLING WITH DIAL LISTS	13
Overview	13
Configuring Dial Lists	13
CHAPTER 3 AUTOMATING AGENT ACTIVITIES	15
Overview	15
The Agent INI File (ETAS.INI)	15
Using Screen Pops	17
Integrating Applications by DDE	17
Introduction	17
First Invocation of ShoreWare Agent Toolbar	18
Configuring ShoreWare Agent Toolbar	18
ShoreWare Agent Toolbar as a DDE Client	19
ShoreWare Agent Toolbar as a DDE Server	20
Advanced Configuration Options.	22
Integrating Applications by ActiveX	23
Configuring ShoreWare Agent Toolbar	24
Managing ShoreWare Agent Toolbar	24
ShoreWare Agent Toolbar Events	25
ShoreWare Agent Toolbar Methods (Services)	25
Activating Applications by Triggers.	28
Introduction	28
Configuring ShoreWare Agent Toolbar	29
Activating an Application	29
Activation Events	29
Command Substitution	29

APPENDIX A	INTEGRATION EXAMPLES	31
	Overview	31
	Web Application by Triggers	31
	Vantive by DDE	31
	MSAccess by DDE	31
	External Application by ActiveX	32
	Code of VbController.exe	32
APPENDIX B	ACTIVEX API QUICK REFERENCE	37
	Overview	37
	Functions	37
	Events	38
APPENDIX C	SUPPORTED SQL SYNTAX	39
	Overview	39
	All simple statements	39
	Statements with computed fields	39
	Joint statements	39
	Nested Statements	39
	Scripts	40
	SQL Syntax Format Limitations	40
INDEX		41

Preface

Objectives

This guide describes how to integrate the ShoreTel Enterprise Contact Center Solution with external applications, including Customer Relationship Management (CRM) products. It details the available interfaces and provides necessary reference information, including examples.

Note: The options and modules required for application integration are only available with the ShoreTel Enterprise Contact Center Solution.

This guide provides information for professional services engineers and application programmers who must plan the integration of the ShoreTel Enterprise Contact Center Solution to selected external applications before development begins.

Audience

This guide assumes that you are familiar with the ShoreTel Enterprise Contact Center Solution architecture, features, and functionality. You should also be familiar with ShoreWare Agent Toolbar. See the *Using ShoreWare Agent Toolbar* guide for more information.

Organization

This document is generally organized into major tasks, presented in the order in which they should be completed.

Documentation Overview

The following documents offer additional information about the ShoreTel Contact Center system:

- Release Notes
- Installing Guide
- Contact Center Administrator Guide
- Using Reports
- Installing and Implementing Enterprise Contact Center Chat
- Supervisor User Guide

- Using ShoreWare Agent Toolbar
- Context-Sensitive Online Help

ShoreTel Contact Center documentation can be found in the Documentation folder of the ShoreTel Contact Center Solution Installation CD and on the ShoreTel web site. The context-sensitive online help can be accessed from the Help menu of each Contact Center application.

Document Conventions

Conventions used in this guide include:

- Data entry field names, hypertext links, control buttons, keywords, and other items within the system management interface are in **boldface** text.
- Information that you enter in data entry fields is in a `data_entry` font.
- **NOTE:** indicates an area of special interest to the user.

Disclaimer

The illustrations, telephone displays, and screen captures in this guide are used to show ShoreWare Contact Center Director features and controls. What appears in the illustrations may differ from the actual display since the available functions and information differ depending on the telephony state. The names of companies, products, people, characters, and/or data in the illustrations are fictitious and in no way represent any real individual, company, product, or event, unless otherwise noted.

Using Call Profile Information

Overview

To automate your ShoreTel Contact Center operation, you can integrate ShoreTel Enterprise Contact Center Solution with your Customer Relationship Management (CRM) application using call control scripts.

In addition, you can extract and store call profile information into TAPI.

CRM Integration with Call Control Scripts

Contact Center ties more closely to your customer information, improving your customer service. For example, you can use your CRM database to change skills, priority, and call flow, and to let callers bypass agents and interact directly with the CRM database in a self-service application.

Integration requires the interaction of three key components:

- Call profiles
- Call control scripts
- CRM database

These components work together in call routing and self-service applications.

Call Profiles

The call profile contains a list of fields, each with a field name and value. Mandatory fields are defined and managed by the Contact Center, and user-defined fields can be created in the Contact Center Director according to your business needs.

The call profile fields can be updated by caller interaction or by a CRM database that uses call control scripts. With call control scripts, you can set call actions and decisions based on the contents of selected call profile fields.

For detailed information about managing call profile fields, see the *ShoreTel Contact Center Administration Guide*.

The following is a list of mandatory fields in the call profile. The field names in the call profile are case-sensitive.

Name	Description	DDE Name
Group	The internal group id related to the call	acd_group
Original Destination		called
Trunk	PBX and call-type depended	trunk
Calling	ANI (caller ID) of caller	calling
Called	Agent extension	

Name	Description	DDE Name
Elapsed	Time since call entered the ShoreTel system (hh:mm:ss)	
Last Redirection	Last device connected to the call	
Originally Called	Dial Number originally called (typically the IRN)	
Status	Current state of the call (connected or hold)	
Time	Current station time (hh:mm:ss)	
Trunk	PBX and call-type depended	
Waiting Time	Time the call waited in the queue	
Type	Call type (1=voice, 2=chat, 3=mail, 4=callback, 5=abandoned, 6=Web_callback, 7=dial from list)	calltyp
Media	Call media (1=voice, 2=chat, 3=email)	Media
Priority	Priority of call	
Service	Service that handles the call	Service
CUSTOMER_NUMBER	Identifier number of the customer (if internal, the Customers Table is used)	Customer Number
CUSTOMER_NAME	Customer name (if internal the Customers Table is used)	Customer Name
ACD_ENTER_DATE	Date the call entered Contact Center	ACD Enter Date
ACD_ENTER_TIME	Time the call entered Contact Center	ACD Enter Time
Q_POSITION	Call position in the queue	Queue Position
AVERAGE_Q_TIME	Expected average wait time in the queue.	Average Queue Time
CALLBACK_TIME	Callback time set by caller (if empty, Abandoned Call)	Call Back Time
CALLBACK_DEST	Callback destination	Call Back Destination
LANGUAGE	The language associated with the call (0=English, 1=Hebrew, 2=Russian, 3=Arabic, 4=Mexican Spanish)	Language
AGENT_NUMBER		Agent Extension
AGENT_NUMBER	Number used to identify the agent	Agent Number
DIAL_LIST_ID		Dial List ID
START_QUEUE_TIME		Start Queue Times
AGENT_NUMBER	Agent number	Agent Number
AGENT_PSW	Agent password	Agent PSW
AGENT_EXT	Agent extension	Agent Extension
EXECUTE_REQ	For any request to be executed	Execute Req
TRUNK_NUMBER		Trunk Number
EMAIL_TO	Email to field	Email To
EMAIL_CC		Email CC
EMAIL_SUBJECT	Email subject field	Email Subject
EMAIL_FROM	Email from field	Email From
EMAIL_REPLY_TO	Email Reply-to-field	Email Sent Date
EMAIL_SENT_DATE	Email Sent date filed	Email Sent Time
EMAIL_SENT_TIME	Email Sent time field	Email To
EMAIL_SENT_OMS_DATE		Email Enter OMS Date
EMAIL_SENT_OMS_TIME		Email Enter OMS Time

Name	Description	DDE Name
PRIMARY_CB		Primary Call Back Destination
ALTERNATIVE1_CB		Alternative Call Back Destination 1
ALTERNATIVE2_CB		Alternative Call Back Destination 2

Call Control Scripts

Call control scripts define the way a system processes automatic call distribution (ACD) calls. The scripts are used to process calls, present information to callers or prompt them for input, collect input digits, and query the organization's database. Call control scripts can be used to:

- Make announcements
- Collect caller information
- Make routing decisions
- Provide self-service
- Provide music-on-hold
- Read and write information to the database

There are two call control script tools available: a graphical call control scripting application called Graphical Call Control Scripts Director (GCCS Director) and Call Control Scripts Director (CCS Director). You can create specific scripts from the set of actions available in the CCS Director or GCCS Director.

A call control script contains a set of actions that are performed on a call. You can change the agent requirements or call flow based on database interactions. Call control scripts can access external databases based on caller-supplied input. Information from the database can be used to modify the call routing or play back information to the caller.

The script's defined actions control the flow of a call to its routed destination. They can also update the call profile fields that control the call handling as well as the agent's display. Some of these actions are logical (for example, Logic Switch), and some are related to telephony (for example, Transfer). Some actions obtain information from the organization's database (for example, SQLQuery), while others require Interactive Voice Response (IVR). When the Contact Center executes IVR actions, such as Menu, Get Digits, and Play File, the call is automatically transferred to an available IVR port.

To use call control scripts:

- Step 1** Plan the actions you want the call control script to perform.
- Step 2** Identify and connect to an external database and create open database connectivity (ODBC) queries.
- Step 3** Create any new call profile fields needed by the call control script.
- Step 4** Record the required announcements.
- Step 5** Create the call control scripts by using CCS or GCCS Director.
- Step 6** Set IVR application parameters from the IVR Application window.
- Step 7** Set the call control scripts as IRN or service destinations.

Integrating to CRM Databases

To integrate your call control scripts with customer information, you must first identify the database and its columns containing the necessary information. When integrating with an external database, the Contact Center acts as the client and uses an ODBC interface on the Contact Center Server to interact with the database.

You must have the appropriate ODBC drivers installed on the Contact Center Server.

To query or do other database interactions, the Contact Center uses Structured Query Language (SQL). External databases must be in SQL format to be accessible by the Contact Center through ODBC.

To integrate to an external database:

- Step 1** Make sure that you have a network connection to the external database by using *ping*.
- Step 2** Install all needed infrastructure software (ODBC drivers and client side processes) and create the ODBC connections on the Contact Center server.
- Step 3** Plan the data items you need to gather, define the conditions, and decide where to put data results.
- Step 4** Define the Data Source Name (DSN) of the database to which you want to connect on the Interface tab of the System configuration window.
- Step 5** After the interface is created, restart the Enterprise Contact Center.
- Step 6** Create the SQLConnect and SQLExecute actions in your call control script using GCCS Director.
 - SQLConnect connects to a database in order to make database queries.
 - SQLExecute sends a query (written as a SQL statement) to the database to obtain specific information, such as the priority of a caller. For example, the query may be able to find and retrieve the priority of the caller, enter the call profile, and update this value.

NOTE: Basic SQL queries such as **SELECT** and **INSERT** are supported. Stored procedure calls are not supported.

Call Routing Based on CRM Information

You can use the data in your customer database to route calls to agents who can better serve a particular customer. The following example routes an incoming call based on a customer type field in the CRM database.

In this scenario, a call arrives at a financial institution and is routed by customer type to either a group of agents handling stock trades or to a group that specializes in bonds. A field for customer type has been added to the call profile.

The incoming call follows these basic stages:

1. The call arrives at the IRN and is sent to a call control script.
2. The call control script uses the Get Digits action to collect the Customer ID number and record it to the call profile.

3. The call control script assesses the database in the following interaction:
 - a. Connects to the CRM database using the SQLConnect action.
 - b. Uses the Query Database action to identify this customer's data in the database.
 - c. Uses the SQLExecute action to retrieve the customer type from the CRM database and write to the call profile.
4. The call control script uses the Decision action to check the Customer Type field and branch to the correct Change Profile action that sets the required service in the call profile.

Self-Service Based on CRM Information

You can use the data in your customer database to provide customers with self-serve options. The application can play back selected WAV files based on the customer status or convert text digits, currency, or dates into speech. The example below provides a balance to a self-service customer.

In this scenario, a customer calls a financial institution and subsequently hears options for checking an account balance. A field for customer balance has been added to the call profile.

The incoming call follows these basic stages:

1. The call arrives at the IRN and is sent to a call control script.
2. The call control script uses the Get Digits action to collect the Customer ID number and records it to the call profile.
3. The call control script accesses the database in the following interaction:
 - a. Connects to the CRM database using the SQLConnect action.
 - b. Uses the Query Database action to identify this customer's data in the database.
 - c. Uses the SQLExecute action to retrieve the customer's account balance from the CRM database and write to the call profile.
4. The call control script uses the Announce action to play back the account balance to the caller.

Extracting and Storing Call Profile Information in TAPI

You can extract and store call profile information in TAPI by using the following process:

1. On the Contact Center Server, in the directory in which ShoreTel Contact Center is installed, use a text editor to create a file named **shoretelcfg.ini**.
2. In the shoretelcfg.ini file, add a section named **call_profile**.
3. In the call_profile section, specify a key named **user_fields** with call profile field names as values. Separate the values with a comma. For example:

```
[call_profile]
user_fields=AccNo, Balance, DueDate
```
4. Restart the Contact Center Server.

The specified call profile field values display in the TAPI call properties.

Outbound Calling with Dial Lists

Overview

Enterprise Contact Center can automatically make outbound calls for agents based on lists of numbers. Calls made from the lists are placed by the system and transferred to an agent. In a mode called Preview Dialing, the outbound call is only placed after an available agent is identified to service the call. After the call is placed, while it is proceeding through the network or ringing, it is transferred to a configured destination, either the reserved agent or to a call control script.

Since the call is transferred while ringing, ShoreTel recommends that calls only be transferred to an agent. The script begins playing any prompts immediately, and the recipient may not hear the first part of a script's prompt or message.

Use the Outbound tab of the System configuration window to define various system parameters for handling outbound calls.

Configuring Dial Lists

Dial lists are sets of telephone numbers that are used for outbound calls, according to rules defined in the system. The Dial Lists configuration window enables you to configure the system's rules for the dial lists. The Contact Center allows dialing from multiple dial lists at the same time.

After the agent receives the call, the agent can mark the call complete (reached a person) or incomplete/try later from a button on ShoreWare Agent Toolbar. To prevent incorrect numbers from staying on the dial list, the agent marks the call incomplete/done. You must define a location in the database for storing the call results.

You have the option of defining priority for the inbound calls versus the outbound dialing from a list. You can control the activation of the specific dial list according to:

- **Simultaneous pending dial numbers in queue:** This is used to control the load of each dial list when the agents in the specific group related to the dial list campaign are also active in another group's activity (such as other dial-lists, inbound or callback calls).
- **Statistical TSF conditions:** The system will activate the dial list only if the Target Service Factor (TSF) is equal to or higher than the percentage defined.

Calls automatically dialed from the list can pass to the agent after the called party answers or while the call is in ringing mode. If the call is passed to the agent while in ring mode, the agent must manually set the call status (ring no answer, FAX, answering machine, and so on).

External Databases

To use dial lists for outbound dialing campaigns, you must have the dial data in an ODBC/SQL database.

To integrate to an external database:

- Step 1** Create the physical connection to the external database. Make sure you can ping between the computers and that they can communicate via the network.
- Step 2** Install the needed infrastructure software (ODBC drivers and client side processes) and create the ODBC entries on the ShoreWare Contact Center Server.
- Step 3** Plan which data items you would like to gather, define the conditions, and decide where to put them in the call control script.
- Step 4** Define the Data Source Name (DSN) of the database to which you want to connect on the Interface tab of the System configuration window.
- Step 5** After you define a new ODBC connector in the interface, restart the Enterprise Contact Center.
- Step 6** Create the SQLConnect and SQLExecute scripts in Call Control Script Director.
 - SQLConnect connects to a database in order to make database queries.
 - SQLExecute sends a query (written as a SQL statement) to the database to obtain specific information, such as the priority of a caller. For example, the query may be able to find and retrieve the priority of the caller, enter the call profile, and update this value.

See Appendix C for more information on ShoreTel supported SQL Syntax.

Output Source

The system can update the results of the outbound dialing to the external database. The database must be an ODBC/SQL database.

The updates can be according to:

- Success
- Single failure (still under retries)
- Final failure

To define various parameters for dial lists, use the Interfaces tab of the System configuration window.

Dial lists should be in the form of a table in a SQL database. More information about Dial lists is available in the Contact Center Administrator Guide - *“Configuring Dial Lists”*.

Automating Agent Activities

Overview

The ShoreTel Contact Center Solution provides interfaces for client-based integration with external applications.

The call profile (a table) contains the information related to a call. The call profile includes all of the call's data in mandatory fields (such as DNIS, ANI, Initiate Time, and Priority) and user-defined fields. The call profile originates when the call enters the ShoreTel system, is accessible and managed during the progress of the call, and is cleared at call completion.

When a call is routed to an agent, ShoreWare Agent Toolbar uses call profile values to provide the appropriate notifications to the agent. Agent Toolbar offers tools for delivering specific fields to external applications that are available at the agent station along with executable commands. For example, the system can send a command to display the caller's screen, based on the Caller ID (ANI).

Contact Center supports standard interfaces for integration with external applications, including dynamic data exchange (DDE), ActiveX, and basic Microsoft prompt commands (called "triggers"). Customization using these interfaces is done through the agent configuration file ETAS.INI.

The Agent INI File (ETAS.INI)

Each agent in the Contact Center has an INI file that includes the application setup definitions and maintains the agent's personal workstation information, along with other valuable data that relate to features, customization, and integration with external applications. The file named ETAS.INI is formatted as a standard Windows INI file. Each agent's ETAS.INI file is located in the Contact Center Server system directory ...\\ShoreTel Contact Center Server\\Agents\\<agent ID>\\subdirectory. Agents must exit ShoreWare Agent Toolbar before you make changes in their ETAS.INI file; otherwise your changes will be overwritten.

The integration data appears in the Microsoft triggers and DDE paragraphs as shown in the following example:

```
[TRIGGERS]
OperateTriggers=1
[TRIG_OnIncoming]
Command=MyApplication.exe %calling %DNIS %Service %callid
[TRIG_OnConnected]
Command=notepad.exe %service.txt

[DDE]
Server=CRM
Topic=Events
[DDE_OnStarted]
```

```

Command=started
[DDE_OnReStarted]
Command=restarted
[DDE_OnDenied]
Command=denied %cause
[DDE_OnStoped]
Command=stopped
[DDE_OnLoggedIn]
Command=loggedin %group
[DDE_OnLoggedOut]
Command=loggedout %group
[DDE_OnHeld]
Command=held %callid
[DDE_OnRetrieved]
Command=retrieved %callid
[DDE_OnIncoming]
Command=incoming %callid, %Calltyp, %DNIS, %calling, %Service,
%Priority, %TimeLn
[DDE_OnConnected]
Command=connected %callid, %Calltyp, %DNIS, %calling, %Service,
%Priority, %TimeLn
[DDE_OnConferenced]
Command=conferenced %callid
[DDE_OnReleased]
Command=released
[DDE_OnResumed]
Command=resumed
[DDE_OnWrapUp]
Command=wrapup
[DDE_OnReady]
Command=ready
[DDE_OnCleared]
Command=cleared %callid

```

For each integration type, a list of agent states exists along with related actions (mainly commands) that are activated at that state.

The available states are detailed later in this document. A command might include free text, parameters prefixed by the percent (%) sign, and control codes. A parameter can be any call profile field name available in ShoreWare Agent Toolbar in either the mandatory or user-defined fields.

Upon a state change, the command related to that state is evaluated, and all parameters and control codes are substituted with their appropriate values.

Parameter/Code	Substitution value
<i>%call_profile_field</i>	Call profile field value
\\	\ (Backslash)
\\t	TAB
\\b	BACKSPACE
\\r	CR
\\n	Newline

Parameter/Code	Substitution value
\f	Formfeed
\a	Alert (Beep)
\xHH	Character HH (HH= hexadecimal value of character)

Using Screen Pops

The [ScreenPops] section of the ETAS.INI file acts as a filter for the DDE and triggers specified to limit screen pops. Initially, this section does not exist in the ETAS.INI file, so the system defaults to displaying screen pops for all types of contacts (ACD, non ACD, voice, chat and web callbacks, and email). You can configured Contact Center to only display screen pops for ACD, voice, chat and web callback, or email contacts. This is done by adding a [ScreenPops] section to the ETAS.INI file with the following appropriate parameter(s):

For ACD calls only:

Add the parameter TRIG_OnACDOnly=YES

To not display screen pops for voice contacts:

Add the parameter TRIG_OnVoiceMedia=NO

To not display screen pops for chat and web callback contacts:

Add the parameter TRIG_OnChatMedia=NO

To not display screen pops for email contacts:

Add the parameter TRIG_OnEmailMedia =NO

For example, the following additions to the ETAS.INI file would trigger screen pops for ACD calls only (and not for email and chat and web callback contacts):

```
[ScreenPops]
TRIG_OnACDOnly=YES
TRIG_OnEmailMedia =NO
TRIG_OnChatMedia=NO
```

Agents must exit ShoreWare Agent Toolbar before you make changes in their ETAS.INI file; otherwise your changes will be overwritten.

Integrating Applications by DDE

The DDE interface is provided by ShoreWare Agent Toolbar to external applications, usually CRM applications.

Introduction

The ShoreWare Agent DDE interface enables external applications to interact with ShoreWare Agent Toolbar.

The applications can get notifications from the Agent DDE when an important event occurs, and they can control some of the telephony and automatic call distribution (ACD) functions that ShoreWare Agent Toolbar supports.

Example: An application working as a DDE server and as a DDE client can perform the functions in the following scenario from the application:

When a call is delivered to an agent and the phone rings, the Agent DDE connects to the application working as a DDE server and invokes an EXECUTE command on the DDE protocol.

1. The application gets the information and the parameters from the Agent DDE event. At this time, the application can perform its functions, such as displaying the customer screen and updating the database.
2. The application provides a button to the agent to answer the call. When the agent clicks the button, the application contacts ShoreWare Agent Toolbar (now working as a DDE server) and invokes an EXECUTE command on the TEL topic requesting to answer the call.
3. The call is answered, and the agent handles the call.
4. The application provides another button for the agent to end the call. When the agent clicks the button, the application contacts ShoreWare Agent Toolbar (now working as a DDE server) and invokes an EXECUTE command on the TEL topic requesting to end the call.

The Contact Center gathers information related to a call from the moment the call enters the system until it is concluded. All this information is stored in the call profile.

ShoreWare Agent Toolbar makes all of the call profile information available to the external application. The application works as a DDE client and DDE server simultaneously.

First Invocation of ShoreWare Agent Toolbar

Any Windows application can start ShoreWare Agent Toolbar by using the following command:

```
ShoreWareAgent.exe 0 AgentId AgentPwd
```

Where:

0 = the number zero
AgentId = the agent ID number of the agent
AgentPwd = the agent's password

Configuring ShoreWare Agent Toolbar

ShoreWare Agent Toolbar provides maximum flexibility during integration. The configuration of the commands should be done at the time the system is installed. Typically, the person installing Contact Center also configures this application.

When ShoreWare Agent Toolbar works as a DDE client, you can configure the following general parameters:

- Server: Name of the server application.
- Topic: Topic to be used by ShoreWare Agent Toolbar when executing a command on the external application.
- Command: Command and parameters to send to the application when a specific event occurs. This command is defined per event type.

ShoreWare Agent Toolbar as a DDE Client

ShoreWare Agent Toolbar can execute commands on an external application when any of the following events occur:

- DDE_OnIncoming: When the call rings at the agent's station.
- DDE_OnConnected: When the call is answered by the agent.
- DDE_OnCleared: The call was terminated.
- DDE_OnHeld: The current call was held.
- DDE_OnRetrieved: The call was retrieved from hold.
- DDE_OnStarted: ShoreWare Agent Toolbar starts to work.
- DDE_OnRestarted: ShoreWare Agent Toolbar restarts.
- DDE_OnDenied: ShoreWare Agent Toolbar cannot work.

The command is a string that is sent "as is" to the external application. To provide parameters, the % sign must precede the field name in the call profile.

Example: To indicate to a CRM application that a call has been delivered to the agent (and is ringing), you may use the following command:

```
[DDE_OnIncoming]
command=Incoming from=%calling group=%acd_group
```

When a call comes in from number 7410953 to the Sales group, the following command is sent to the external application:

```
Topic=ShoreWareAgent
DDE_Command= DDE_EXECUTE
Command= Incoming from=7410953 group=Sales
```

The following rules apply:

- All of the commands and parameters are not case-sensitive.
- If a parameter requested in the command has no value, a blank is sent.
- The structure of commands and events is defined for each installation. The person installing ShoreWare Agent Toolbar performs the agent configuration.
- The configuration file ETAS.INI is stored on the Contact Center Server.

DDE Commands

The DDE commands that can be used:

Event	Parameter	Default	Description
DDE_OnIncoming	Command	Incoming %CallId, %Calling, %Service, %Group	A call is ringing at the agent phone.
DDE_OnConnected	Command	Connected %CallId, %Calling, %Service, %Group	The agent answers the call.
DDE_OnCleared	Command	ConnCleared %CallId	The call terminates.
DDE_OnRelease	Command	Released	Agent was released.
DDE_OnResume	Command	Resumed	Agent was resumed.
DDE_OnWrapUp	Command	WrapUp	Agent goes to wrap-up.
DDE_OnReady	Command	Ready	Agent returns from wrap-up.
DDE_OnLogin	Command	LoggedIn	Agent logged in.
DDE_OnLogout	Command	LoggedOut	Agent logged off.

Event	Parameter	Default	Description
DDE_OnHeld	Command	Held %CallId	A call was held.
DDE_OnRetrieved	Command	Retrieved %CallId	A call was retrieved.
DDE_OnStarted	None	Started	ShoreWare Agent Toolbar started.
DDE_OnRestarted	None	Restarted	ShoreWare Agent Toolbar lost connection to the server, and now it is online again.
DDE_OnDenied	Cause	Denied %cause	ShoreWare Agent Toolbar is unable to work.

The possible OnDenied causes:

Cause	Name	Description
1	LOGGED_ON_TO_ANOTHER_EXT	Agent is already logged in to another extension.
2	NEW_AGENT_ON_EXT	Another agent is logged in to current extension.
3	WRONG_AGENT	Wrong agent ID is supplied.
4	WRONG_PASSWORD	Wrong password is supplied.
5	MAX_NUM	Maximum number of agents in system is exceeded.
6	FAILED_TO_READ_FROM_DBS	Internal error.
7	STILL_INITIALIZING	ShoreWare Agent Toolbar cannot start because the system is initializing.

More than one active call can occur at the same time. The call ID values are important.

ShoreWare Agent Toolbar as a DDE Server

Working as a DDE server, ShoreWare Agent Toolbar supports four topics:

- ACD for ACD-related functions
- TEL for all telephony-related functions
- REQUESTS to get information from ShoreWare Agent Toolbar
- OTHER for miscellaneous commands

The ACD Topic

The commands that are invoked using DDE_EXECUTE:

Name	Parameters	Description
LoginPrimaryGroups	None	Logs in to all primary groups.
LogoutFromPrimaryGroups	None	Logs off from all primary groups.
LoginGroup	Group name	Logs in to a specific group.
LogoutFromGroup	Group name	Logs off from a specific group.
Release	None	Makes agent enter “release” state.
Resume	None	Makes agent enter “resume” state.
Wrapup	Wrap-up code	Enters wrap-up code.
AgentReady	None	Terminates wrap-up time.

Name	Parameters	Description
TransferToAgent	Agent number	Transfers call to specified agent.
Help	None	Help request from supervisor.
SetCallProfile	Field name, new field value	Sets call profile field value.

DDE_REQUEST supports the CommandList command, which has no parameters. It gets a space-delimited list of supported commands for the topic.

The following rules apply:

- All of the parameters are strings.
- The commands and parameters are not case sensitive.

The TEL Topic

The commands that are invoked using DDE_EXECUTE:

Name	Parameters	Description
Alternate	None	Exchanges between the call on hold and the active call.
Answer	None	Answers the call that is ringing.
HangUp	None	Terminates the current call.
ClearConnection	Call ID	Terminates the call with the selected Call ID.
Divert	Destination dial number	Diverts the call to the desired destination before answering the call.
Hold	None	Puts the current call on hold.
Retrieve	None	Retrieves a call that was previously put on hold.
MakeCall	Destination dial number	Makes an outgoing call.
TransferCall	Destination dial number	Starts the transfer process.
CompleteTransfer	None	Removes the agent from the call, completing the transfer.
Conference	Destination dial number	Begins a conference call.
CompleteConference	None	Completes the conference call.
SingleStepTransfer	Destination dial number	Transfers the current call to another telephone.
Reconnect	None	Returns the agent to the original call and updates the agent state.

DDE_REQUEST supports the CommandList command, which has no parameters. It gets a space-delimited list of supported commands for the topic.

The REQUEST Topic

The commands that are supported using DDE_REQUEST:

Item name	Parameters	Returns (string)	Description
TopicItemList	None	List	Gets a space-delimited list of supported commands for the topic.
IsPhoneBusy	None	TRUE or FALSE	TRUE if the phone is busy.
HeldCallsCount	None	Number of calls on hold	Number of calls currently on hold () or 1.
GetCpParam	Field name	Field value	Retrieve the value of a specific call profile field.

Item name	Parameters	Returns (string)	Description
IsRelease	None	TRUE or FALSE	TRUE if the agent is in the release state.
IsLoggedIn	Group Name	TRUE or FALSE	TRUE if the agent is logged in.
IsWrapUp	None	TRUE or FALSE	TRUE if the agent is currently in wrap-up time.

The OTHER Topic

- DDE_EXECUTE supports the Terminate command, which has no parameters. It closes ShoreWare Agent Toolbar.
- DDE_REQUEST supports the CommandList command, which has no parameters. It gets a space-delimited list of supported commands for the topic.

Advanced Configuration Options

When integrating to other applications, the basic DDE client behavior of ShoreWare Agent may not be enough. There may be situation where it is necessary to have more control on the DDE commands sent by the Agent application to the CRM application. This section contains information on advanced DDE commands.

Using DDE Poke instead of DDE Execute

Some applications require the use of the DDE Poke command instead of the standard DDE Execute used by Agent application.

In order to support this, you can replace the full command line following the desired event with the following:

```
Poke(destination)=Value
```

As an example of this usage, let's assume that it is necessary to set the ANI number of the caller for the current call on some MS Excel™ sheet when a call is answered by the agent. This can be done with the following lines inside the ETAS.INI file:

```
[DDE_OnConnected] Poke(Excel,Sheet1,rc)=%calling
```

Using multiple DDE commands for the same event

Some applications require the use of multiple DDE commands on response for a specific event.

In order to support this, you can replace the full command line following the desired event with a list of DDE commands to perform. The DDE commands Poke and Execute can be used.

An example of managing MS Excel this way can be found on the following example:

```
[DDE_OnIncoming]
Poke(Excel,Sheet1,rc)=%calling
Execute(Excel,Sheet1)={RIGHT}
Poke(Excel,Sheet1,rc)=%Service
Execute(Excel,Sheet1)={RIGHT}
Poke(Excel,Sheet1,rc)=%ACD_ENTER_DATE
Execute(Excel,Sheet1)={RIGHT}
Poke(Excel,Sheet1,rc)=%ACD_ENTER_TIME
Execute(Excel,Sheet1)={RIGHT}
Poke(Excel,Sheet1,rc)=David Young, ID 555
Execute(Excel,Sheet1)={RIGHT}
Execute(Excel,Sheet1)={DOWN}
```

```
Execute(Excel,Sheet1)={HOME}
```

In this case, Agent application will enter several values into several different Excel cells when the call rings at the Agent's phone.

Working with more than one DDE Server application at the same time

In certain situations it is necessary to send DDE commands to more than one application as a result for a specific trigger.

The following example illustrates how to set up the response for the DDE_OnIncoming event on the ETAS.INI file to set a value on an Excel Sheet and set the URL for Internet Explorer to www.shoretel.com and finally, moving the cursor to a different location on the Excel sheet.

```
[DDE_OnIncoming]
Poke(Excel,Sheet1,rc)=%calling
Execute(IEExplore,WWW_OpenURL)="www.shoretel.com,,0"
Execute(Excel,Sheet1)={DOWN}
Execute(Excel,Sheet1)={HOME}
```

Integrating Applications by ActiveX

ShoreWare Agent Toolbar can integrate by using the ActiveX control that comes with the Contact Center. It describes the interface control, available methods, parameters, and additional information required to perform the integration.

This guide assumes that the programmer is familiar with ActiveX specifications and use. ShoreWare Agent Toolbar ActiveX control enables external applications to use ShoreWare Agent Toolbar to perform telephony and ACD functionality.

The applications can get events from ShoreWare Agent Toolbar when some telephony, ACD, and management events occur, and can control the telephony, ACD, and management functions supported by the application.

Example: The following scenario details the process of an incoming call.

1. The moment a call is delivered to an agent and the phone rings, ShoreWare Agent Toolbar initiates an OnIncoming event to the application. The event supplies standard parameters (Call ID, Call type, DNIS, ANI, Service, Priority, and Time).
2. The application can get more call parameters from ShoreWare Agent Toolbar by getting properties (call profile fields).
3. At this time, the application can perform its functions, including displaying the customer screen and updating the database.
4. The application provides a button to the agent to answer the call. When the agent clicks the button, the application commands ShoreWare Agent Toolbar (using the Answer method) to answer the call.
5. The call is answered, the application gets an OnConnected event, and the agent handles the call.
6. The application provides another button to the agent to disconnect the call. When the agent clicks the button, the application commands ShoreWare Agent Toolbar to disconnect the call.

The Contact Center gathers information about a call from the moment the call enters the system until the call is disconnected. The parameters are collected from the caller's input in the IVR, read from the organization's database, or set in the process of handling the call. All information is stored in the call profile.

ShoreWare Agent Toolbar makes all the call profile information available to the external application.

The call profile fields are either fixed (e.g. priority, required service, and time that the call entered the system) or user-defined in the Contact Center Director. Either type of field can be used to pass parameters that the Contact Center can collect and share with the CRM application.

Configuring ShoreWare Agent Toolbar

During the DDE integration, the configuration of the ActiveX behavior is done on the ETAS.INI file.

No configuration is required on ShoreWare Agent Toolbar. All events are sent, all call profile information is sent on relevant events, and all commands are enabled. It is the responsibility of the CRM application to ignore what is not required.

Managing ShoreWare Agent Toolbar

Management includes starting ShoreWare Agent Toolbar, and also stopping and monitoring the connection state to the Contact Center Server.

The application can start ShoreWare Agent Toolbar using the following method:

```
StartEa(AgentNumber AgentPwd)
```

Where:

AgentNumber = the agent's number
AgentPwd = the agent's password

The application waits for a response event. Two possible events are:

Event	Description
Started()	ShoreWare Agent Toolbar started and logged in.
Denied(Cause number)	ShoreTel Contact Center Server denied the login. See "Possible OnDenied Causes" on page 28.

The application can stop ShoreWare Agent Toolbar using the following method:

```
StopEa()
```

No parameters are required.

The Stopped() is the event indicating a successful action.

Monitoring the connection state to the Contact Center Server uses the Connection lost and Restarted events. When Connection is lost (until a Restarted event is received), the ShoreWare Agent DDE is Out of Service. No new calls are received, and all telephony and ACD methods are inoperative while ShoreWare Agent Toolbar is Out of Service.

After starting ShoreWare Agent Toolbar, the ActiveX component must be initialized by calling the init() method.

The init method has the following signature:

```
HRESULT Init(BSTR name)
```

Where the default name is "CRM" and must match the Server definition on the DDE part of the ETAS.INI file.

Note: After calling the `StopEa()` method, the ActiveX component will stop to respond. In order to re-activate it is necessary to call the `init()` method again.

ShoreWare Agent Toolbar Events

ShoreWare Agent Toolbar sends events to the external application when the events shown below occur.

The ACD Events

The following table lists the ACD events:

Command	Parameters	Description
OnLogin	GroupName	Agent logged in.
OnLogout	GroupName	Agent logged off.
OnRelease		Agent was released.
OnResume		Agent was resumed.
OnWrapUp		Agent goes to wrap-up.
OnReady		Agent returns from wrap-up.

The following rules apply:

- More than one active call can exist at the same time. Therefore the values for the call ID are important.

The TEL Events

The telephony events:

Command	Parameters	Description
OnIncoming	CallId, Calltyp, DNIS, ANI, Service, Priority, ACD TimeIn	A call is ringing at the agent phone.
OnConnected	CallId, Calltyp, DNIS, ANI, Service, Priority, ACD TimeIn	The agent answers the call.
OnCleared	CallId	The call terminates.
OnHeld	CallId	A call was held.
OnRetrieved	CallId	A call was retrieved.

ShoreWare Agent Toolbar Methods (Services)

ShoreWare Agent Toolbar supports the following commands:

- ACD for ACD-related functions
- TEL for all telephony-related functions

The ACD Methods

The ACD methods:

Command	Parameters	Description
LoginPrimaryGroups	None	Logs in to all primary groups.
LogoutPrimaryGroups	None	Logs off from all primary groups.
LoginGroup	Group name	Logs in to a specific group.
LogoutFromGroup	Group name	Logs off from a specific group.
ReleaseAgent	None	Makes agent unavailable.
ResumeAgent	None	Makes agent available.
WrapUpCode	WrapUp code	Enters wrap-up code.
ControlledWrapUp	None	Lets the agent continue in wrap-up mode until the agent is finished (otherwise a predetermined period is set by the administrator).
AgentReady	None	Terminates the wrap-up time.
TransferToAgent	Agent number	Transfers call to specified agent.
Help	None	Help request from supervisor.

The TEL Methods

The telephony methods:

Command	Parameters	Description
Answer	None	Answers the ringing call.
HangUp	None	Clears the current call.
MakeCall	Destination dial number	Makes an outgoing call.
ClearConnection	Call ID	Terminates the call with the selected Call ID.
Hold	None	Puts the current call on hold.
Retrieve	None	Retrieves a call that was previously put on hold.
Divert	Destination dial number	Diverts the call to the desired destination before answering the call.
StartTransfer	Destination dial number	Starts the transfer process.
CompleteTransfer	None	Removes the agent from the call, completing the transfer.
SingleStepTransfer	Destination dial number	Transfers the current call to another telephone.
StartConference	Destination dial number	Begins a conference call.
CompleteConference	None	Complete a conference call.
Alternate	None	Exchanges between the call on hold and the active call.
Reconnect	None	Returns the agent to the original call and terminates the consultation.

Properties (Get Only)

The properties. All strings are not case-sensitive:

Item name	Parameters	Returns (string)	Description
ISPHONEBUSY	None	TRUE or FALSE (0 or 1 Long)	TRUE if the phone is busy.
HeldCallsCount	None	Number of calls on hold (Long)	Number of calls currently on hold.
User or skill CallProfile field	Call Profile field name	Field value (String)	Retrieve the value of a specific call profile field.

Item name	Parameters	Returns (string)	Description
IsRelease	None	TRUE or FALSE (0 or 1 Long)	TRUE if the agent is in the release state.
IsLoggedIn	Group name	TRUE or FALSE (0 or 1 Long)	
IsWrapUp	None	TRUE or FALSE (0 or 1 Long)	TRUE if the agent is currently in wrap-up state.

Possible OnDenied Causes

The possible OnDenied causes:

Cause	Name	Description
1	LOGGED_ON_TO_ANOTHER_EXT	Agent is already logged in to another extension.
2	NEW_AGENT_ON_EXT	Another agent is logged in to current extension.
3	WRONG_AGENT	Wrong agent ID is supplied.
4	MAX_NUM	Maximum number of agents in system is exceeded.
5	FAILED_TO_READ_FROM_DBS	Internal error.
6	STILL_INITIALIZING	ShoreWare Agent Toolbar cannot start because the system is still initializing.

Variable Types

All strings are VB strings/BSCR in C language and are not case-sensitive:

Name	Type	Comment
AgentNumber	String	
AgentPwd	String	
Cause number	Long	
CallId	Long	
Calltyp	Long	
DNIS	String	
ANI	String	
Service	String	
Priority	Long	
TimeIn	String (HH:MM:SS)	
GroupName	String	
Destination Dial number	String	
WrapUp code	String	
Any User field	String	Fields can be defined in administration.
Any Skill field	String	Fields can be defined in administration.

Activating Applications by Triggers

The interface, called triggers, is provided by ShoreWare Agent Toolbar to activate external applications available at the agent environment.

Introduction

ShoreWare Agent Toolbar triggers interface enables the activation of external applications at events (mainly states) of ShoreWare Agent Toolbar.

When receiving an event (a state change or activity performed by the agent), the command line related to that event on the INI file is activated.

Configuring ShoreWare Agent Toolbar

The configuration of ShoreWare Agent Toolbar to activate triggers is done on the triggers area of the ETAS.INI file for the appropriate agent. Initially, the triggers area appears as follows:

```
[TRIGGERS]

OperateTriggers=1
```

To activate the triggers option, OperateTriggers should remain set to 1.

Activating an Application

You can activate an application at a specific event by configuring the command for the event. For example, you might activate an application called “MyApplication.exe” at a call ringing on the agent station and then pass the application several parameters related to the call, such as an ANI number, DNIS, and the called extension. To activate the event, set the following commands:

```
[TRIG_OnIncoming]

Command=MyApplication.exe %calling %DNIS %Service %called
```

Activation Events

ShoreWare Agent Toolbar can activate a command when the following events occur:

```
TRIG_OnStarted
TRIG_OnReStarted
TRIG_OnDenied
TRIG_OnStopped
TRIG_OnLoggedIn
TRIG_OnLoggedOut
TRIG_OnHeld
TRIG_OnRetrieved
TRIG_OnIncoming
TRIG_OnConnected
TRIG_OnConferenced
TRIG_OnReleased
TRIG_OnResumed
TRIG_OnWrapUp
TRIG_OnReady
TRIG_OnCleared
```

Command Substitution

The command line can include free text and call profile parameters (mandatory and user-defined fields). For more information, see [Chapter 1, CRM Integration with Call Control Scripts](#).

Integration Examples

Overview

The following examples show possible integrations of ShoreTel Contact Center Solution with external applications.

Web Application by Triggers

The following is an example of activating a web application using the triggers interface. The application is started in a new Windows Internet Explorer window and shows all activities related to the calling customer:

```
[TRIGGERS]
OperateTriggers=1

[TRIG_OnIncoming]
command=start c:\\Program files\\internet explorer\\iexplore.exe
http://www.company.com/HomePages/CmaInternalPage/
1,1564,11,FF.html?EZRu n=1&PersonalID=%calling
```

Vantive by DDE

The following example is for displaying a file for the calling customer on a Vantive application:

```
[DDE]
Server=Vantive
Topic=System

[DDE_OnIncoming]
Command=maEaXIncoming X%calling
```

MSAccess by DDE

The following example starts the MSAccess application activating a database named MYDB at ShoreWare Agent Toolbar activation and then on each arriving call opens the callers file:

```
[DDE]
ActivateServer=True
Server=MsAccess
Topic=System
Database=D:\MYDB.mdb
[DDE_OnIncoming]
Command=[OpenForm cust,,, phone = %calling]
```

```
[DDE_OnStarted]
Command=[OpenDatabase D:\MYDB.mdb]
```

NOTE: Remember you must have MS Access open with the database to get the screen popup.

External Application by ActiveX

The following example is the Visual Basic 6 implementation of a console that enables the user to activate the controls of the ShoreWare Agent Toolbar software. Events are received in the List window of the console, as shown:

Code of VbController.exe

```
Private Sub Button_Alternate_Click()
EaX.Alternate
End Sub

Private Sub Button_Answer_Click()
EaX.Answer
End Sub

Private Sub cClearConn_Click()
End Sub

Private Sub Button_CompTrans_Click()
EaX.CompleteTransfer
End Sub

Private Sub Button_Divert_Click()
EaX.Divert Edit_Divert.Text
End Sub

Private Sub Button_Exit_Click()
End
End Sub

Private Sub Button_GetCp_Click()
Edit_CallProfileValue.Text =
EaX.CallProfile(Edit_CallProfileName.Text)
End Sub

Private Sub Button_HangUp_Click()
EaX.HangUp
End Sub

Private Sub Button_Hold_Click()
EaX.Hold
End Sub

Private Sub Button_IsLoggedIn_Click()
If EaX.IsLoggedIn(Edit_IsLogInGroupName.Text) = 1 Then
Shape_IsLoggedIn.BackColor = &H80000003
Else
```

```
Shape_IsLoggedIn.BackColor = &HFF&
End If
End Sub

Private Sub Button_LogIn_Click()
EaX.LoginGroup Edit_LogInGroupName.Text
End Sub

Private Sub Button_LogInPrim_Click()
EaX.LoginPrimaryGroups
End Sub

Private Sub Button_LogOut_Click()
EaX.LogoutFromGroup eLoGroup.Text
End Sub

Private Sub Button_LogoutPrim_Click()
EaX.LogOutFromPrimaryGroups
End Sub

Private Sub Button_MakeCall_Click()
EaX.MakeCall Edit_MakeCallDestination.Text
End Sub

Private Sub Button_LogOff_Click()
EaX.StopEa
End Sub

Private Sub Button_LogOn_Click()
EaX.StartEa Edit_AgentId.Text, Edit_Password.Text
End Sub

Private Sub Button_CompConf_Click()
EaX.CompleteConference
End Sub

Private Sub Button_Ready_Click()
EaX.AgentReady
End Sub

Private Sub Button_Reconnect_Click()
EaX.Reconnect
End Sub

Private Sub Button_Release_Click()
EaX.ReleaseAgent
End Sub

Private Sub Button_Resume_Click()
EaX.ResumeAgent
End Sub

Private Sub Button_Retrieve_Click()
EaX.Retrieve
End Sub
```

```
Private Sub AddToLog(ByVal line As String)
If List.ListCount > 20 Then
List.Clear
End If
List.AddItem line
End Sub

Private Sub Button_SingleStepTransfer_Click()
EaX.SingleStepTransfer Edit_SingleStepTransfer.Text
End Sub

Private Sub Button_StartConf_Click()
EaX.StartConferrence Edit_Conferrence.Text
End Sub

Private Sub Button_StartTrans_Click()
EaX.StartTransfer Edit_Transfer.Text
End Sub

Private Sub Button_TransferToAgent_Click()
EaX.TransferToAgent Edit_TransferTyAgendId.Text
End Sub

Private Sub Button_Wrapup_Click()
EaX.WpapUp Edit_WrapupCode.Text
End Sub

Private Sub EaX_OnCleared(ByVal callid As Long)
AddToLog "Cleared: " + Str(callid)
End Sub

Private Sub EaX_OnConnected(ByVal callid As Long)
AddToLog "Connected: " + Str(callid)
End Sub

Private Sub EaX_OnHeld(ByVal callid As Long)
AddToLog "Held:" + Str(callid)
End Sub

Private Sub EaX_OnIncoming(ByVal callid As Long)
AddToLog "Incoming: " + Str(callid)
End Sub

Private Sub EaX_OnLoggedIn(ByVal group As String)
AddToLog "LoggedIn " + group
End Sub

Private Sub EaX_OnLoggedOut(ByVal group As String)
AddToLog "LoggedOut " + group
End Sub

Private Sub EaX_OnReady()
AddToLog "Ready "
End Sub
```

```

Private Sub EaX_OnReleased()
AddToLog "Released"
End Sub

Private Sub EaX_OnResumed()
AddToLog "Resumed"
End Sub

Private Sub EaX_OnRetrieved(ByVal callid As Long)
AddToLog "Retrieved " + Str(callid)
End Sub

Private Sub EaX_OnWrapUp()
AddToLog "WrapUp"
End Sub

Private Sub EaX_Started()
AddToLog "Started"
End Sub

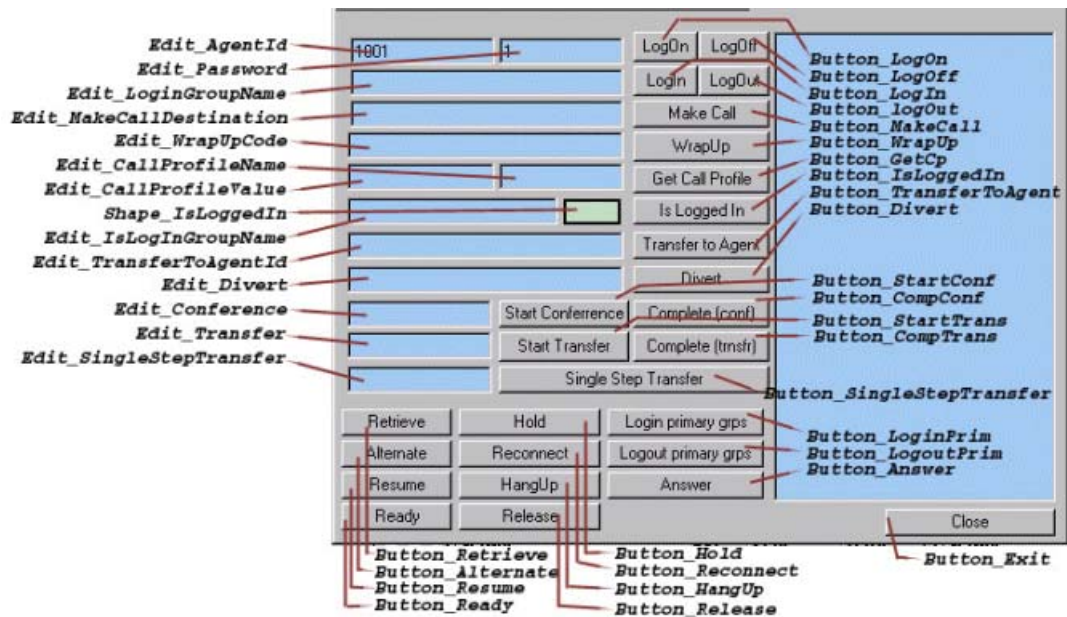
Private Sub EaX_Stopped()
AddToLog "Stopped"
End Sub

Private Sub Form_Load()
EaX.Init "Temp"
End Sub

```

The Control Panel

The Control panel:



ActiveX API Quick Reference

Overview

The ShoreTel Contact Center Solution contains a set of APIs that allows you build custom Active X interfaces.

Functions

The following functions are used with the ShoreTel Contact Center Solution.

```

HRESULT AgentReady();
HRESULT Alternate();
HRESULT Answer();
HRESULT CallProfile(BSTR cname, [in] BSTR newVal); // Set a CP
HRESULT CallProfile(BSTR cname, [out, retval] BSTR *pVal); // Get
a CP
HRESULT CallProfileByCID(long cid, BSTR cname, [out, retval] BSTR
*pVal);
HRESULT CloseAgentBoard();
HRESULT CompleteConference();
HRESULT CompleteTransfer();
HRESULT Divert(BSTR dest);
HRESULT ExecProc(BSTR path,BSTR directory);
HRESULT ExtendWrapUp();
HRESULT HangUp();
HRESULT HeldCallsCount([out, retval] long *pVal);
HRESULT Help();
HRESULT Hold();
HRESULT Init(BSTR name);
HRESULT IsInEmailState([out, retval] BOOL *pVal);
HRESULT IsLoggedIn(BSTR group, [out, retval] BOOL *pVal);
HRESULT IsPhoneBusy([out, retval] BOOL *pVal);
HRESULT IsReleased([out, retval] BOOL *pVal);
HRESULT IsSystemAvailable([out, retval] BOOL *pVal);
HRESULT LoginGroup(BSTR group);
HRESULT LoginPrimaryGroups();
HRESULT LogoutFromGroup(BSTR group);
HRESULT LogOutFromPrimaryGroups();
HRESULT MakeCall(BSTR dest);
HRESULT OpenAgentBoard();
HRESULT OpenCallsLog();
HRESULT OpenCallsStatus();
HRESULT OpenLoginGroupManager();
HRESULT Reconnect();
HRESULT ReinsertCBBusy();

```

```

HRESULT ReinsertCBNoAnswer();
HRESULT ReinsertCBTerminate();
HRESULT ReleaseAgent();
HRESULT ReleaseAgentWithCode(BSTR code);
HRESULT ResumeAgent();
HRESULT Retrieve();
HRESULT RouteMoreEmails();
HRESULT SingleStepTransfer(BSTR dest);
HRESULT StartConferrence(BSTR dest);
HRESULT StartEa(BSTR aid,BSTR pwd);
HRESULT StartEaEx(BSTR aid,BSTR pwd,BSTR ext,BSTR email,BSTR
email_pwd);
HRESULT StartTransfer(BSTR dest);
HRESULT StopEa();
HRESULT ToggleEmailState();
HRESULT TransferToAgent(BSTR aid);
HRESULT Version([out, retval] BSTR *pVal);
HRESULT WpapUp(BSTR code);
HRESULT WrapUp(BSTR code);

```

Events

A list of the ActiveX Control Events is listed below.

```

HRESULT Denied(long cause);
HRESULT OnCleared(long callid);
HRESULT OnConferenced(long callid);
HRESULT OnConnected(long callid, long Calltyp, BSTR DNIS, BSTR
ANI, BSTR Service, long Priority, BSTR ACD_Time_In);
HRESULT OnFailed(long cause);
HRESULT OnHeld(long callid);
HRESULT OnIncoming(long callid,long Calltyp, BSTR DNIS, BSTR ANI,
BSTR Service, long Priority, BSTR ACD_Time_In);
HRESULT OnLoggedIn(BSTR group);
HRESULT OnLoggedOut(BSTR group);
HRESULT OnReady();
HRESULT OnReleased();
HRESULT OnReserved(long callid);
HRESULT OnResumed();
HRESULT OnRetrieved(long callid);
HRESULT OnSystemAvailable(bool sys_on);
HRESULT OnWrapUp();
HRESULT Restarted();
HRESULT Started();
HRESULT Stopped();

```

Supported SQL Syntax

Overview

Listed in this section is the SQL syntax statements supported by the ShoreTel Contact Center Solution.

All simple statements

The following is an example of a simple SQL statement.

```
Select field1, field2, field3 from table1 where field1 = value
Update table1 set field2=value where where field1 = value
Insert into table1 values ( field1, field2, field3 )
Delete from table1 where field1 = value
```

Statements with computed fields

The following is an example of a SQL statement with comuted fields.

```
Select field1, fieldx*2 as field2, field3 from table1 where field1
= value
```

Joint statements

The following is an example of joint SQL statements.

```
Select field1, field2 from table1, table2 where table1.id1 =
table2.id2 and field1 > 0
Select field1, field2 from table1 left join table2 on table1.id1 =
table2.id2 where field1 > 0
Select field1, field2 from table1 inner join table2 on table1.id1 =
table2.id2 where field1 > 0
```

Nested Statements

The following is an example of nested SQL statements.

```
Select field1 from table1 where id1 in ( select id2 from table2 )
Update table1 set field1 = value1 where id1 in ( select id2 from
table2 )
Delete from table1 where id1 in ( select id2 from table2 )
```

Scripts

The following is an example of a SQL statement with scripts.

```
Declare @value1 int
Declare @value2 int
Set @value1 = 5
Select @value2 = field2 from table1 where field1 = @value1
```

SQL Syntax Format Limitations

The following SQL syntax format restrictions apply to the ShoreTel Contact Center Solution.

- In a field that accepts only select statements, a statement not starting with word "select" will not work.
- The outcome of complex scripts that involves more than one (1) basic statement are usually dependent on the ODBC driver of the database.
- In order for a database integration to work successfully, music on hold must be configured. During the execution of a SQL execute, the system will play the default hold music (set in System->Misc) until a result is received from the database. If the music on hold is not set, the system will hang up on the caller.

ShoreTel recommends that ShoreTel professional services or implementation services be engaged in database or dial list scripting to ensure that the your requirements are met.

Index

A	
ACD	17, 20, 23
ACD calls	9
ACD events	25
ACD methods	25
activate an application	29
activation events	29
ActiveX	15, 23
Agent DDE interface	17
agent requirements	9
agent states	16
Agent Toolbar	13, 15, 16, 17, 18, 23, 28
ANI	15
Announce action	11
announcements	9

B	
BSCR in C	28

C	
Call Control Script Director	14
call control scripts	7, 9
Call Control Scripts Director	9
call flow	9
call profile	7, 9, 15, 19, 23, 29
call results	13
call routing	7, 9
caller information	9
client side processes	14
client-based integrations	15
command substitution	29
Contact Center Server	14
CRM applications	17
CRM database	7, 9
customer database	11
Customer Relationship Management (CRM)	7

D	
Data Source Name (DSN)	10, 14
Data-entry fields	
conventions	6
DDE	15
DDE commands	19
DDE interface	17
DDE server	20
DDE_EXECUTE	20, 21, 22
DDE_REQUEST	21, 22

dial lists	13
DNIS	15

E	
Enterprise Contact Center	7
events	25
EXECUTE command	18
external applications	15, 28, 31
external database	14

G	
Graphical Call Control Scripts Director	9

H	
Hypertext links	
conventions	6

I	
infrastructure software	14
INI file	15, 28
Interactive Voice Response (IVR)	9
IVR port	9

M	
Microsoft triggers	15, 28
music-on-hold	9

O	
ODBC drivers	14
ODBC interface	10
ODBC/SQL database	14
OnDenied causes	20, 28
outbound calls	13, 14

P	
pending dial numbers	13
preview dialing	13
properties	26

R	
response event	24

S

screen pops, implementing17
self-serve applications11
self-service application7
self-service applications9
SQLConnect 10, 11, 14
SQLExecute 10, 14
SQLQuery9
Structured Query Language (SQL)10

T

Target Service Factor (TSF)13
TEL21
TEL events25
TEL methods26
TEL topic18
telephony 9, 17, 23
telephony events25
telephony methods26
trigger events29

V

VB strings28

W

Windows application18
Windows INI file15