© 2001, Avaya Inc.
All Rights Reserved

Notice
Every effort was made to ensure that the information in this book was complete and accurate at the time of printing. However, information is subject to change.

Preventing Toll Fraud
“Toll fraud” is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or working on your company’s behalf). Be aware that there may be a risk of toll fraud associated with your system and that, if toll fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Fraud Intervention
If you suspect that you are being victimized by toll fraud and you need technical support or assistance, call Technical Service Center Toll Fraud Intervention Hotline at +1 800 643 2353.

Providing Telecommunications Security
Telecommunications security (of voice, data, and/or video communications) is the prevention of any type of intrusion to (that is, either unauthorized or malicious access to or use of your company’s telecommunications equipment) by some party.

Your company’s “telecommunications equipment” includes both this Avaya product and any other voice/data/video equipment that could be accessed via this Avaya product (that is, “networked equipment”).

An “outside party” is anyone who is not a corporate employee, agent, subcontractor, or working on your company’s behalf. Whereas, a “malicious party” is anyone (including someone who may be otherwise authorized) who accesses your telecommunications equipment with either malicious or mischievous intent.

Such intrusions may be either throughput synchronous (time-multiplexed and/or circuit-based) or asynchronous (character-, message-, or packet-based) equipment or interfaces for reasons of:
- Utilization (of capabilities special to the accessed equipment)
- Theft (such as, of intellectual property, financial assets, or toll-facility access)
- Eavesdropping (privacy invasions to humans)
- Mischief (troubling, but apparently innocuous, tampering)
- Harm (such as harmful tampering, data loss or alteration, regardless of motive or intent)

Be aware that there may be a risk of unauthorized intrusions associated with your system and/or its networked equipment. Also realize that, if such an intrusion should occur, it could result in a variety of losses to your company (including but not limited to, human/data privacy, intellectual property, material assets, financial resources, labor costs, and/or legal costs).

Your Responsibility for Your Company’s Telecommunications Security
The final responsibility for securing both this system and its networked equipment rests with you - an Avaya customer’s system administrator, your telecommunications peers, and your managers. Base the fulfillment of your responsibility on acquired knowledge and resources from a variety of sources including but not limited to:
- Installation documents
- System administration documents
- Security documents
- Hardware/software-based security tools
- Shared information between you and your peers
- Telecommunications security experts

To prevent intrusions to your telecommunications equipment, you and your peers should carefully program and configure your:
- Avaya-provided telecommunications systems and their interfaces
- Avaya-provided software applications, as well as their underlying hardware/software platforms and interfaces
- Any other equipment networked to your Avaya products.

Federal Communications Commission Statement
Part 15: Class A Statement. This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Part 15: Class B Statement. This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:
- Reorient the receiving television or radio antenna where this may be done safely.
- To the extent possible, relocate the receiver with respect to the telephone equipment.
- Where the telephone equipment requires ac power, plug the telephone into a different ac outlet so that the telephone equipment and receiver are on different branch circuits.

Part 15: Personal Computer Statement. This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC Rules. Only peripherals (computing input/output devices, terminals, printers, etc.) certified to comply with the Class B limits may be attached to this computer. Operation with noncertified peripherals is likely to result in interference to radio and television reception.

Part 68: Network Registration Number. This equipment is registered with the FCC in accordance with Part 68 of the FCC Rules. Each registered circuit card is identified by an FCC registration number located on the card.

Part 68: Answer-Supervision Signaling. Allowing this equipment to be operated in a manner that does not provide proper answer-supervision signaling is in violation of Part 68 rules. This equipment returns answer-supervision signals to the public switched network when:
- Answered by the called station
- Answered by the attendant
- Routed to a recorded announcement that can be administered by the CPE user

This equipment returns answer-supervision signals on all DID calls forwarded back to the public switched telephone network. Permissible exceptions are:
- A call is unanswered
- A busy tone is received
- A reorder tone is received

Canadian Department of Communications (DOC) Interference Information
This digital apparatus does not exceed the Class A limits for radio noise emissions set out in the radio interference regulations of the Canadian Department of Communications.

Le Présent Appareil Numérique n’émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la class A précrites dans le règlement sur le brouillage radioélectrique édicté par le ministère des Communications du Canada.

European Union Declaration of Conformity
Avaya Business Communications Systems declares that equipment specified in this document conforms to the referenced European Union (EU) Directives and Harmonized Standards listed below:

EMC Directive 89/336/EEC
Low Voltage Directive 73/23/EEC
The “CE” mark affixed to the equipment means that it conforms to the above Directives.

Trademarks
- Avaya Inc. -- Registered trademarks: CONVERSANT, DEFINITY, CentreVu. Trademark: Avaya.
- Siebel Systems, Inc. -- Registered trademark: Siebel.

Ordering Information
Call: Avaya Publications Center
Voice +1 800 457 1235
Fax +1 800 457 1764
International Voice 317 322 6416
International Fax 317 322 6699

Write: Globalware Solutions
200 Ward Hill Avenue
Haverhill, MA 01835 USA

Order: Document No. 585-352-200, Issue 1, September 2001
You can be placed on a Standing Order list for this and other documents you may need. Standing Order will enable you to automatically receive updated versions of individual documents or document sets, billed to account information that you provide. For more information on Standing Orders, or to be put on a list to receive future issues of this document, please contact the Avaya Publications Center.

Warranty
Avaya Inc. provides a limited warranty on this product. Refer to the “Limited use Software License Agreement” card provided with your package.

Avaya National Customer Care Center
Avaya provides a telephone number for you to use to report problems or to ask questions about your contact center. The support telephone number is 1-800-242-2121.

Avaya Web Page
http://www.avaya.com

Comments
To comment on this document, return the comment card at the end of the document.

Acknowledgment
This document was written by the CRM Development group of Avaya University.
CONVERSANT System Version 8.0 Computer Telephony Integration

Chapter 1: Overview
  Introduction ................................................................. 3
  Description of CTI ......................................................... 3
  Principles of operation ............................................... 4
  Functionality .............................................................. 4
  Call Center features .................................................... 5

Chapter 2: Setup and administration
  CTI DIP setup ............................................................. 7
    Installing the CentreVu CT Client ................................ 7
    Installing the CTI DIP ................................................ 7
    Set Up the Configuration File ...................................... 8
    Verify external function files .................................... 11
  Telephony setup .......................................................... 12
    Analog tip/ring connections ........................................ 12
    Line side digital connections ...................................... 12
  CONVERSANT and PBX administration ............................... 12
    Ethernet administration ............................................. 13
    Administering the tip/ring, LSE1, and LST1 lines ............. 14
  CVCT administration ................................................... 14
    CVCT administration of CONVERSANT ................................ 14
    Administration of the CVCT servers ............................... 15
    Setup and administration of a Siebel Client ................... 15

Chapter 3: Application development
  External functions ....................................................... 17
    ctiCallInfo .................................................................. 18
    ctiCallState ............................................................ 20
    ctiConfer .................................................................. 22
    ctiDial .................................................................... 23
    ctiDiscon .................................................................. 24
    ctiHold .................................................................... 25
    ctiNotify .................................................................. 26
    ctiPrivData ............................................................. 27
    ctiRetrieve .............................................................. 29
    ctiTransfer ............................................................... 30
    Sample application .................................................... 31

Glossary ........................................................................... 39
Chapter 1: Overview

Introduction

This book provides information about the setup, administration, and use of the Computer Telephony Integration (CTI) feature for the CONVERSANT System Version 8.0. It is intended for application developers who are familiar with:

- Script Builder or Voice@Work
- CentreVu Computer Telephony
- ASAI
- DEFINITY PBX operation

Description of CTI

Applications that make function calls to the CTI data integration process (CTI DIP) on CONVERSANT can use a CentreVu Computer Telephony (CVCT) server to control ports on a PBX and to interact with Siebel eBusiness applications that are located on a client connected to the server. CTI is primarily an alternative to the manipulation of a PBX via a direct ASAI connection to the CONVERSANT (for a description of ASAI, see Chapter 3, “Adjunct/Switch Application Interface”, in CONVERSANT System Version 8.0 Communication Development, 585-313-220).

One of the main benefits of CTI is redundancy. Up to two extra CVCT servers can be connected to a CONVERSANT as backups to the server handling calls. If connection to the primary CVCT server is lost, then connection is made to a second server; if this connection is lost, then the third server is used.

Full CTI support is for:

- DEFINITY 6.3 or later
- CentreVu CT Release 9.1 Version 1 or later
- Private Data Version 6
- TSAPI Version 2
Overview

Principles of operation

CTI operation is shown schematically in CTI connectivity below.

An application on CONVERSANT uses the CTI DIP (administered via a configuration file) to interact with CVCT Client for UnixWare. The CVCT Client sends and receives TSAPI messages over a LAN to a network interface card (NIC) on the CVCT server. This same NIC can be used to communicate with a Siebel eBusiness application residing on a client that uses CVCT Client software and a special driver (CentreVu CT for Siebel Client).

The CVCT server utilizes a telephony server (TSRV) to route messages to the PBX with the aid of a special driver and another NIC. Messages at the Definity PBX are handled by its MAPD circuit card.

CTI connectivity

Functionality

Function calls to the CTI DIP are designed to do the following on the active ports of a CONVERSANT:

- Put calls on hold
- Retrieve calls from hold
- Disconnect calls
- Transfer two calls
- Conference two calls
- Report information to the application about a call (port extension, call ID, ANI, Called Number)
- Report the state of each call to the application
- Provide answer notification to a voice script
Call Center features

Like ASAI, the CTI provides capabilities for use in DEFINITY call center environments.

- **Universal Call ID (UCID)** — UCID provides a unique identifier (8-byte binary or 20-character ASCII) for every call in a DEFINITY call center customer environment. UCID allows for uniform data-tracking for all call-related data in a call center, regardless of the system. DEFINITY uses the ASAI interface to pass the UCID to the CVCT Server.

- **User-to-User Information element (UUI)** — UUI allows the customer to specify additional information to be passed in external function arguments, which can contain up to 96 bytes of information (compare with the 32 bytes available using ASAI).
Chapter 2: Setup and administration

Before CTI can be used, the following must be done:

1. CTI DIP setup
2. Telephony setup
3. CONVERSANT and PBX administration
4. CVCT administration
5. Setup and administration of a Siebel Client

CTI DIP setup

Setup of the CTI DIP has the following steps:

1. Installing the CentreVu CT Client
2. Installing the CTI DIP
3. Set Up the Configuration File
4. Verify external function files

Installing the CentreVu CT Client

To learn how to install the CentreVu CT Client for UnixWare, see Chapter 4, "Installing CVCT TSAPI Client Software," in CentreVu Computer Telephony Release 9.1, Version 1, Telephony Services and CallVisor PC Installation.

Installing the CTI DIP

The CTI DIP is installed as the optional feature package ctidip. If the package has already been installed, there will be a /vs/data/cti directory. If the directory does not exist, install the package as follows:

1. Insert the diskette into the diskette drive.
2. At the UNIX prompt, type pkgadd -d diskette1 and press ENTER.
   The screen displays the default, which is to install everything.
3. Press ENTER to accept the default.
   The installation continues.
4. Remove the diskette from the diskette drive.
For more information about installing packages, see Chapter 7 of 
CONVERSANT System Version 8.0 UCS 1000 Maintenance, 585-313-150.

**Set Up the Configuration File**

The configuration file, `dialer_cfg.dat`, is in the `/vs/data/cti` directory. It consists of eight required fields which must be in the order listed in the following table. Field values should be on separate lines that do not include comments.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug Flag</td>
<td>This Boolean flag indicates that the CTI DIP is to provide message tracing in addition to error logging. If this setting is turned off, only error logging will be provided. It is recommended that this field be set to off while the CTI DIP is handling a high volume of traffic. 0 denotes the OFF setting, while 1 denotes the ON setting. The default is 0, OFF.</td>
</tr>
<tr>
<td>System Status</td>
<td>The value in this field determines how frequently (in seconds) the CTI DIP sends a heartbeat to the switch. This is vital to check if the CVCT server is still in service. Note, however, that system performance may be affected if this value is set to less than 10 seconds. The default value is 10 seconds.</td>
</tr>
<tr>
<td>Timer Interval</td>
<td></td>
</tr>
<tr>
<td>Error Log File</td>
<td>This field contains the path to the log file where the CTI DIP will record error and debugging messages. The default value for the error log file is <code>/vs/data/cti/cdial.log</code>.</td>
</tr>
</tbody>
</table>
## CTI DIP configuration file fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Name</td>
<td>This field contains the CVCT login id for the CONVERSANT user application. The CTI DIP uses this login, along with the password to access to the CVCT server. The initial value for the login id is <code>&lt;login name&gt;</code>.</td>
</tr>
<tr>
<td>Password</td>
<td>This field contains the CVCT password for the CONVERSANT user application. The CTI DIP uses this login and password to access to the CVCT server. The initial value for the password is <code>&lt;password&gt;</code>.</td>
</tr>
<tr>
<td>Advertised Telephony Service for CVCT server #1</td>
<td>This field contains a string that identifies Telephony server #1 and the service on that telephony server that the CTI DIP is instructed to use. The Telephony server name or IP address must be in the <code>/usr/lib/tslibrc</code> file. The initial value for this field is <code>&lt;Tserver1&gt;</code>.</td>
</tr>
</tbody>
</table>
Setup and administration

CTI DIP configuration file fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advertised Telephony Service for CVCT server #2</td>
<td>This field contains a string that identifies Telephony server #2 and the service on that telephony server that the CTI DIP is instructed to use. The Telephony server name or IP address must be in the <code>/usr/lib/tslibrc</code> file. The initial value for this field is <code>&lt;Tserver2&gt;</code>.</td>
</tr>
<tr>
<td>Advertised Telephony Service for CVCT server #3</td>
<td>This field contains a string that identifies Telephony Server #3 and the service on that telephony server that the CTI DIP is instructed to use. The Telephony server name or IP address must be in the <code>/usr/lib/tslibrc</code> file. The initial value for this field is <code>&lt;Tserver3&gt;</code>.</td>
</tr>
</tbody>
</table>

⚠️ CAUTION:                                                                                                                                                                                                 |
There can be no empty lines in the file (note that the “#” symbol may be used to turn unused lines into comments).
The following figure shows a sample configuration file.

**Sample CTI DIP configuration file**

```
# Dialer Client Configuration File
# must be located in "/vs/data/cti/dialer_cfg.dat"
#
# System Status Timer interval (seconds)
# 10
#
# Error Log File
# /vs/data/cti/cdial.log
#
# CVCT server user login name for application
# joeblow
#
# CVCT server user password for application
# 125p96
#
# Advertised telephony service for Tserver #1
# LUCENT#G3_SWITCH#CSTA#IVRINT15
#
# Advertised telephony service for Tserver #2
# LUCENT#G3_SWITCH#CSTA#WILBUR
#
# Advertised telephony service for Tserver #3
# LUCENT#G3_SWITCH#CSTA#ROSINANTE
```

**Verify external function files**

Verify that the following external function files are in the `/vs/bin/ag/lib` directory:

- ctiCallInfo.t
- ctiCallState.t
- ctiConfer.t
- ctiDial.t
- ctiDiscon.t
- ctiHold.t
Setup and administration

- ctiNotify.t
- ctiPrivData.t
- ctiTransfer.t
- ctiRetrieve.t

If any of the files are missing, then reinstall the CTI DIP (see Installing the CTI DIP on page 7).

Telephony setup

You must make digital and/or analog telephony connections to the PBX. They are the same as for ASAI.

**Analog tip/ring connections**

You must install analog tip/ring circuit cards in the CONVERSANT system with each line connected separately. For information on tip/ring circuit card capabilities for ASAI, see Chapter 2, “Hardware,” of *CONVERSANT System Version 8.0 System Description*, 585-313-219.

**Note:**

Analog connections are not supported on the UCS 1000.

**Line side digital connections**

Digital connections between the CONVERSANT system and the line side of the switch are made with either line side FXS T1 or line side FXS E1.

This type of connection allows the use of various switch features that are not compatible with an ordinary T1 trunk connected between the CONVERSANT system and switch. These features include call transfer and call progress tone (CPT) detection, either in conjunction with Full CCA or where an E1/T1 interface circuit card is used for communications.

CONVERSANT and PBX administration

You must administer the CONVERSANT and the DEFINITY PBX as follows:

1. Administer the MAPD in DLG mode. See *CallVisor PC LAN over MAPD Installation, Administration, and Maintenance*, 555-230-113.

2. Install and administer the CONVERSANT Ethernet circuit card. See the "Installing or Replacing Circuit Cards" chapter in the maintenance book for your platform. Station administration is the same for either.
Note:
The UCS 1000 has dual, integrated LAN connections on the CPU Complex. Refer to "Installing Base System Software" in CONVERSANT System Version 8.0 UCS 1000 Maintenance, 585-313-150, for information on administer the LAN on the UCS 1000.

3. Administer the tip/ring, E1, or T1 telephone lines on the PBX.

Once you have completed these steps, assign telephone numbers (or extensions) to the ports of the CONVERSANT that will use CTI. See Chapter 3, “Voice System Administration,” of the CONVERSANT System Version 8.0 Administration, 585-313-510, for information on how to assign these services.

Ethernet administration

You must administer the DEFINITY for ASAI connectivity between the DEFINITY and the CVCT server.

Use the DEFINITY add station or change station commands. See the following table for the appropriate values.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Required or Optional?</th>
<th>Valid Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extension:</td>
<td>Required</td>
<td>Whatever fits your dial plan</td>
</tr>
<tr>
<td>Type:¹</td>
<td>Required</td>
<td>ADJLK</td>
</tr>
<tr>
<td>Port:</td>
<td>Required</td>
<td>The port that connects to the ASAI line</td>
</tr>
<tr>
<td>Name:</td>
<td>Optional</td>
<td>Can be used as an identifier</td>
</tr>
<tr>
<td>XID:¹</td>
<td>Required</td>
<td>y</td>
</tr>
<tr>
<td>Fixed TEI:¹</td>
<td>Required</td>
<td>y</td>
</tr>
<tr>
<td>TEI:¹</td>
<td>Required</td>
<td>3</td>
</tr>
<tr>
<td>MIM Support:¹</td>
<td>Required</td>
<td>n</td>
</tr>
<tr>
<td>CRV Length:²</td>
<td>Required</td>
<td>2</td>
</tr>
</tbody>
</table>

1. To match the built-in administration of the Ethernet circuit cards with the ASAI software, this field must have the value indicated.

2. In some previous releases, the CRV Length field required a value of 1. You must use the value 2 for CONVERSANT System Version 8.0.
Administering the tip/ring, LSE1, and LST1 lines

See Chapter 5, “Switch Interface Administration,” of CONVERSANT System Version 8.0 Administration, 585-313-510, to administer tip/ring, LSE1, and LST1 lines. To be certain that you select options that are compatible with the DEFINITY G3 switch (only certain versions), select DEFINITY in the PBX defaults screen.

Note:
The UCS 1000 does not support analog connections.

Note:
DEFINITY is the default setting. Consequently, if you are administering a new system, the lines are configured correctly by default.

Place all the lines into service. To do so, see the information on changing maintenance state in Chapter 3, “Voice System Administration,” of CONVERSANT System Version 8.0 Administration, 585-313-510.

⚠️ CAUTION:
Do not proceed until the lines are in the inserv state.

CVCT administration

You must administer the CONVERSANT and the CVCT servers to make the CONVERSANT a CVCT client.

CVCT administration of CONVERSANT

On the CONVERSANT, you must modify the tslibrc file located under /usr/lib/ to include the IP addresses of the CVCT servers.

See the sample file in Sample tslibrc file on page 15.

Note:
Unlike the dialer_cfg.dat file, the tslibrc file may contain blank lines.
Administration of the CVCT servers

Do the following steps on each CVCT server:

1. Create a new user name for the CONVERSANT.
2. Register the user.
3. Create a device for each port of the CONVERSANT that will use CTI.

See CentreVu Computer Telephony Release 9.1, Version 1, Telephony Services and CallVisor PC Installation for details on how to administer the CVCT server.

Setup and administration of a Siebel Client

If a Siebel client will be used, see CentreVu ® CT Integration for SIEBEL ® eBusiness Applications (Release 1.1, Version 1.2.205) Client Installation Guide. That guide will provide information and references regarding the related setup and administration of both the client and the CVCT server.
This chapter describes and demonstrates the use of CTI external functions.

**Note:**
For information about using CTI with Siebel eBusiness applications, see *CONVERSANT Version 8.0 Application Development with Siebel eBusiness*, 585-310-784.

## External functions

You must invoke CTI external functions to either change the state of a call or to move information about a call.

There are ten external functions available to Script Builder and Voice@Work:

- `ctiCallInfo`
- `ctiCallState`
- `ctiConfer`
- `ctiDial`
- `ctiDiscon`
- `ctiHold`
- `ctiNotify`
- `ctiPrivData`
- `ctiRetrieve`
- `ctiTransfer`

These functions are described in detail on the following pages.
**ctiCallInfo**

**Description**
The *ctiCallInfo* external function provides the call ID, ANI, called number, and CONVERSANT port extension associated with an active call at the CONVERSANT.

**Output**

- **Call ID** - Script variable where the call ID will be stored. (Type: num)
- **ANI** - Script variable where the ANI of the call will be stored. (Type: char, Size: 16)
- **Called Number** - Script variable where the Called Number or DNIS will be stored. (Type: char, Size: 16)
- **Station Extension** - Script variable where the CONVERSANT Port extension will be populated. (Type: char, Size: 7)
- **Skill** - Script variable where the skill/split hunt-group number will be stored. (Type: char, Size: 7)

**Return values**
The return code indicates whether the CTI DIP can find any call present at the CONVERSANT port.

- If successful, the result is > 0.
- On failure, the result is <=0.

**Usage notes**
The *ctiCallInfo* function is often the first external function that is called in a script. It identifies the call ID of the call that has reached the CONVERSANT. This is useful for any controlling function that might be used later in the script, such as placing the call on hold (*ctiHold*). It also identifies the number that was dialed by the caller. If the call was inbound from an ISDN trunk group, it also identifies the DNIS associated with the call. This feature is most useful in scripts that dynamically assign a CONVERSANT script on DNIS.

If there is more than one call present at a CONVERSANT port, *ctiCallInfo* reports the latest call. For example, if one call is on hold while another call is active, *ctiCallInfo* reports only on the most recent call to
arrive at the CONVERSANT port, regardless of whether this call is on hold or not.

Example

    External Function
        Function Name: ctiCallInfo
        Use Arguments: cust_callid ani calledNum station_id skill
ctiCallState

Description

ctiCallState provides the state of all the calls present at the CONVERSANT port extension.

Output

Call ID of Call #2 - Script variable where the call ID of call #2 will be populated.
(Type: num)

Call ID of Call #3 - Script variable where the call ID of call #2 will be populated.
(Type: num)

State of Call #1 - Script variable where the state of call #1 will be populated. See States of calls
(Type: num)

State of Call #2 - Script variable where the state of call #2 will be populated. See States of calls
(Type: num)

State of Call #3 - Script variable where the state of call #2 will be populated. See States of calls
(Type: num)

Return values

The return code indicates whether the CTI DIP can find a call present at the CONVERSANT port.

- If successful, the result is the call ID of call #1.
- On failure, the result is -1.

Usage notes

A CONVERSANT port can support up to 3 simultaneous calls. This external function can be used to determine the call ID and associated
state (ALERTING, HELD, ESTABLISHED) of the calls present at the port. The states are represented by values:

### States of calls

<table>
<thead>
<tr>
<th>Value</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>1</td>
<td>Delivered</td>
</tr>
<tr>
<td>2</td>
<td>Established</td>
</tr>
<tr>
<td>3</td>
<td>Held</td>
</tr>
<tr>
<td>4</td>
<td>Conferenced</td>
</tr>
<tr>
<td>5</td>
<td>Transferred</td>
</tr>
<tr>
<td>6</td>
<td>Initiated</td>
</tr>
<tr>
<td>7</td>
<td>Originated</td>
</tr>
</tbody>
</table>

**Note:**
The ordering of the calls is determined by the arrival order. Call #1 is most likely the first call received by the CONVERSANT.

**Example**

External Function

- Function Name: ctiCallState
- Use Arguments: call2id  call3id  call1state  call2state  call3state
- Return Field: retcode
ctiConfer

Description

ctiConfer requests that two calls present at a CONVERSANT port be conferenced together.

Input

Call ID of held call - Call ID of held call.
(Type: num)

Call ID of active call - Call ID of active call.
(Type: num)

Return values

The return code indicates whether the two calls were conferenced.

If successful, the result is >0.

On failure, the result is <=0.

Usage notes

ctiConfer requires one of the calls involved in the transfer be on hold at the CONVERSANT port prior to execution.

Example

External Function
Function Name: ctiConfer
Use Arguments: cust_callid interp_callid
Return Field: retcode
**ctiDial**

**Description**

**ctiDial** requests a call be placed from the CONVERSANT port to a destination.

**Input**

*Phone number* - The destination number to be dialed.
(Type: char, Size: 64)

*User-user-information* - Up to 96 bytes of user-provided data.
(Type: char, Size: 96)

**Return values**

The return code indicates whether the call was successfully placed.

- If successful, the result is >0.
- On failure, the result is <=0.

**Usage notes**

There are no special provisions for this function at this time.

**Example**

```plaintext
External Function
Function Name: ctiDial
Use Arguments: "5552499"  UUI
Return Field: retcode
```
**ctiDiscon**

**Description**

*ctiDiscon* requests a call be placed from the CONVERSANT port to disconnect a call.

**Input**

`call ID` - The call ID to be disconnected.

(Type: num)

**Return values**

The return code indicates whether the call was successfully disconnected.

- If successful, the result is >0.
- On failure, the result is <=0.

**Usage notes**

*ctiDiscon* is best used in conjunction with the *ctiCallState* function to determine the state of a call before it is disconnected.

Also, the CONVERSANT port itself will not detect any dial tone on the line after the call is disconnected. If the script needs to terminate before the call is disconnected, it must explicitly do so after a successful return.

**Example**

```plaintext
External Function
Function Name: ctiDiscon
Use Arguments: call2id
Return Field: retcode
```
**ctiHold**

**Description**

`ctiHold` requests that a call that is active at the CONVERSANT port be placed on hold.

**Input**

*Call ID* - The ID of the call to be placed on hold.

*(Type: num)*

**Output**

*Error Code* - Script variable that is populated with an error code if the operation fails.

*(Type: num)*

**Return values**

The return code indicates whether the call was successfully placed on hold.

- If successful, the result is >0.
- On failure, the result is <=0.

**Usage notes**

`ctiHold` is best used in conjunction with the `ctiCallState` function to determine the state of a call before it is placed on hold.

If an error is encountered, the CTI DIP will populate the *Error Code* field. The value will be of type CSTA_Universal_Failure_t. Consult CentreVu CT documentation for more details about this error code.

**Example**

```
External Function
  Function Name: ctiHold
  Use Arguments: cust_callid  error
  Return Field: retcode
```
**ctiNotify**

**Description**

ctiNotify provides answer notification to a voice script that has launched a call.

**Input**

None.

**Return values**

The return code indicates whether a tracked call has been answered.

- If successful, the result is the call ID of the established call, which means that the tracked call is answered.
- On failure, the result is -1, which means that the CTI DIP found no call at the CONVERSANT port to track.

**Usage notes**

**ctiNotify** is most useful in outbound calling when used in conjunction with **ctiDial**. The function is solely dependent on receiving answer supervision on a timely basis from the network. Therefore, if the call is sent over a non-ISDN trunk to a destination, the function may return several seconds after the call is actually connected.

When the **ctiNotify** request is made, a message is sent to the CTI DIP. The CTI DIP checks the status of the calls that are currently at the CONVERSANT port.

If a call is found to be either in the DELIVERED or ORIGINATED state, the CTI DIP:

- Tags this call to receive answer notification.
- Tracks the call that has been tagged.
- When the tracked call is answered, sends the call ID of the answered call to the script.
- If no call is currently in either the DELIVERED or ORIGINATED state, the CTI DIP returns a "-1" to **ctiNotify**. If **ctiNotify** receives a "-1," it tries up to three times to find a call at the CONVERSANT port that is in either the DELIVERED or ORIGINATED state. There is a 500-millisecond pause between attempts.

**Example**

```plaintext
External Function
Function Name: ctiNotify
Return Field: interp_callid
```
ctiPrivData

Description

ctiPrivData requests that any private data associated with the call at the CONVERSANT port be provided. Currently this function supports User-to-User Information (UUI) and Universal Call ID (UCID).

Output

Private Data - A buffer that the CTI DIP will populate with private data associated with a call.

(Type: char, size: up to 96 characters)

Input

Type - A constant that represents the type of private data that is requested

(Type: num)

0 = UUI

1 = II DIGITS (not currently implemented)

2 = UCID

Call ID - The call ID of the current call at the CONVERSANT port.

(Type: num)

Vendor - The PBX vendor (for example, Lucent).

(Type: char, size: 32)

Return values

The return code indicates whether the CTI DIP successfully processed the message.

If successful, the result is 1.

On failure, the result is <=0, which indicates that no private data exists.

Usage notes

The information provided by this function is vendor specific. Currently, the only vendor private data that is implemented is per Lucent Technologies.

Universal Call ID (UCID) requires DEFINITY 6.3 or later and CentreVu CT 3.10 Version 2.0 or later.
Example

External Function
Function Name: ctiPrivData
Use Arguments: PDATA 0 CALLID "LUCENT"
Return Field: retcode
ctiRetrieve

Description

*ctiRetrieve* requests that a held call at the CONVERSANT port be retrieved.

Input

**Call ID** - Call ID of the held call

(Type: num)

Output

**Error Code** - Script variable where the error code from the 
CSTA_UNIVERSAL_FAILURE_CONF event will be populated (only if the 
operation fails)

(Type: num)

Return values

The return code indicates whether the call was successfully retrieved.

If successful, the result is > 0, which indicates that the CTI DIP 
received a CSTA_RETRIEVE_CALL_CONF event.

On failure, the result is <=0, which indicates that the call was not 
successfully retrieved.

Usage notes

This function is typically used only to recover from an error, for example, 
if the CONVERSANT dials a busy number and the script needs to 
reconnect to the held party to inform them of the busy party.

Example

External Function

Function Name: ctiRetrieve

Use Arguments: cust_callid  error

Return Field:  retcode
ctiTransfer

**Description**

*ctiTransfer* requests that two calls present at a CONVERSANT port be connected.

**Input**

*Call ID of held call* - Call ID of held call
(Type: num)

*Call ID of active call* - Call ID of active call
(Type: num)

**Return values**

The return code indicates whether the transfer occurred.

- If successful, the result is >0, which indicates that the CTI DIP received a CSTA_TRANSFERRED_CALL_CONF event.
- On failure, the result is <=0, which indicates that the transfer failed to occur.

**Usage notes**

*ctiTransfer* requires that one of the calls involved in the transfer be on hold at the CONVERSANT port prior to the execution of this function.

**Example**

```
External Function
Function Name: ctiTransfer
Use Arguments: cust_callid  interp_callid
Return Field:  retcode
```
Sample application

The use of the CTI DIP will now be illustrated by a very simple, sample application, written in Script Builder. Comments are provided to identify what the application is doing.

start:
1. Answer Phone
2. External Function
   Function Name: ctiCallInfo
   Use Arguments: cust_callid ani calledNum station_id
   skill
   Return Field: retcode
#=================================================
# IF no CTI info, application cannot service call
# then recall the error message in the log table
# and quit.
#=================================================
3. Evaluate
   If retcode <= 0
4. Set Field Value
   Field: szDescription = "cannot get info on ctiCallInfo"
   Field: log_type = "CTICALLINFO_ERROR"
5. External Function
   Function Name: u_datetime
   Use Arguments: date time $UNIX_TIME
6. Modify Table
   Table Name: LOG Operation: Add
   Field: DATE = date
   Field: TIME = time
   Field: CHANNEL = $CHANNEL_NUMBER
   Field: DESCRIPTION = szDescription
   Field: ERRORCODE = log_type
7. Goto QUIT
End Evaluate

#Answer Phone if getting CTI info...

#========================================
#. ....Initial Customer Greeting
#========================================

GREET_CUSTOMER:
8. Announce
   Speak Without Interrupt
   Text: "Welcome to CTI test Call Center "
   Text: "Please wait while we transfer to our agent "
#========================================
# Hold the current call while waiting to transfer.
#=================================================

9. External Function
   Function Name: ctiHold
   Use Arguments: cust_callid  error
   Return Field:  retcode
   #=================================================================
   # If no CTI hold failed, then recall the error
   # message in the log table
   # and quit.
   #=================================================================

10. Evaluate
    If retcode  <= 0
11.     Set Field Value
           Field: szDescription = "cannot place call on ctiHold"
           Field: log_type = "CTIHOLD_ERROR"
12.     External Function
           Function Name: u_datetime
           Use Arguments: date  time  $UNIX_TIME
13.     Modify Table
           Table Name: LOG Operation: Add
           Field: DATE = date
           Field: TIME = time
           Field: CHANNEL = $CHANNEL_NUMBER
           Field: DESCRIPTION = szDescription
           Field: ERRORCODE = log_type

14.     Goto QUIT
    End Evaluate

#=======================================
# Set UUI information
#=======================================

15. External Action: Concat5
    Destination:  UUI
    String1:  "This is CTI UUI Information."
    String2:  "Your account number is 23456."
    String3:  ""
    String4:  ""
    String5:  ""
    Max_Destination_Length:  255
    Return Field:  retcode
    End External Action

#=======================================
# Use ctiDial to pass UUI information
#=======================================

16. External Function
    Function Name: ctiDial
    Use Arguments: "5552499"  UUI
    Return Field:  retcode
17. Evaluate
If retcode  <= 0
18.     Set Field Value
        Field: szDescription = "cannot place call on ctiDial"
        Field: log_type = "CTIDIAL_ERROR"
19.     External Function
        Function Name: u_datetime
        Use Arguments: date  time  $UNIX_TIME
20.     Modify Table
        Table Name: LOG Operation: Add
        Field: DATE = date
        Field: TIME = time
        Field: CHANNEL = $CHANNEL_NUMBER
        Field: DESCRIPTION = szDescription
        Field: ERRORCODE = log_type
21.     Goto QUIT
End Evaluate

#=========================================  
# Use ctiNotify to get the second callid
#=========================================  
22.    External Function
        Function Name: ctiNotify
        Return Field:  interp_callid

#=============================================  
# Decide if the call has been delivered 
# If not delivered or originated, then:
# (1) disconnect the second call, 
# (2) retrieve the first call and try again
#=============================================  
23.    Evaluate
        If interp_callid  <  0
24.     Set Field Value
        Field: szDescription = "cannot get callid on ctiNotify"
        Field: log_type = "CTINOTIFY_ERROR"
25.     External Function
        Function Name: u_datetime
        Use Arguments: date  time  $UNIX_TIME
26.     Modify Table
        Table Name: LOG Operation: Add
        Field: DATE = date
        Field: TIME = time
        Field: CHANNEL = $CHANNEL_NUMBER
        Field: DESCRIPTION = szDescription
        Field: ERRORCODE = log_type

#=============================================  
# Use ctiCallState to get the second call id
#=============================================  
27.    External Function
Function Name: ctiCallState
Use Arguments: call2id call3id call1state call2state call3state

Return Field: retcode

28. Evaluate
   If retcode <= 0
5. Set Field Value
   Field: szDescription = "Cannot get call state info"
   Field: log_type = "CTICALLSTATE_ERROR"

30. External Function
    Function Name: u_datetime
    Use Arguments: date time $UNIX_TIME

31. Modify Table
    Table Name: LOG Operation: Add
    Field: DATE = date
    Field: TIME = time
    Field: CHANNEL = $CHANNEL_NUMBER
    Field: DESCRIPTION = szDescription
    Field: ERRORCODE = log_type

32. Goto QUIT
End Evaluate

#Disconnect the dialed call first

33. External Function
    Function Name: ctiDiscon
    Use Arguments: call2id
    Return Field: retcode

34. Evaluate
   If retcode <= 0
5. Set Field Value
   Field: szDescription = "Cannot disconnect the call"
   Field: log_type = "CTIDISCONN_ERROR"

36. External Function
    Function Name: u_datetime
    Use Arguments: date time $UNIX_TIME

37. Modify Table
    Table Name: LOG Operation: Add
    Field: DATE = date
    Field: TIME = time
    Field: CHANNEL = $CHANNEL_NUMBER
    Field: DESCRIPTION = szDescription
    Field: ERRORCODE = log_type

38. Goto QUIT
End Evaluate

#===========================
# Retrieve the first call
#===========================

39. External Function
    Function Name: ctiRetrieve
    Use Arguments: cust_callid error
Return Field: retcode
40. Evaluate
   If retcode <= 0
41. Set Field Value
    Field: szDescription = "cannot retrieve on cttiretrieve"
    Field: log_type = "CTIRETRIEVE_ERROR"
42. External Function
    Function Name: u_datetime
    Use Arguments: date time $UNIX_TIME
43. Modify Table
    Table Name: LOG Operation: Add
    Field: DATE = date
    Field: TIME = time
    Field: CHANNEL = $CHANNEL_NUMBER
    Field: DESCRIPTION = szDescription
    Field: ERRORCODE = log_type
44. Goto QUIT
End Evaluate
45. Announce
   Speak With Interrupt
   Text: "Sorry, all agents are busy now."
   Text: "Please wait while we transfer to the next available agent."

#============================================
# Retrieve the first call as an active call
# Hold the first call again.
#============================================
46. External Function
    Function Name: ctiHold
    Use Arguments: cust_callid error
    Return Field: retcode
47. Evaluate
   If retcode <= 0
48. Set Field Value
    Field: szDescription = "cannot place call on ctiHold"
    Field: log_type = "CTIHOLD2_ERROR"
49. External Function
    Function Name: u_datetime
    Use Arguments: date time $UNIX_TIME
50. Modify Table
    Table Name: LOG Operation: Add
    Field: DATE = date
    Field: TIME = time
    Field: CHANNEL = $CHANNEL_NUMBER
    Field: DESCRIPTION = szDescription
    Field: ERRORCODE = log_type
51. Goto QUIT
End Evaluate
52. External Function
    Function Name: ctiDial
Use Arguments: "5552499" UUI
Return Field: retcode
53. Evaluate
   If retcode <= 0
54. Set Field Value
   Field: szDescription = "cannot place call on ctiDial"
   Field: log_type = "CTIDIAL2_ERROR"
55. External Function
   Function Name: u_datetime
   Use Arguments: date time $UNIX_TIME
56. Modify Table
   Table Name: LOG Operation: Add
   Field: DATE = date
   Field: TIME = time
   Field: CHANNEL = $CHANNEL_NUMBER
   Field: DESCRIPTION = szDescription
   Field: ERRORCODE = log_type
57. Goto QUIT
    End Evaluate
58. External Function
   Function Name: sleep
   Use Arguments: 2

#==========================================
# Use ctiCallState to get second call id
#==========================================
59. External Function
   Function Name: ctiCallState
   Use Arguments: interp_callid call3id call1state call2state call3state
   Return Field: cust_callid
60. Evaluate
   If retcode <= 0
61. Set Field Value
   Field: szDescription = "Cannot get call state info"
   Field: log_type = "CTICALLSTATE2_ERROR"
62. External Function
   Function Name: u_datetime
   Use Arguments: date time $UNIX_TIME
63. Modify Table
   Table Name: LOG Operation: Add
   Field: DATE = date
   Field: TIME = time
   Field: CHANNEL = $CHANNEL_NUMBER
   Field: DESCRIPTION = szDescription
   Field: ERRORCODE = log_type
64. Goto QUIT
    End Evaluate

End Evaluate
# Use ctiTransfer to merge the first and second calls
# #======================================================

65. External Function
   Function Name: ctiTransfer
   Use Arguments: cust_callid interp_callid
   Return Field: retcode

66. Evaluate
   If retcode <= 0

67. Set Field Value
   Field: szDescription = "cannot transfer on ctiTransfer"
   Field: log_type = "CTITRANSFER_ERROR"

68. External Function
   Function Name: u_datetime
   Use Arguments: date time $UNIX_TIME

69. Modify Table
   Table Name: LOG Operation: Add
   Field: DATE = date
   Field: TIME = time
   Field: CHANNEL = $CHANNEL_NUMBER
   Field: DESCRIPTION = szDescription
   Field: ERRORCODE = log_type

70. Goto QUIT
End Evaluate

QUIT:
# #================================================
# Use the ctiCallState to get the state of all active calls on the current CONVERSANT port
# If any active calls exist, then use ctiDiscon to disconnect the call
# #================================================

71. External Function
   Function Name: ctiCallState
   Use Arguments: CallID2 CallID3 State1 State2 State3
   Return Field: CallID1

72. Evaluate
   If CallID1 != 0

73. External Function
   Function Name: ctiDiscon
   Use Arguments: CallID1
   Return Field: RC
End Evaluate

74. Evaluate
   If CallID2 != 0

75. External Function
   Function Name: ctiDiscon
   Use Arguments: CallID2
   Return Field: RC
End Evaluate

76. Evaluate
If CallID3 ≠ 0

77. External Function
    Function Name: ctiDiscon
    Use Arguments: CallID3
    Return Field: RC
    End Evaluate

78. Disconnect
79. Quit
## Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Adjunct/Switch Application Interface (ASAI)</strong></td>
<td>An optional feature that provides an ISDN-based interface with Avaya PBXs and adjunct processors.</td>
</tr>
<tr>
<td><strong>CentreVu Computer Telephony (CVCT)</strong></td>
<td>Also CentreVu CT. Avaya’s implementation of Computer Telephony Integration with TSAPI.</td>
</tr>
<tr>
<td><strong>Computer Telephony Integration (CTI)</strong></td>
<td>The generation, reception, and management of telephone calls with a computer. This functionality within a CONVERSANT enables communication with a CentreVu Computer Telephony server, using the CTI DIP.</td>
</tr>
<tr>
<td><strong>Computer Telephony Integration Data Interface Process (CTI DIP)</strong></td>
<td>Software within a CONVERSANT that lets applications communicate with a CVCT server.</td>
</tr>
<tr>
<td><strong>data interface process (DIP)</strong></td>
<td>A software process that communicates with interactive voice response (IVR) applications.</td>
</tr>
<tr>
<td><strong>DEFINITY</strong></td>
<td>A brand of PBX produced by Avaya.</td>
</tr>
<tr>
<td><strong>Local Area Network (LAN)</strong></td>
<td>A data communications network in a limited geographical area. The LAN provides communications between computers and peripherals.</td>
</tr>
<tr>
<td><strong>Multi-Application Platform for DEFINITY (MAPD)</strong></td>
<td>A DEFINITY circuit card that handles access to LANs and adjuncts.</td>
</tr>
<tr>
<td><strong>Network Interface Card (NIC)</strong></td>
<td>A circuit card that enables communication between a LAN and a workstation.</td>
</tr>
<tr>
<td><strong>Private Branch Exchange (PBX)</strong></td>
<td>A private switching system, either manual or automatic, that usually serves an organization such as a business or government agency. It is usually located on the customer premises.</td>
</tr>
<tr>
<td><strong>Telephony Server (TSRV)</strong></td>
<td>Also: “T-server”. A computer that connects a LAN with a PBX or other telephony device.</td>
</tr>
<tr>
<td><strong>Telephony Server Application Programming Interface (TSAPI)</strong></td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>An application programming interface that can monitor and control many aspects of computerized telephony.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Universal Call ID (UCID)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>UCID provides a unique identifier (8-byte binary or 20-character ASCII) for every call in a DEFINITY call center customer environment. UCID allows for uniform data-tracking for all call-related data in a call center, regardless of the system. DEFINITY uses the ASAI interface to pass the UCID to the CVCT Server.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>User-to-User Information element (UUI)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>UUI allows the customer to specify additional information to be passed in external function arguments, which can contain up to 96 bytes of information (compare with the 32 bytes available using ASAI).</td>
</tr>
</tbody>
</table>
How Are We Doing?

Document Title: CONVERSANT Version 8.0 Computer Telephony Integration
Document No.: Replace variable w/ doc #
Issue No. Issue 1
Date: September 2001

We welcome your feedback on this document. Your comments are of great value in helping us to improve our documentation.

Please rate the effectiveness of this document in the following areas:

<table>
<thead>
<tr>
<th></th>
<th>Excellent</th>
<th>Good</th>
<th>Fair</th>
<th>Poor</th>
<th>Not Applicable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ease of Use</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clarity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Completeness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Organization</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Examples</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Illustration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Please check the ways you feel we could improve this document:

- [ ] Improve the overview/introduction
- [ ] Improve the table of contents
- [ ] Improve the organization
- [ ] Include more figures
- [ ] Add more examples
- [ ] Add more detail
- [ ] Make it more concise/brief
- [ ] Add more step-by-step procedures/tutorials
- [ ] Add more troubleshooting information
- [ ] Make it less technical
- [ ] Add more/better quick reference aids
- [ ] Improve the index

Please provide details for the suggested improvement.

____________________________________________________________________________
____________________________________________________________________________

What did you like most about this document?

____________________________________________________________________________
____________________________________________________________________________

Feel free to write any comments below or on an attached sheet.

____________________________________________________________________________
____________________________________________________________________________
____________________________________________________________________________
____________________________________________________________________________

If we may contact you concerning your comments, please complete the following:

Name: _______________________________ Telephone Number: ________________________
Company/Organization: _____________________________ Date: ________________________

When you have completed this form, please fax to +1-xxx-xxx-xxxx.