

Lucent Technologies
Bell Labs Innovations



***CentreVu*[®] Call Management System**

Release 3 Version 6

Open Database Connectivity

585-215-852
Comcode 108144791
Issue 1
May 1998

Copyright © 1998, Lucent Technologies
All Rights Reserved
Printed in U.S.A.

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.

Your Responsibility for Your System's Security

Toll fraud is the unauthorized use of your telecommunications system by an unauthorized party, for example, persons other than your company's employees, agents, subcontractors, or persons working on your company's behalf. Note that there may be a risk of toll fraud associated with your telecommunications system and, if toll fraud occurs, it can result in substantial additional charges for your telecommunications services.

You and your system manager are responsible for the security of your system, such as programming and configuring your equipment to prevent unauthorized use. The system manager is also responsible for reading all installation, instruction, and system administration documents provided with this product in order to fully understand the features that can introduce risk of toll fraud and the steps that can be taken to reduce that risk. Lucent Technologies does not warrant that this product is immune from or will prevent unauthorized use of common-carrier telecommunication services or facilities accessed through or connected to it. Lucent Technologies will not be responsible for any charges that result from such unauthorized use.

Lucent Technologies 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.

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 instruction manual, 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: 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 prescrites dans le règlement sur le brouillage radioélectrique édicté par le ministère des Communications du Canada.

Trademarks

- DEFINITY is a registered trademark of Lucent Technologies.
- CentreVu is a registered trademark of Lucent Technologies.
- CONVERSANT is a registered trademark of Lucent Technologies.
- Informix is a registered trademark of Informix Software, Inc.
- Intel is a registered trademark of Intel.
- Microsoft, MS, MS-DOS, Windows, Windows 95, Windows NT, and Access are registered trademarks of Microsoft Corporation.
- OpenLink is a trademark of OpenLink Software.
- Crystal Reports is a trademark of SeaGate Software.
- Solaris is a trademark of Sun Microsystems, Inc.
- SPARC trademarks, including the SCD compliant logo, are trademarks or registered trademarks of SPARC International, Inc. SPARCstation, SPARCserver, SPARCengine, SPARCworks, and SPARCcompiler are licensed exclusively to Sun Microsystems, Inc. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.
- Sun and Sun Microsystems are trademarks or registered trademarks of Sun Microsystems, Inc.
- Ultra Enterprise 3000 and Ultra 5 are trademarks of Sun Microsystems, Inc.
- UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.
- All other products mentioned herein are the trademarks of their respective owners.

Ordering Information

Call: Lucent Technologies Publications Center
Voice: 1-800-457-1235
International Voice: 317-322-6416
Fax: 1-800-457-1764
International Fax: 317-322-6699

Write: Lucent Technologies BCS Publications Center
2855 N. Franklin Road
Indianapolis, IN 46219

Order: Document No. 585-215-852
Comcode 108144791
Issue 1, May 1998

Acknowledgment

This document was developed by Lucent Technologies Network Systems Customer Training and Information Products Organization.



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 Lucent Technologies Publications Center.

Warranty

Lucent Technologies provides a limited warranty on this product. Refer to the "Limited use Software License Agreement" card provided with your package.

European Union Declaration of Conformity

Lucent Technologies Business Communications Systems declares that XXX 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.

Disclaimer

Intellectual property related to this product (including trademarks) and registered to AT&T Corporation has been transferred to Lucent Technologies Incorporated.

Any references within this text to American Telephone and Telegraph Corporation or AT&T should be interpreted as references to Lucent Technologies Incorporated. The exception is cross references to books published prior to December 31, 1996, which retain their original AT&T titles.

Heritage Statement

Lucent Technologies—formed as a result of AT&T's planned restructuring—designs, builds, and delivers a wide range of public and private networks, communication systems and software, consumer and business telephone systems, and microelectronics components. The world-renowned Bell Laboratories is the research and development arm for the company.

Comments

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

How Are We Doing?

Document Title: *CentreVu™* Call Management System Release 3 Version 6
 Open Database Connectivity

Document No.: 585-215-852 Issue 1 Date: May 1998

Lucent Technologies welcomes your feedback on this document. Your comments are of great value in helping us to improve our documentation.

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

	Excellent	Good	Fair	Poor	Not Applicable
Ease of Use					////////////////////
Clarity					////////////////////
Completeness					////////////////////
Accuracy					////////////////////
Organization					////////////////////
Appearance					////////////////////
Examples					////////////////////
Illustration					
Overall Satisfaction					////////////////////

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

- | | |
|--|---|
| <input type="checkbox"/> Improve the overview/introduction | <input type="checkbox"/> Make it more concise/brief |
| <input type="checkbox"/> Improve the table of contents | <input type="checkbox"/> Add more step-by-step procedures/tutorials |
| <input type="checkbox"/> Improve the organization | <input type="checkbox"/> Add more troubleshooting information |
| <input type="checkbox"/> Include more figures | <input type="checkbox"/> Make it less technical |
| <input type="checkbox"/> Add more examples | <input type="checkbox"/> Add more/better quick reference aids |
| <input type="checkbox"/> Add more detail | <input type="checkbox"/> Improve the index |

Please provide details for the suggested improvement. _____

3. What did you like most about this document?

4. 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 fold, tape, and return to address on back or you can fax the form to 303-538-2195.

Do Not Tear — Fold Here Last, Then Tape the Long Edge and Both Ends of the Mailer

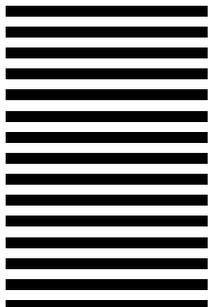


Lucent Technologies
Department JCOE11400
1200 West 120th Avenue
Westminster, CO 80234-2795

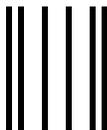
POSTAGE WILL BE PAID BY ADDRESSEE

FIRST CLASS PERMIT NO. 1999 GREENSBORO, NC

BUSINESS REPLY MAIL



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



Lucent Technologies
Bell Labs Innovations

Do Not Tear — Fold Here First

How Are We Doing?

Document Title: *CentreVu™* Call Management System Release 3 Version 6
 Open Database Connectivity

Document No.: 585-215-852 Issue 1 Date: May 1998

Lucent Technologies welcomes your feedback on this document. Your comments are of great value in helping us to improve our documentation.

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

	Excellent	Good	Fair	Poor	Not Applicable
Ease of Use					////////////////////
Clarity					////////////////////
Completeness					////////////////////
Accuracy					////////////////////
Organization					////////////////////
Appearance					////////////////////
Examples					////////////////////
Illustration					
Overall Satisfaction					////////////////////

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

- | | |
|--|---|
| <input type="checkbox"/> Improve the overview/introduction | <input type="checkbox"/> Make it more concise/brief |
| <input type="checkbox"/> Improve the table of contents | <input type="checkbox"/> Add more step-by-step procedures/tutorials |
| <input type="checkbox"/> Improve the organization | <input type="checkbox"/> Add more troubleshooting information |
| <input type="checkbox"/> Include more figures | <input type="checkbox"/> Make it less technical |
| <input type="checkbox"/> Add more examples | <input type="checkbox"/> Add more/better quick reference aids |
| <input type="checkbox"/> Add more detail | <input type="checkbox"/> Improve the index |

Please provide details for the suggested improvement. _____

3. What did you like most about this document?

4. 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 fold, tape, and return to address on back or you can fax the form to 303-538-2195.

Do Not Tear — Fold Here Last, Then Tape the Long Edge and Both Ends of the Mailer

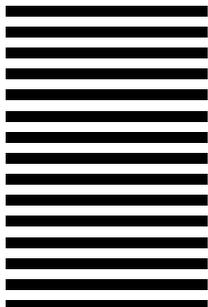


Lucent Technologies
Department JCOE11400
1200 West 120th Avenue
Westminster, CO 80234-2795

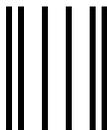
POSTAGE WILL BE PAID BY ADDRESSEE

FIRST CLASS PERMIT NO. 1999 GREENSBORO, NC

BUSINESS REPLY MAIL



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



Lucent Technologies
Bell Labs Innovations

Do Not Tear — Fold Here First

CentreVu® Call Management System

Release 3 Version 6

Open Database Connectivity

Table of Contents

Preface		
Overview		P-1
Organization of Document		P-1
Audience		P-1
Conventions Used in This Document		P-2
Terminology		P-2
Related Documents.		P-3
 Open Database Connectivity		
General Information.		1-1
Overview		1-1
ODBC Feature.		1-2
Overview		1-2
Queries		1-2
Performance Impact		1-3
Security		1-3
CMS Feature Interactions Requiring Client Support		1-3
ODBC Background and Functionality		1-5
Definition		1-5
Data Access Through ODBC		1-5
Structured Query Language (SQL)		1-6
CMS ODBC Drivers.		1-6
Uses for ODBC Data		1-7
Requesting Data Using ODBC		1-7
 ODBC Driver and Installation		
Overview		2-1
ODBC Driver		2-2
Introduction		2-2
Function		2-2
Additional Information		2-2
Installation and Configuration Procedures		2-3
Overview		2-3
Installation on the CMS Server		2-4
Introduction		2-4
System Requirements.		2-4
Installation Procedure		2-4
Server Utilities		2-6
Introduction		2-6
Debug Levels		2-6
Log Levels		2-6
Run ODBC		2-6

<i>Solaris</i> Client Installation	2-7
Introduction	2-7
System Requirements	2-7
Installation Procedure	2-7
Configuration Procedure	2-8
Test ODBC Connectivity	2-9
Build an ODBC Application	2-10
Desktop Computer Client Installation	2-11
Introduction	2-11
System Requirements	2-11
Installation	2-11
Desktop Computer Client Configuration	2-13
Introduction	2-13
ODBC Driver Administration Utility	2-13
Define Data Sources	2-15
Add System Data Sources	2-16
Add Login ID-Based Data Sources	2-17
Remove a Data Source	2-18
Configure a New ODBC Data Source	2-18
Administer ODBC Options	2-22
View Installed ODBC Drivers	2-23
Test Desktop Computer Connectivity	2-25
Introduction	2-25
Connect to and Access Data	2-25
Disconnect from a Data Source	2-26
Troubleshooting	2-27
Introduction	2-27
Server Log File	2-27
Client Trace	2-27
<i>OpenLink</i> Error Messages	2-27

Database Tables

Overview	3-1
How Database Items are Presented	3-2
Database Item Tables	3-2
Index Database Items	3-2
Data Types	3-3
Call-Based Data	3-3
Interval-Based Data	3-3
Column Types	3-4
ECS Cross-Reference Tables	3-4
Database Logic Structure	3-5
Terminology	3-6
CMS Feature Interactions	3-6
Interactions with ECS Features and Tracking of ECS Capabilities	3-6
Database Table Names	3-7
Split/Skill Database Items	3-9
Overview	3-9
Split/Skill Database Items	3-9
ECS Cross-Reference	3-35
Agent Database Items	3-39
Overview	3-39
Agent Database Items	3-39
ECS Cross-Reference	3-59

Trunk Group Database Items	3-63
Overview	3-63
Trunk Group Database Items	3-63
ECS Cross-Reference	3-74
Trunk Database Items	3-76
Overview	3-76
Trunk Database Items	3-76
ECS Cross-Reference	3-82
Vector Database Items	3-83
Overview	3-83
Vector Database Items	3-83
ECS Cross-Reference	3-92
VDN Database Items	3-94
Overview	3-94
VDN Database Items	3-94
ECS Cross-Reference	3-109
Call Work Codes Database Items	3-113
Overview	3-113
Call Work Codes Database Items	3-113
ECS Cross-Reference	3-115
Agent Login/Logout Database Items	3-116
Overview	3-116
Agent Login/Logout Database Items	3-116
ECS Cross- Reference	3-118
Agent Trace Database Items	3-120
Overview	3-120
Agent Trace Database Items	3-120
ECS Cross-Reference	3-123
Current Day Configuration Forecast Database Items	3-125
Overview	3-125
Current Day Configuration Forecast Database Items	3-125
ECS Cross-Reference	3-126
Current Day Forecast Report Database Items	3-127
Overview	3-127
Current Day Forecast Report Database Items	3-127
ECS Cross-Reference	3-129
Call Record Database Items	3-130
Overview	3-130
Call Record Database Items	3-130
ECS Cross-Reference	3-139
Data Collection Period Database Items	3-141
Overview	3-141
Data Collection Period Database Items	3-141
Archiver Execution Status Database Items	3-142
Overview	3-142
Archiver Execution Status Database Items	3-142
Customer Log Database Items	3-144
Overview	3-144
Customer Log Database Items	3-144
Agent Group Database Items	3-145
Overview	3-145
Agent Group Database Items	3-145
Synonyms Database Items	3-146
Overview	3-146

Synonyms Database Items	3-146
ACD Shifts Database Items	3-148
Overview	3-148
ACD Shifts Database Items	3-148
Database Items	3-149
Overview	3-149
Database Items	3-149
Exceptions Historical Database Items	3-150
Overview	3-150
Agent Exceptions Database Items	3-150
Disk Full Exceptions Database Items	3-153
Split/Skill Exceptions Database Items	3-154
Trunk Group Exceptions Database Items	3-155
VDN Exceptions Database Items	3-157
Vector Exceptions Database Items	3-158
Malicious Call Trace Exceptions Database Items	3-159
Data Collection Exceptions Database Items	3-161
 Glossary	 GL-1

Preface

Overview

Open Database Connectivity (ODBC) is an optional *CentreVu*® Call Management System (CMS) feature that allows you to access data in the *CentreVu* CMS database for use in other software applications such as spreadsheet programs. With ODBC, you can access the CMS data directly from your application, without needing to understand database connectivity or format.

This feature is especially useful for call centers with multiple sites. ODBC allows access to data at multiple sites for reports. ODBC uses Structured Query Language (SQL) to access data, eliminating the need to access the database in the language in which it was written.

The ODBC feature is a client/server feature; that is, the clients must be attached to a network that is fully functional to access the server. The “clients” are the computers that are accessing data through ODBC; the “server” is the CMS machine where the CMS database is located.

Organization of Document

Chapter 1 presents an overview of the ODBC feature, its interaction with other *CentreVu* CMS features, and its performance impact. Also included is an overview of ODBC functionality and its interaction with the *CentreVu* CMS database.

Chapter 2 presents an overview of the off-the-shelf driver that will be used for the ODBC feature and its function in relation to *CentreVu* CMS data. The majority of this chapter covers basic installation, configuration, and initialization procedures for the driver on the server and the clients.

Chapter 3 describes the *CentreVu* CMS open database tables and the items in the database tables.

Audience

This document is written for *CentreVu* CMS users who need to understand open access to database items. It is also written to help users decide which database items to use in custom reports, spreadsheets, and other user applications with external data access. The database items can be exported through ODBC for use in a variety of software programs. For data extraction purposes, users will need to have a basic understanding Structured Query Language (SQL) before using this document. Users should also have a basic understanding of database logic and purpose.

Conventions Used in This Document

The following conventions are used throughout this document:

<i>Related document callouts</i>	Include the document number (if a Lucent Technologies publication) and the book title in italic. For example, see the <i>CentreVu® CMS R3V5 Administration</i> (585-215-820) document for more information.
Courier font	Designates <i>Informix</i> * table names; for example, the <code>csplit</code> table.
Bold font	Designates something you type into an ODBC input screen. For example, type 60 in the Others: Row Buffer Size field.
“Chapter Names”	Are always in quotes.
<i>italic</i>	Highlights references to file names and directories. Also indicates trademarked and registered trademarked words.
Subsystem References	Are always initial capitalized. For example, set the access permissions for <i>CentreVu CMS</i> users from the User Permissions subsystem.
Window Titles	Are always capitalized.

*Informix is a registered trademark of Informix Software, Inc.

Terminology

The following terms are used in this document.

API	Application Programming Interface (API). For ODBC, an API is the user interface that can generate queries to a database.
CMS	Call Management System.
DBMS	Database Management System (DBMS). CMS uses the <i>Informix</i> DBMS.

Driver	An implementation of an ODBC API that supports a particular DBMS; for example, <i>Informix</i> DBMS. More specifically, a driver is a dynamic link library.
Driver Manager	A driver manager loads specific drivers based on the type of data being accessed.
Informix	A relational database in which CMS stores much of its data, including administration and historical data. More specifically, the CMS database is an <i>Informix</i> Standard Engine (SE).
ODBC	Open Database Connectivity. ODBC is an interface that allows applications to access data from a database using SQL.
SQL	Structured Query Language.

Related Documents

The following documents can help you use the *CentreVu* CMS software to its maximum capability. Those most closely related to this document are the Reports and Custom Reports documents.

- *CentreVu[®] CMS R3V6 Administration* (585-215-850)
- *CentreVu[®] CMS R3V6 Reports* (585-215-851)
- *CentreVu[®] CMS R3V6 Custom Reports* (585-215-852)
- *CentreVu[®] CMS R3V6 Change Description* (585-215-853)
- *CentreVu[®] CMS R3V6 External Call History Interface* (585-215-854)
- *CentreVu[®] CMS R3V6 Upgrades and Migrations* (585-215-856)
- *CentreVu[®] CMS R3V6 Sun^{*} SPARCserver[†] Computers Installation and Maintenance, Volume 1 and Volume 2* (585-215-857)
- *CentreVu[®] CMS R3V6 Sun SPARCserver Computers Connectivity Diagram* (585-215-858) *CentreVu[®] Report Designer Version 6 User Guide* (585-215-861)
- *CentreVu[®] Supervisor and Report Designer Version 6 Change Description* (585-215-862)
- *CentreVu[®] CMS R3V6 Documentation CD-ROM* (585-215-891).

*Sun is a registered trademark of Sun Microsystems, Inc.

†SPARCserver is a registered trademark of Sun Microsystems, Inc.

Open Database Connectivity

General Information

Overview

This chapter presents an overview of the *CentreVu*® Call Management System (CMS) Open Database Connectivity (ODBC) feature, its interaction with other *CentreVu* CMS features, and its performance impact. An overview of ODBC functionality and its interaction with the *CentreVu* CMS database is also included.

ODBC Feature

Overview

An *OpenLink*^{*} ODBC driver provides the ODBC communication and connectivity that allows external data access to the *CentreVu* CMS database, and is installed at the call center. The driver supports *Solaris*[†], *Windows 95*[‡], and *Windows NT*[§] clients. All historical CMS database tables, dictionary tables, and customer-provided tables can be accessed by ODBC clients through standard ODBC-enabled software applications. All standard Structured Query Language (SQL) queries generated by the user applications are supported by the ODBC driver as limited by permissions. Table-level, read-only permissions restrict access to database tables. The tables accessible in the *CentreVu* CMS Release 3 Version 6 database are described in detail in Chapter 3, "Database Tables."

Queries

Queries can be generated from client to server, from user applications. The ODBC driver is installed on the server, and the accompanying software is installed on the clients. The extracted data can be used in workforce management packages, network routers, and blended inbound/outbound applications. Any application that supports ODBC functionality can access permissible tables; for example, the CMS Dictionary.

The same recommendations for composing queries in the CMS custom report editor also apply to ODBC queries. For example, accessing large tables such as the split/skill or agent tables, or joining tables in queries may have a negative impact on CMS performance. Use the exact table and database item names when querying the database. When performing calculations, keep in mind that arithmetic operations are performed in order from left to right. Multiplication and division operations are performed before addition and subtraction operations, unless the addition or subtraction operations are enclosed in parentheses. Operations in parentheses are always performed first. Therefore, it is very important to review queries before sending them to the database.

*OpenLink is a trademark of *OpenLink* Software.

† Solaris is a trademark of Sun Microsystems, Inc.

‡Microsoft, MS, MS-DOS, Windows, and Windows 95 are registered trademarks of Microsoft Corporation

§ Windows NT is a registered trademark of Microsoft Corporation.

Additional Information on Queries

For more information on how to compose efficient database queries, see the “Edit | Queries” chapter in the *CentreVu® CMS R3V6 Report Designer* document, Lucent Technologies (585-215-861).

Performance Impact

The number, size, and types of queries received by the CMS may impact performance. To keep system impact at a minimum, certain types of queries, such as accessing large tables or performing table joins, should be run during a period of low agent and real-time report activity; for example, during off hours. Prioritize very involved queries the same way you prioritize reports during high business activity. The impact of complex or multiple queries on the database will be similar to running multiple reports. Another important performance impact consideration is the number of simultaneous database accesses allowed. To optimize system performance, minimize the number of database connects and disconnects from an application.

Security

ODBC users log into the CMS server with password protection. Users have SQL access to *Informix* tables as limited by the table permissions in Chapter 3, “Database Tables.” All historical and dictionary database tables have read-only access permission. The customer-created tables on the host (any table names beginning with “c_”) have read and write permissions. All other tables are not accessible through ODBC.

CMS Feature Interactions Requiring Client Support

The following CMS features require specific client administration and interaction. Most importantly, data received by the client application is in “raw” form; that is, it will need to be formatted for use within your application. Dictionary names and certain time fields are most affected by this formatting. Refer to Chapter 3, “Database Tables,” for further information on data format and values.

- *Dictionary Names*: Clients can access CMS Dictionary names. The client must map the synonym to the report. Underlying data is numeric; for example, different splits are stored as numbers and not by their names.

 **NOTE:**

To minimize performance impact when accessing synonyms, download the synonyms to your client application or database and perform the join at the client.

- *Permissions*: Applications which access *Informix* externally, such as database access scripts, may not work if the table permission script tries to access a table to which permission is denied. Note that CMS-like permissions to individual entities such as splits and vectors are not available through the ODBC interface. It is up to the client application to create and enforce permissions at this level if this is required.
- *Field Display*: The time and date data received from the database may not be formatted. Generally, times may be shown in seconds or in military format. Review data for formatting when importing it into your software application. See your software's documentation for further information on formatting data.

ODBC Background and Functionality

Definition

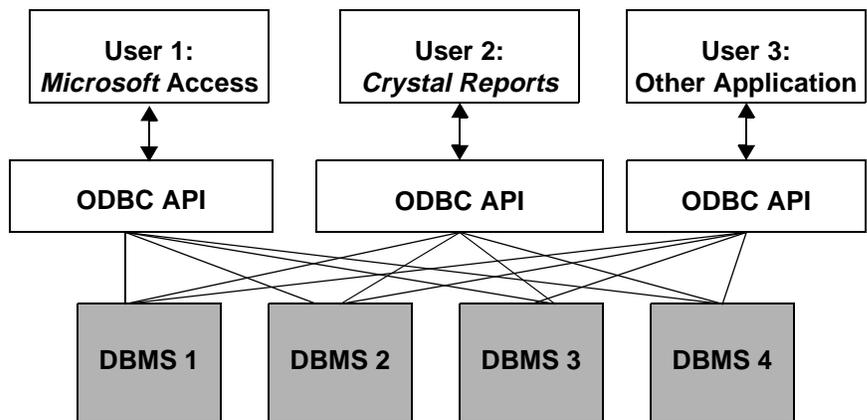
Open Database Connectivity (ODBC) is an Application Programming Interface (API) that interfaces with one or many Database Management Systems (DBMSs) and allows queries to access data in the database for extraction and use in reports and other outside applications.

Data Access Through ODBC

ODBC was developed as a solution to accessing different types of data. Within one call center, users may be working with different applications such as *Microsoft Access*^{*} or *Seagate Crystal Reports*[†], that must access call center data. Users may need to import CMS data into many different applications. ODBC provides a standard method of database access that shields users from the underlying functionality of network software, naming conventions, and the other complexities involved in accessing data through a DBMS. The data only needs to be queried through the embedded SQL query function in the *Windows*[‡] application that you are using. Refer to your specific application documentation for further information on its embedded SQL function.

Data Access Diagram

The following figure illustrates user data access through ODBC.



*Access is a trademark of Microsoft Corporation.

†Crystal Reports is a trademark of SeaGate Software.

‡Windows is a registered trademark of Microsoft Corporation.

Structured Query Language (SQL)

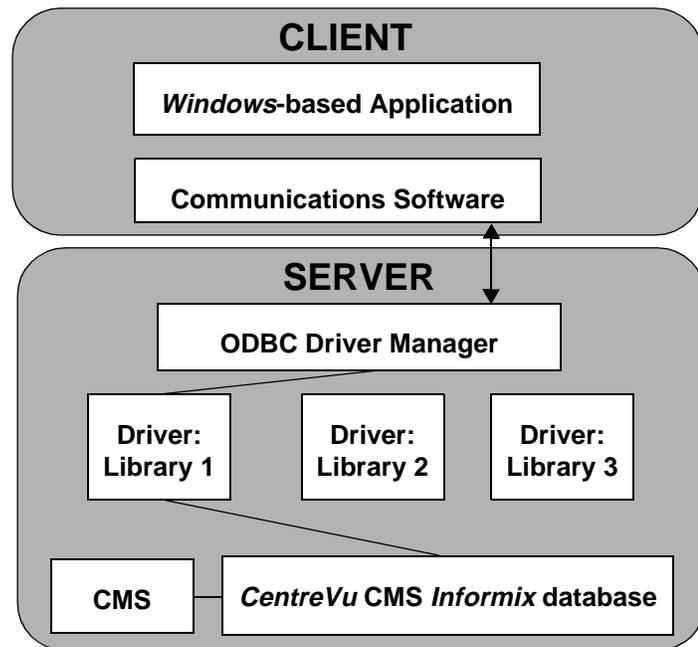
ODBC uses SQL to query and access data. Because SQL is a language, queries written in SQL can be used to access data with different formats. SQL is the basis for relational database access. The simplest description of a relational database model is the table—data is stored in rows and columns, and relationships between tables are established through data items with matching values between the tables. SQL queries access the data stored in the relational database tables and extract it for use in other applications. SQL can also be used to construct data calculations. This is useful if you want to look at a sum of the data; for example, the total number of calls routed to a particular split or skill. An SQL query is composed from the *Windows* application for which you need the data. The SQL query function is embedded in the application itself.

CMS ODBC Drivers

The *CentreVu CMS Informix* database management system is supported by the *OpenLink Multi-Tier ODBC Driver*. A driver is an implementation of an ODBC application programming interface that supports a particular database management system; in this case, the *Informix* DBMS used by *CentreVu CMS*. More specifically, a driver is a dynamic link library that is specific to a type of DBMS. The driver manager provides the link between the user's applications and the DBMS itself by selecting the dynamic link library using the format of the data being queried by processing the received ODBC function calls. More information on the *OpenLink* driver and its installation procedures can be found in Chapter 2, "ODBC Driver and Installation."

ODBC Driver Functionality

The following figure illustrates how drivers provide the interface between user applications and the data being accessed, assuming that driver dynamic link library 1 is the *Informix* specific library.



Uses for ODBC Data

Data extracted and stored by an ODBC application can be used by ODBC-enabled programs, such as workforce management packages, network routers, and blended inbound/outbound applications. Data can also be extracted for use in ODBC-enabled programs, including spreadsheets and report writers. An example of an ODBC data application is generating reports using data from multiple call center sites and their databases.

Requesting Data Using ODBC

The ODBC driver installed on the CMS is compatible with the CMS *Informix* database. All queries in ODBC must be composed to ensure continued CMS performance. The query is invoked differently in each *Windows* application. For more information on how your application interfaces with ODBC, refer to your application's user documentation. For more information on how to compose efficient database queries, see the "Edit | Queries" chapter in the *CentreVu® CMS R3V6 Report Designer* document, Lucent Technologies (585-215-861), and "Performance Impact" and "Structured Query Language (SQL)" sections in this chapter.

ODBC Driver and Installation

Overview

This chapter presents an overview of the driver used for the Open Database Connectivity (ODBC) feature and its function in relation to *CentreVu*® Call Management System (CMS) data. The second half of this chapter covers basic installation, configuration, and initialization procedures for the driver. Server connection procedures for the client are also given.

Procedures are included for personal computer clients, *Solaris* clients, and the *CentreVu* CMS server. Troubleshooting tips and *OpenLink* ODBC Error Messages are included at the end of the chapter.

All server and client driver components are included on the Call Management System *OpenLink* ODBC Driver compact disk. You will use this disk for both client and server installation.

 **CAUTION:**

If you choose to develop an application for the ODBC driver, Lucent Technologies cannot provide support for that application, or for any other third party software or related mapping.

ODBC Driver

Introduction

An *OpenLink* Multi-Tier ODBC driver facilitates client querying of data for extraction from the *CentreVu* CMS database. The client driver supports client access from *Solaris*, *Windows 95*, and *Windows NT* clients. All historical CMS database tables, dictionary tables, and customer-defined tables can be accessed by ODBC clients through standard Structured Query Language (SQL) queries. All standard SQL queries are supported by the ODBC driver as limited by permissions. Table-level permissions will restrict access to database tables. The tables accessible in *CentreVu* CMS Release 3 Version 6 are described in detail in Chapter 3, "Database Tables."

Function

OpenLink ODBC drivers are composed of two main components: a generic ODBC driver and the *OpenLink* Request Agent. The *OpenLink* Request Agent resides on the client and interfaces with the ODBC driver through proprietary protocol. ODBC drivers are controlled by an ODBC driver manager, in this case, the *OpenLink* Request Broker. The *OpenLink* Request Broker resides on the server. ODBC uses data source names as the link between the ODBC Request Broker and the relevant ODBC driver for a particular database.

The *OpenLink* ODBC driver is a multi-tier driver; that is, the driver controls both ODBC calls and the manner in which these calls are transferred to the relevant database management system (in *CentreVu* CMS, *Informix*) over the communications infrastructure. The *OpenLink* Request Broker provides the link between the user's applications and the database management system (DBMS—in CMS, *Informix*). Once it is installed and administered, the ODBC driver and its components are transparent to the client. The CMS ODBC feature allows multiple, synchronous accesses from clients, users, and/or applications.

Supported Logins

The CMS ODBC feature supports up to five or ten simultaneous logins, in increments of five, depending on your licensing agreement.

Additional Information

For additional information on the ODBC driver, see the on-line help file that is included on the Call Management System *OpenLink* ODBC Driver compact disk.

Installation and Configuration Procedures

Overview

The server and the client ODBC driver software will be installed from the CMS Call Management System *OpenLink* ODBC Driver compact disk. This disk contains all necessary files for server, *Solaris* client, and *Windows* client installation. *OpenLink* ODBC driver installation and configuration procedures in this chapter are presented for the *Sun Solaris/CentreVu* CMS server, *Sun Solaris* clients, and personal computer clients. Refer to the installation and configuration procedures that are specific to your client/server environment. Installation on your server will not impact system performance; the ODBC server components may be installed at any time.

- To install and configure the ODBC driver in a *Sun Solaris/CentreVu* CMS client/server environment, see “Installation on the CMS Server” and “Solaris Client Installation” in this chapter.
- To install and configure the ODBC driver on your CMS server and on your personal computer for accessing *CentreVu* CMS data, see “Installation on the CMS Server”, “Desktop Computer Client Installation”, and “Desktop Computer Client Configuration” in this chapter.
- For additional information on installing, configuring, and operating the ODBC driver, see the on-line help file that is included on the Call Management System *OpenLink* ODBC Driver compact disk.
- Once installed and configured, a valid CMS login has ODBC access.

Installation on the CMS Server

Introduction

This section describes and presents step-by-step instruction for the system requirements, software installation procedures, configuration procedures, and initialization procedures for the *OpenLink* Multi-Tier ODBC driver on the *CentreVu* Call Management System server.

⇒ NOTE:

If you have purchased the ODBC feature to be installed by Lucent Technologies on your server prior to its delivery, then you may skip this procedure. Continue with client installation.

System Requirements

The following system requirements must be met in the client/server environment prior to installing the software:

- The ODBC server is on a standard *CentreVu* CMS hardware *Sun* platform with *Solaris* 2.4 or later running.
 - Any R3V6 CMS load will support ODBC.
 - The network communication software is correctly installed and configured, and the network is fully functional, so that the server can communicate with the clients.
-

Installation Procedure

Complete the following steps to install the server components of the *OpenLink* ODBC driver on the CMS server. Install the software to the `</usr/openlink>` directory on the CMS server.

1. Insert the Call Management System *OpenLink* ODBC Driver compact disk into the compact disk drive.
2. Log on to the CMS server as **root**.
3. Enter **cd /usr** to change directories to the `/usr` directory.
4. Enter **mkdir /usr/openlink** to create the *OpenLink UNIX*^{*} directory.
5. Enter **cd /openlink** to change directories to `/usr/openlink`.
6. Enter **cp /cdrom/cdrom0/server/* .** to copy the server components from the compact disk.

*UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

7. Enter **./install.sh** to run the *OpenLink* server side installation script.
8. At the prompt, enter **root** as the name of the user who will own the programs.
9. At the prompt, enter **root** as the name of the group that will own the programs.
10. Enter **/cms/dc/odbc/odbc_init** to configure and initiate the ODBC software on the server.
11. Enter **ps -ef | grep oplrqb** to verify that the ODBC Request Broker is active on the server. You should see an *oplrqb* process running.

At this point, the software is registered, installed, and running. If you do not see an *oplrqb* process running after completing step 11, check that the software was correctly installed.

Server Utilities

Introduction

Once the ODBC driver is installed on the server, a CMS utility program is available for setting the debug and log levels, and running and deactivating the ODBC feature by using the run feature. This utility program is located in the */cms/dc/odbc* directory.

Debug Levels

To set the server ODBC debug level, complete the following steps:

1. Enter **cd /cms/dc/odbc** to change directories to */cms/dc/odbc*.
 2. Enter **./odbc_init -d 0** or **1** to run *./odbc_init -d* (**0** turns the debug utility off, and **1** turns the debug utility on).
-

Log Levels

To set the CMS server log level for ODBC, complete the following steps:

1. Enter **cd /cms/dc/odbc** to change directories to */cms/dc/odbc*.
 2. Enter **./odbc_init -l 0-7** to run *./odbc_init -l* (where **0** turns CMS server logging off, and **1-7** sets the log level, with **1** as the lowest level and **7** as the highest [**5** is the default]).
-

Run ODBC

To run or deactivate the ODBC feature on the server, complete the following steps:

1. Enter **cd /cms/dc/odbc** to change directories to */cms/dc/odbc*.
2. Enter **./odbc_init -r 0** or **1** to run *./odbc_init -r* (where **0** deactivates the ODBC feature and **1** runs it).

Solaris Client Installation

Introduction

The *OpenLink* ODBC driver software can be installed on a *Solaris* client using the installation procedure in this section. Once the software is installed, it must also be configured using the procedure in this section. For additional information on installing and configuring the ODBC driver, see the client on-line help file that is included on the Call Management System *OpenLink* ODBC Driver compact disk.

System Requirements

Before installing the *OpenLink* ODBC driver software on a *Solaris* client, verify that you have met the following list of requirements.

- *The client is communicating with the CMS server over the network:* Use the network protocol's *ping* utility to ensure that communication between the client and the CMS server is functional. Do not proceed if basic communications between the client and the server cannot be established. If the client cannot recognize the network, then the ODBC driver will not function properly.
 - The ODBC clients are appropriate *Sun* computers with the *Solaris* 2.4 (or later) operating systems installed.
 - To develop an application using the ODBC API, the complete *Solaris* distribution (which provides the libraries and headers) and the *SPARCCompiler** must be installed. The development environment on the *Sun* client must be a *Solaris* 2.4 (or later) operating system.
-

Installation Procedure

NOTE:

If the client is on the same machine as the server, then the client software was installed with the server software. If this is your situation, go to "Configuration Procedure" in this section.

If you are installing a *Solaris* client on a machine other than the CMS server, continue with the following installation procedure. To install *OpenLink* ODBC software on the *Solaris* client:

1. Insert the Call Management System *OpenLink* ODBC Driver compact disk into the compact disk drive.

*SPARCCompiler is a trademark of SPARC International, Inc. and is licensed exclusively to Sun Microsystems, Inc.

2. Log in as **root**.
3. Enter **cd /usr** to change directories to the */usr* directory.
4. Enter **mkdir /usr/openlink** to create the */usr/openlink* directory.
5. Enter **cd /openlink** to change directories to */usr/openlink*.
6. Enter **cp /cdrom/cdrom0/client/solaris/* .** to copy the Solaris client components to */usr/openlink*.
7. Enter **./install.sh** to run the Solaris client installation script.
8. At the prompt, enter **root** as the name of the user who will own the programs.
9. At the prompt, enter **root** as the name of the group that will own the programs.

The libraries and other files needed for ODBC application development are now installed. A test application to verify client/server connectivity is included. See “Test ODBC Connectivity” in this chapter for more information.

Configuration Procedure

The *cms_odbc.ini* file located in */usr/openlink* must be modified and placed in the *\$HOME* directory as *.odbc.ini* for each user that will initiate the client application. First edit the *cms_odbc.ini* file, then copy it to *\$HOME/.odbc.ini*. The *cms_odbc.ini* file resides in */usr/openlink*.

The host parameter in the *Informix 7* connection segment must be changed. The *Informix 7* segment will be edited. The modified *Informix 7* section should look like the following example, with your default host name entered in the Host parameter field:

```
[Informix7]
Driver       = /usr/openlink/client/odbcsdk/lib/oplodbc.so
Host        = YOUR DEFAULT HOST NAME
ServerType  = Informix7
ServerOptions =
Database    = /cms/db/inf/cms
Options     =
ReadOnly    =
FetchBufferSize = 60
```

CAUTION:

To correctly resolve database connectivity, the **.odbc.ini** file must exist in the **\$HOME** directory of the initiating user on the ODBC client. The client application will fail if it is initiated by a user who does not have an **.odbc.ini** file. The following steps complete the client administration:

1. Enter **cd /usr/openlink** to change to the *OpenLink* directory.
2. Edit the host parameter in the *Informix 7* section of the **cms_odbc.ini** file (as described on the previous page).
3. Copy the **cms_odbc.ini** file to **\$HOME/.odbc.ini**.
4. Enter **cd \$HOME** to change to your home directory.
5. Enter **vi .profile** to edit your *.profile* file.
6. Enter **export LD_LIBRARY_PATH = /usr/openlink/odbcsdk/lib:\$LD_LIBRARY_PATH** to export *LD_LIBRARY_PATH*.
7. Enter **export UDBCINI = \$HOME/.odbc.ini** to export *UDBCINI*.
8. Enter **.wq** to exit your *.profile* edit.
9. Enter **. .profile** to source your *.profile*.

Once the *Solaris* client software is installed and configured, you can proceed with writing your application using the ODBC API.

Test ODBC Connectivity

On the *Solaris* client, initiate the test application to verify connectivity with the following procedure:

1. Enter **cd /usr/openlink/odbcsdk/examples** to change directories to */usr/openlink/odbcsdk/examples*.
2. Enter **./odbctest** to run *odbctest*.
3. At the “Enter ODBC connect string” prompt, enter:
DSN = Informix7; UID = Valid _UNIX_User_on_Server;
PWD = User_password to set these parameters (DSN = data source name, UID = user ID, and PWD = password).
4. At *SQL >*, enter a valid SQL query, for example:
select SPLIT, ACDCALLS from hsplit where ROW_DATE = “6/27/98” and STARTTIME = 1600 to test data connection.
5. Enter **exit** to exit.

If an error occurs, an error message is reported and the test application closes.

Possible Failure Causes

A test application error could result from one of the following causes:

- The driver is not active on the *CentreVu* CMS server. Verify that the ODBC driver is active on the server by entering **ps -ef | grep oplrqb**. You should see an *oplrqb* process running.
- There was an error in the *ODBC.connect* string entry. Resolve this

by re-entering the correct DSN, host, user ID and password at the *ODBC.connect* string prompt.

- The *.odbc.ini* file does not reside in the initiator's *\$HOME* directory.
- The *.odbc.ini* file is incorrect. Verify that the entered host is correct the database is */cms/db/inf/cms*.
- The *LD_LIBRARY_PATH* is not set. Enter **echo \$LD_LIBRARY_PATH** to ensure that *LD_LIBRARY_PATH* is included in the display (see "Test ODBC Connectivity" for more information).
- The *UDBCINI* is not set. Enter **echo \$UDBCINI** to ensure that *\$HOME/.odbc.ini* is included in the display. See "Configuration Procedure" in this section for more information.
- The data source is not consistent with the *cmsrqb_init* entry. Verify the data source you have entered (*Informix 7*).

Review the */usr/openlink/client/doc/STARTUP.DOC* file for other possible errors.

Build an ODBC Application

Provided that the development environment is installed, all necessary components to build an ODBC application exist in the */usr/openlink/client/odbcsdk* directory. Necessary headers and library routines reside in the respective */include* and */lib* directories. The actual application test code and associated *makefile* reside in */usr/openlink/client/odbcsdk/examples*. This code illustrates the use of the ODBC API. Copy the code to another directory prior to your use and development.

The ODBC application is responsible for implementing the embedded ODBC code provided by the ODBC libraries linked to the driver manager for host/database connectivity, data access, and any other necessary processing. If the test connection is successful ("Test ODBC Connectivity" in this section), the client has been properly configured and the ODBC header and libraries have been verified to be fully accessible to an application. Lucent Technologies cannot provide support for client-generated applications, or other third-party software or related mapping. See the caution in this chapter's overview, or contact your Lucent Technologies account representative.

Desktop Computer Client Installation

Introduction

The *OpenLink* ODBC Request Agent software can be installed on your desktop computer or on your network for access by each client using the procedures in this section. Once the software is installed, it must also be configured using the procedure in this section. For additional information on installing and configuring the ODBC driver, see the on-line help file that is included on the Call Management System *OpenLink* ODBC Driver compact disk.

System Requirements

Before installing the *OpenLink* ODBC driver software on your computer, verify that you have met the following list of requirements.

- *Client network software is installed:* Ensure that you have a “winsock”-compliant desktop TCP/IP product installed. Check for the existence of the *winsock.dll* file with the *Windows* Explorer function or through your file manager.
 - *Your computer is communicating with the CMS server over the network:* Use your desktop TCP/IP product’s Packet Internet Groper (PING) utility to ensure that communication between your computer and the CMS server is functional. Do not proceed if basic communications between your computer and the server cannot be established.
 - You are running *Microsoft Windows 95*, *Windows NT 3.51*, or *Windows NT 4.0*.
-

Installation

Use the following procedures to install *OpenLink* ODBC client software.

Windows 95, ***Windows NT 4.0***

For *Windows 95* or *Windows NT 4.0*, complete the following steps:

1. Start *Microsoft Windows*.
2. Insert the Call Management System *OpenLink* ODBC driver compact disk into the compact disk drive.
3. Open the *Windows* Explorer.
4. Select your compact disk drive.
5. Select the *Client* folder.

6. Select the *Win32* folder.
7. Double click on the *setup.exe* file to install the client software.
8. Follow the instructions on your screen.

 **NOTE:**

It is recommended that you use the *Normal* installation option.

9. Configure your ODBC data sources using the ODBC Administrator utility resident within your *Windows* Control Panel and the procedures in this section.

Windows NT 3.51

For *Windows NT* 3.51, complete the following steps:

1. Start *Microsoft Windows*.
2. Insert the Call Management System *OpenLink* ODBC Driver compact disk into the compact disk drive.
3. From the *Windows* Program Manager, choose **File (Alt+F)**, and then the **Run** command.
4. Type the compact disk drive letter, followed by **:*client*\Win32\setup**. For example, if your compact disk drive is drive *a*, then type **a:*client*\Win32\setup**.
5. Follow the instructions on your screen. It is recommended that you use the *Normal* installation option.
6. Configure your ODBC data sources using the ODBC Administrator utility resident within your *Windows* Control Panel and the procedures in this section.

Install Clients from a Single Network Point

To install ODBC on a desktop computer client that does not have a compact disk drive, or to install ODBC from a single network point, complete the following steps:

1. Install the ODBC driver on a networked machine with a compact disk drive, using the procedure for the machine's operating system (*Windows 95*, *Windows NT 4.0* or *Windows NT 3.51*) in this section.
2. Open your computer's program manager or *Windows Explorer* and go to the directory for the networked machine.
3. **Copy** the *OpenLink* folder from the networked machine.
4. Change directories to your client computer's hard drive.
5. **Paste** the *OpenLink* folder to your hard drive.
6. Run the *setup.exe* file using the *Normal* option and configure your ODBC data sources using the procedures in this section.

Desktop Computer Client Configuration

Introduction

The procedures on the following pages use the *OpenLink* ODBC Driver Administration facility to administer data sources for the client(s).

⇒ NOTE:

There are slight variations between the displayed ODBC dialog boxes for *Windows 95*, *Windows NT 3.51*, and *Windows NT 4.0* and *Windows NT 4.0* if the underlying *Microsoft* ODBC driver is version 3.0. If your underlying *Microsoft* ODBC driver is version 3.0, and you are running *Windows NT 4.0*, the ODBC Data Source dialog box displays with function tabs across the top instead of function buttons on the face of the dialog box. Throughout the configuration procedures, any exceptions or differences for these dialog boxes are given in parentheses and are discussed in those cases where the differences affect the procedure.

ODBC Driver Administration Utility

The ODBC Driver Administration utility is a standard utility that resides within your desktop environment's control panel. This utility is responsible for adding and removing ODBC drivers. To access the ODBC Driver Administration utility after the *OpenLink* ODBC driver software has been installed, complete the following steps:

For *Windows 95* or *Windows NT 4.0*, complete the following steps:

1. Select the **Start** button in the taskbar.
2. Select **Settings**.
3. Select **Control Panel**.
4. Select **32 Bit ODBC**.

For *Windows NT 3.51*, complete the following steps:

1. Select **Main** from the **Program Manager**.
2. Select **Control Panel**.
3. Select **32 bit ODBC**.

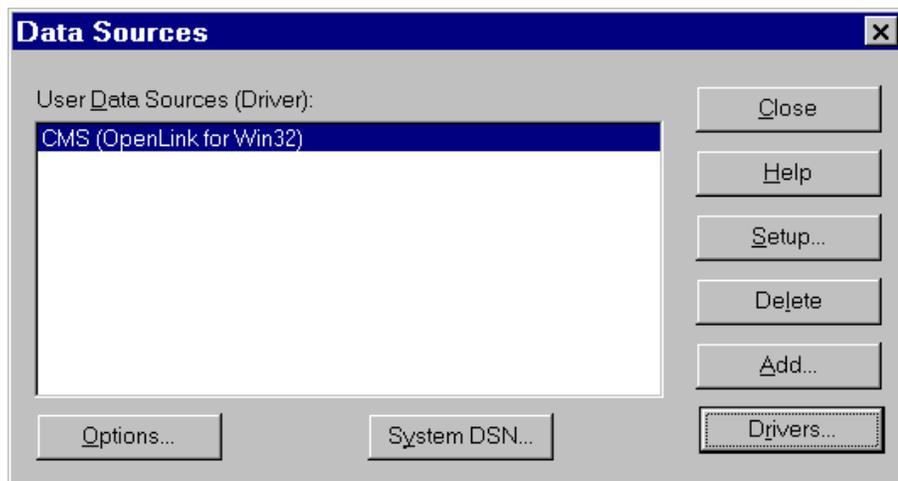
ODBC Data Source Dialog Box

The ODBC Data Sources dialog box (versions without and with an underlying version 3.0 ODBC driver are shown on the following page) displays, with available ODBC data sources listed. To access the CMS server, a data source must be administered, after it has been added and configured. See "Define Data Sources" in this section for more information. The ODBC Data Source dialog box and the Add Data

Sources dialog box (not used with the version 3.0 driver) allow you to select an ODBC driver for which you want to add a data source. In this case, you will add the *CMS (OpenLink for Win32)* driver. You will configure data sources for the driver using the *OpenLink Setup* dialog box.

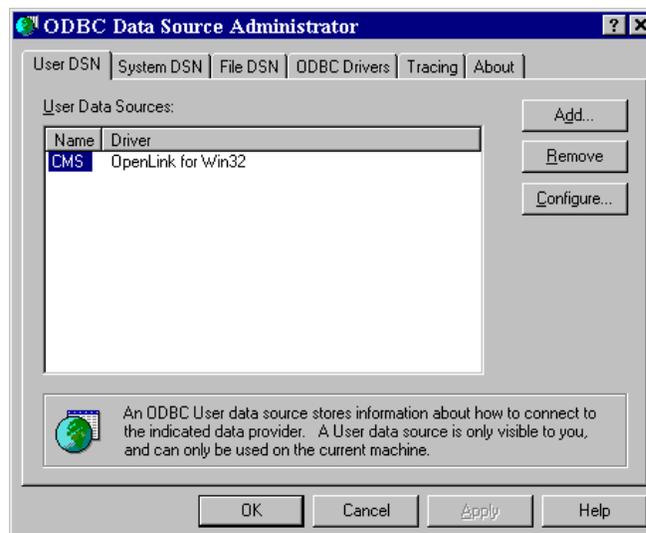
Data Sources Dialog Box—Without Version 3.0 ODBC Driver

The following figure shows an *OpenLink Data Sources* dialog box as it displays without an underlying version 3.0 ODBC driver.



Data Sources Dialog Box—With Version 3.0 ODBC Driver

The following figure shows an *OpenLink Data Sources* dialog box as it displays with an underlying version 3.0 ODBC driver.



Define Data Sources

You may define one or more data sources for each installed driver. The data source name should provide a unique description of the data; for example, Call Management System. The defined data sources for all of the currently installed drivers are listed in the User Data Sources (Driver) list on the Data Sources dialog box.

ODBC Data Source Fields

The following table defines the ODBC Data Source dialog box fields (and alternate names and tabs for the version 3.0 driver).

Field	Description
Close	Closes the dialog box and exits the ODBC Control Panel.
Help	Accesses the ODBC on-line Help file.
Setup (Configure)	Allows you to configure an existing data source. You should select the name of the data source you want to configure before selecting the Setup button.
Delete (Remove)	Removes an existing data source. You should select the name of the data source you want to delete from the list before selecting the Delete button.
Add	Adds a new data source. If you select this button, a dialog box prompts you for the driver for which you are adding a data source. After you select a driver, a driver-specific setup dialog box is displayed. If you select this button, a dialog box prompts you for the information required to set up a login ID-based data source. This is not recommended for CMS.
Drivers (ODBC Drivers tab)	Displays information about an ODBC driver. If you select this button, a dialog box displays a list of the types of drivers currently installed on your computer.

Field	Description
System DSN (tab with version 3.0 driver)	Allows you to add, delete, or configure data sources local to a computer, rather than dedicated to a user. If you select this button, a dialog box prompts you for the information required to set up the system data source. This is recommended for CMS.
Options (Tracing tab)	Allows you to set ODBC options. If you select this button, a dialog box prompts you, asking if the ODBC calls should be traced; and, if so, the name of the trace file.
User DSN (version 3.0 driver only)	Allows you to add, delete, or configure data sources that are dedicated to a specific user login ID. If you select this tab, the User DSN dialog box prompts you for the information required to set up a login ID-specific data source. This is not recommended for CMS, but is presented as an option in this document.
File DSN (version 3.0 driver only)	Allows you to add, delete, or configure file-based data sources in a shared folder that is accessible by every client on the network. If you select this tab, the File DSN dialog box and accompanying wizard prompts you for the information required to set up a login ID-specific data source. This is not recommended for CMS and is not described in this document; however, you may choose this option if you have a large number of networked <i>Windows NT</i> 4.0 clients.
About (version 3.0 driver only)	Displays a dialog box with detailed information about the ODBC drivers available on your system.

Add System Data Sources

Before adding a data source, you should decide if you want to add a system-wide data source or a data source specific to a user login ID. Administering data sources on a per-user login ID basis is an optional procedure. Use the following procedure to administer a system data

source. Follow the procedure in “Add Login ID-Based Data Sources” in this section to administer data sources on a per-login ID basis.

 **NOTE:**

It is recommended that you add system-wide data sources for *CentreVu* CMS, rather than having to administer data sources on a per-login ID basis.

Add System Data Sources Procedure

To add a system data source, complete the following steps:

1. On the ODBC Data Source dialog box, select the **System DSN...** button (or tab). The System Data Sources dialog box displays.
2. Select the **Add...** button. The Add Data Source dialog box displays.

If you are using the *Windows* version 3.0 ODBC driver, the Create New Data Source Wizard displays.

3. On the Add Data Source dialog box (or wizard), select *OpenLink for Win32*.
4. Select the **OK** button (or **Finish** button) to select the *OpenLink* generic ODBC driver.

Once you have selected the *OpenLink* ODBC driver, the *OpenLink* Setup dialog box displays. See “Configure a New ODBC Data Source” in this section for data source setup procedures.

Add Login ID-Based Data Sources

As an option, you may administer data sources on a per-user login ID basis. This procedure is useful if you are providing access for a specific user; however, it is strongly recommended that you administer system data sources. To administer login ID-based data sources, complete the following steps:

1. On the ODBC Data Source dialog box, select the **Add...** button. The Add Data Source dialog box displays.

If you are using the version 3.0 driver, select the User DSN tab on the ODBC Data Source dialog box. The User DSN dialog box displays.

2. Select the generic *OpenLink* ODBC driver, *OpenLink for Win32*.
3. Select the **OK** button to select the *OpenLink* generic ODBC driver.

If you are using the version 3.0 driver, select the **Add...** button on the User DSN dialog box. The Create New Data Source wizard displays.

4. For the version 3.0 driver only, select the *OpenLink for Win32* driver and select the **Finish** button.

Once you have selected the *OpenLink* ODBC driver, the *OpenLink* Setup dialog box displays. See “Configure a New ODBC Data Source” in this section for data source setup procedures.

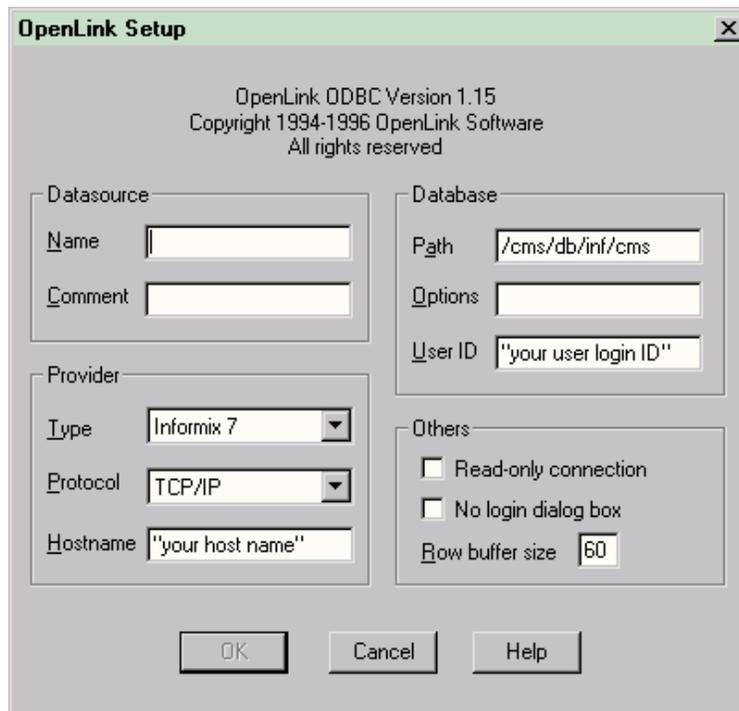
If you are using the version 3.0 driver, select the **Configure** button on the User DSN dialog box to configure the data source. See “Configure a New ODBC Data Source” in this section for data source setup procedures.

Remove a Data Source

To remove any data source, select the **Delete** button (or **Remove** button) on the appropriate ODBC Data Source dialog box and follow the prompts.

Configure a New ODBC Data Source

The CMS data source is now associated with the *OpenLink for Win32* driver. Specific server and database access must now be configured through the generic *OpenLink* Setup dialog box, shown in the following figure.



Configure a New ODBC Data Source Procedure

To configure your ODBC driver software to access CMS data, complete the following steps:

1. In the **Datasource: Name** field, enter a descriptive name for your data source. In a multi-site call center situation, you can use this field to differentiate between call center locations.
2. In the **Datasource: Comment** field, you may optionally enter “Call Management System” (or other information related to the CMS data source).
3. In the **Provider: Type** field, enter *Informix 7* or select *Informix 7* from the pull-down list.
4. In the **Provider: Protocol** field, enter **TCP/IP**, or select **TCP/IP** from the pull-down list.
5. In the **Provider: Hostname** field, enter the name of your database host machine (CMS server).
6. In the **Database: Path** field, enter **/cms/db/inf/cms**.
7. Leave the **Database: Options** field blank.
8. In the **Database: User ID** field, you may optionally enter a valid CMS user login ID. Entering a CMS user login ID displays a default user name for each login to the data source.
9. *Do not* select the **Read-only connection** check box in the **Others** section. This allows complete ODBC access to customer-defined database tables (**c_*** tables).
10. *Do not* select the **No login dialog box** check box in the **Others** section. A login dialog box displays each time you connect to the *CentreVu* CMS database.

NOTE:

You may select this check box to prevent a login dialog box from displaying with each database access; however, this is recommended only if the software through which you are accessing the database, such as *Crystal Reports*, is password-enabled.

11. In the **Others: Row Buffer Size** field, enter a row buffer size of **60**.
12. Select the **OK** button, or select the **Cancel** button to cancel the configuration.

At this point, the ODBC driver software is installed on your computer. For further configuration options, you may continue with “Administer ODBC Options” in this section.

Once the driver is configured, the *OpenLink* ODBC driver is accessible to ODBC-enabled applications on your computer. Any queries that you send

to the CMS database from client *Windows* applications, such as *Microsoft Access*, will use the ODBC feature to access data and copy it to your applications.

 **NOTE:**

You will need to format the data within your application. The data returned from your SQL queries will be formatted in the manner described in Chapter 3, “Database Tables.”

OpenLink Setup Input Fields

The following table describes the input fields on the *OpenLink* Setup dialog box.

Field	Description
Datasource	
Name	A representative name for the server/database you are connecting to, such as CMS.
Comment	A description of the data source you are connecting to, such as CMS.
Provider	
Type	The database type or user-defined logical representation of the underlying DBMS you are establishing an ODBC session with. The <i>OpenLink</i> ODBC driver allows the creation of custom providers, which enables you to mask the identity of the underlying database to which connections are made. This provides ODBC-based client/server infrastructure management flexibility.
Protocol	The network protocol you are using to access database data items; the default is TCP/IP.
Hostname	The network alias (or IP address) of the machine on your network hosting the database(s) to which you want to make ODBC connections.
Database	
Path	The field used to map the ODBC data source names to specific database names for a provider type. For example, the CMS <i>Informix</i> database has a path field value of /cms/db/inf/cms , identifying the CMS database to be associated with an ODBC data source name.
Options	An optional entry field used for database connections after the database environment has been initialized. These settings generally take the form of operating environment variables which cannot be entered in this field. An entry is not required or recommended in this field.

Field	Description
User ID	An optional entry field that contains a valid CMS user login ID for the database being represented by the provider type.
Others	
Read-only connection	A check box that allows you to choose if you do or do not want to enforce read-only database sessions for ODBC-compliant applications. This box should not be checked if you are accessing customer data, and should be checked if you are not accessing customer data.
No login dialog box	A check box that allows you to choose if you do or do not want to disable the pop-up login dialog box during ODBC session establishment with a selected ODBC data source.
Row buffer size	A variable-entry field that determines at ODBC run time the size of the <i>OpenLink</i> network array fetch buffer (the number of records transported from an <i>OpenLink</i> database agent to and ODBC-compliant application environment during a single network hop). This setting affects the performance of the <i>OpenLink</i> driver. The recommended setting for the CMS database is 60.

Administer ODBC Options

You may specify how the ODBC driver traces ODBC function calls. If tracing is activated, a file is generated that contains the actual ODBC function calls. To set these ODBC options, complete the following steps:

1. On the ODBC Data Source dialog box, select the **Options...** button. The ODBC Options dialog box displays.
If you are using the version 3.0 ODBC driver, select the Tracing tab. The Tracing dialog box displays.
2. To trace ODBC calls or observe ODBC activity, select the **Trace ODBC Calls** check box (or the **When to Trace: One time only** check box).

⚠ CAUTION:

If you select the **Trace ODBC Calls** check box, you must also select the **Stop Tracing Calls Automatically** check box to avoid compromising the client system performance.

3. To stop tracing ODBC function calls automatically, select the **Stop Tracing Calls Automatically** check box. This will terminate the ODBC tracing upon completion of the ODBC session.

⇒ NOTE:

To start tracing ODBC function calls again, you must manually reselect the **Trace ODBC Calls** check box (or the **When to Trace: One time only** check box) and restart your application.

4. To select or change the file to which the *OpenLink* Request Broker writes tracing information, select the **Select File...** button (or **Browse...** button). The default trace file is `\SQL.LOG`. The Select ODBC Trace File dialog box displays.
5. Select the file to which you want the *OpenLink* Request Broker to write tracing information.
6. Select the **Open** button.
7. You will automatically return to the ODBC Options dialog box. The file you have chosen displays in the **Trace File:** field (or **Logfile Path** field).
8. To view the ODBC trace file, open the `SQL.LOG` file (or the file that you have designated as the trace file).

View Installed ODBC Drivers

To view a list of installed ODBC drivers, complete the following steps. This is useful for verifying if the ODBC driver has been properly installed.

1. On the ODBC Data Source dialog box, select the **Drivers...** button (or tab). The Drivers dialog box displays.

⇒ NOTE:

Use the Drivers dialog box to verify installation of the *OpenLink for Win 32* driver. If the *OpenLink for Win32* driver does not display on the Drivers dialog box, return to “Desktop Computer Client Installation” in this chapter and reinstall the drivers.

2. To view detailed information about an installed driver, select the driver from the list and select the **About** button (or tab). The About dialog box displays.

3. When you have finished viewing information about the driver, select the **OK** button to exit the About dialog box.
4. Select the **Close** button (or **OK** button) to exit the Drivers dialog box. The ODBC Data Source dialog box displays again.

 **NOTE:**

If you are using the *Microsoft* version 3.0 ODBC driver (dialog box with tabs), the ODBC Driver Administration utility will close when you exit directly from any dialog box. You will need to reopen the ODBC Data Source dialog box using the procedure in “ODBC Driver Administration Utility” in this section.

Test Desktop Computer Connectivity

Introduction

Once you have installed the *OpenLink* ODBC driver software on both the client and the server, you may open a demonstration connection to a data source on the server to show connectivity and test SQL access. To connect to an administered data source, use the following procedure.

Connect to and Access Data

To connect to a data source from the client, complete the following steps:

1. Select **OpenLink** or **Applications** on your computer.
2. Select **C++ Demo 32 bit**. The ODBC SDK C++ Demo window displays.
3. From the **Environment** menu, select **Open Connection**. The SQL Data Sources dialog box displays.

 **NOTE:**

Depending on the ODBC driver version you are using, the Select Data Source dialog box may display and you will be prompted to select a file or machine data source. Select the Machine Data Source tab and select the CMS data source.

4. Select the data source that you want to use from the list or enter it in the **Select Data Source** field. The *OpenLink* ODBC login dialog box displays. This dialog box only displays if you did not check the **No Login Dialog Box** check box on the Setup dialog box.

 **NOTE:**

You should have already configured the CMS data source. If you do not see a data source on the SQL Data Sources dialog box, see “Desktop Computer Client Configuration” in this chapter.

5. Enter your CMS server user name and password (CMS server login).
6. Select **OK**. You will return to the ODBC SDK C++ Demo screen.
7. Select **SQL** to begin writing your SQL query.
8. Select **Execute SQL**. The ODBC SDK 2.0 C++ Demo SQL dialog box displays.
9. Enter the following SQL query in the text box: **select count (*) from hsplit**. The queried data displays in the ODBC SDK C++ Demo window; however, any valid SQL query may be entered.

Result

You should see a count column on the ODBC SDK 2.0 C++ Demo window. The value in this column is the number of columns in the CMS `hsplit` table. This result confirms that you have successfully accessed the database from the client. Use this test as a troubleshooting tool in the future to verify connectivity and data access from the client.

Disconnect from a Data Source

Once you have completed test querying the database, you can disconnect from the data source through the ODBC driver software. To disconnect, complete the following steps:

1. From the **Environment** menu, select **Close Connection**. The ODBC SDK 2.0 C++ Demo Close Current Connection dialog box displays.
2. Select **OK** to close the connection, or **Cancel** to remain connected to the data source.

Troubleshooting

Introduction

General troubleshooting procedures and error messages on the *OpenLink* ODBC driver are included in this section. For more detailed information, see the on-line help file that is included on the Call Management System *OpenLink* ODBC Driver compact disk.

Server Log File

Once the ODBC driver is installed and initiated, the server log file, *odbc.log*, records the logging levels of all ODBC activities. The default log level is **5**. See the “Server Utilities” section in this chapter for more information. You may review the *odbc.log* file for information about an ODBC sessions. Archives of past ODBC sessions are maintained in the *odbc.log.01* and *odbc.log.02* files. These logging levels are set by running the *./odbc_init -l* file located in the */cms/dc/odbc* directory. The values for the *./odbc_init -l* are **0-7**, where **0** turns CMS server logging off, and **1-7** sets the logging level, with **1** as the lowest level and **7** as the highest.

Client Trace

The *OpenLink* client component configuration utility, located in the *OpenLink* group on your Windows desktop, allows you to enable or disable ODBC trace logging. Trace logging provides you with output on your entire ODBC session, including all ODBC calls made by the ODBC-compliant application you are using. See the “Administer ODBC Options” section in this chapter for complete information on this utility. This file also includes native database error messages that might not have been replaced by the ODBC-compliant application you were using.

OpenLink Error Messages

The error messages that you may receive from the *OpenLink* ODBC driver are detailed in the following list. These error messages are displayed on the client and in the server log file.

**Invalid
Username/Password**

This error message is displayed when the operating system-level username and password verification is in use, and when you enter an invalid operating system-level username and password combination when connecting to your *OpenLink* ODBC Data Source, even though the username and password combination entered is valid at the database level.

**Unable to Locate
Requested Service**

This error message is displayed as a result of the *OpenLink* Session Rules Book being incorrectly configured. It typically takes the form of an invalid reference to the *OpenLink* database agent executable program responsible for providing database access. This error will not occur with the default *cmsrqb_init* settings.

**Unable to Load
OpenLink Request
Agent**

This error message indicates a client machine problem at the network transport level, indicating the need to verify that your client machine's network software is correctly configured.

**Remote Procedure
Call (RPC) Unable to
Send**

This indicates a corruption of the communications channel being used by the *OpenLink* Request Agent. This error typically occurs when the server Request Broker has been shut down. Verify the status of the Request Broker, then try to reinitiate the ODBC session.

RPC Timed Out

This error message is displayed when timeout settings in either the client side or server side Session Rules Books have been exceeded. This message typically occurs when communication cannot be established from client to server or server to client. Verify that the *oplrqb* is running on the server side by entering **ps -ef | grep oplrqb**. You should see an *oplrqb* process running.

RPC Unable to Receive	This message indicates that the server Request Broker is no longer communicating with the client. This error occurs when the server Request Broker has been shut down or reinitiated during a session. Verify that the Request Broker is active on the server.
RPC Host Unknown	This message results from a network failure or invalid host entry in the Data Source Setup dialog box. Check the Hostname field entry in that window; if you suspect a network problem, verify this and correct it if necessary; or retry your ODBC session.
Unknown Database Agent Requested	This problem results when the client administration associated with the ODBC session does not resolve to the <i>oplrqb</i> rules in <i>cmsrqb_init</i> . To resolve this error, the Data Source Setup dialog box Type field should be set to <i>Informix 7</i> . Retry your ODBC session.
Unable to Start the Requested Database Agent	The <i>oplrqb</i> was able to resolve to a database agent, but was unable to execute the program. To resolve this, the Type field on the Data Source Setup dialog box should be set to <i>Informix 7</i> . Alternately, the <i>generic_inf7</i> should be set to <i>iinf7_sv</i> . Finally, the <i>inf7_sv</i> should exist in <i>/usr/openlink/bin/</i> as executable.
Database Errors	Database errors are displayed after failed database accesses. To resolve this, correct the database query and resubmit it.

Database Tables

Overview

This chapter defines the historical database items and tables populated by *CentreVu®* CMS that can be accessed through ODBC. The chapter is organized as follows:

- How Database Items are Presented
- Database Table Names
- Split/Skill Database Items
- Agent Database Items
- Trunk Group Database Items
- Trunk Database Items
- Vector Database Items
- Vector Directory Number (VDN) Database Items
- Call Work Codes Database Items
- Agent Login/Logout Database Items
- Agent Trace Database Items
- Current Day Configuration Forecast Database Items
- Current Day Forecast Report Database Items
- Call Record Database Items
- Data Collection Period Database Items
- Archiver Execution Status Database Items
- Customer Log Database Items
- Agent Group Database Items
- Synonyms Database Items
- Automatic Call Distribution (ACD) Shifts Database Items
- Database Items
- Exceptions Historical Database Items

How Database Items are Presented

Database Item Tables

The database items are presented in a table format, according to ACD entity (split/skill, agent, vector, VDN, trunk, trunk group, exception, synonym, call work code, and forecast).

Sample Database Item Table

The following table is an example of how the table information is presented:

Database Item	Description	Data Type	Column Type	Length
DATABASE ITEM	The definition of the database item is given here. Any additional information, such as other database items that are included in the sum of the database item, or the specific ECS release to which the database item applies, is also listed.	C, A, I, N, M, or B	integer, smallint, date, smallfloat, or char(n)	length in bytes

Many database items appear in more than one database table. When an item is in more than one table, the definition is probably not the same from table to table. Database items that are used in the description of another database item are in boldface type. Where applicable, Enterprise Communications Server (ECS) information has been included in the database item descriptions.

⇒ NOTE:

If you are using a G2.2 ECS without the Call Center Features (CCF = N), you should refer to information for the G2.1 ECS for functionality.

Index Database Items

The index database items in each table are marked. Indexes add structure to table rows so that *CentreVu* CMS can retrieve data faster. The row search criteria you define for custom reports should be based on indexes whenever possible. For historical custom reports, always include a “where” clause based on the **ROW_DATE** database item.

Data Types

Each database item contains one of the following types of data:

C = Cumulative data: accumulates throughout the collection interval. Most real-time database items contain cumulative data.

A = Administrative data: administered on the ECS or on *CentreVu* CMS. For example, the database item INTRVL in the split/skill table contains the number of minutes in the intrahour interval (15, 30, 60) assigned for the specified ACD on *CentreVu* CMS.

I = Row Identifier data: gives data that is common to all tables, such as time, date, split in the split/skill tables, and so on.

N = Special Table data: belongs only to a specific table, such as the Historical Agent Login/Logout table or Current Day Forecast table.

M = Maximum Interval Value data: gives data that is the maximum reached for any value in the specified interval.

B = Busy Hour data: gives data that is only meaningful for the busy hour.

The letter C, A, I, N, M or B appears in the Data Type column for each database item.

Cumulative, Administrative, Maximum Value, Row Identifier and Busy Hour data items apply to historical and real-time database items. **Special Table** data items apply only to historical database items.

Call-Based Data

In addition to the types of data described above, items in the *CentreVu* CMS database can be either call-based or interval-based. Most *CentreVu* CMS database items are call-based. **Call-based data** are committed to the database after a call completes. Therefore, if a call starts and ends in different collection intervals, all of the data are recorded in the interval in which the call and any After Call Work (ACW) are completed.

Interval-Based Data

Interval-based data represents the amount of time during a collection interval that is used for a particular activity. Interval-based items are updated throughout the collection interval and timing is restarted at the end of the interval. Most interval-based items start with **I_** or **TI_**. The database items **ALLINUSETIME** (trunk-group tables) and **MBUSYTIME** (trunk and trunk-group tables) are also interval-based. Each database item has a defined column type.

Column Types

The database column type indicates the format of the information within that database item. The column type definition includes either the length in bits of the database item or the *Informix* data type. The column types are included in the database item tables, along with the length or *Informix* data type. Column type and length information is included for the user's reference. Because data gathered through ODBC can be used in a variety of applications, it is helpful to know what type of data you are accessing and how long it is in bytes. Any difference or exception in the column type or length between tables in a table group; such as, between the `hsplit` and `msplit` tables in the `split/skill` table group, are indicated in the **Column Type** and **Length** columns.

Informix Column Types

The following table defines the data column types in the open database.

Column Type	Definition
<code>char(n)</code>	An ASCII string of <i>n</i> characters, 1 byte per character.
<code>date</code>	<i>Informix</i> date type, 4 bytes long. The <i>Informix</i> date format is yyyy-mm-dd. For example, May 19, 1998 would display as 1998-05-19.
<code>integer</code>	4 byte integer
<code>smallint</code>	2 byte integer
<code>smallfloat</code>	<i>Informix</i> floating point numerical type, 4 bytes long. The <i>Informix</i> smallfloat format is a decimal type used for percentages, and includes a comma and a plus/minus sign.

ECS Cross-Reference Tables

CentreVu CMS database items apply to specific ECS releases. Following each database item table is an ECS cross-reference table. The ECS cross-reference tables list each database item by ECS release.

Sample ECS Cross-Reference Table

The following table is an example of how the table information is presented:

⇒ NOTE:

If you are using the G2.2 ECS without the Call Center Features (CCF = N), you should refer to the G2.1 column for functionality

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY® ECS R5	DEFINITY ECS R6
DATABASE ITEM	X	X	X	X	X	X	X
	ECS releases that this database item applies to are marked with X's.						

Database Logic Structure

CMS historical tables store information in one record per row format. This formatting affects the way data can be accessed through ODBC. For example, in the agent tables, a record will be created for each split/skill that an agent is logged into. If an agent is logged into four splits/skills, there will be four records for that agent. Similarly, if an agent starts the day with four splits/skills, and is added to a fifth split/skill before the end of the day, the agent's fifth record will be generated only from the point at which the additional split/skill was added. The other four records will reflect the total logon time. When accessing data in the historical tables, you may need to sum the information to retrieve complete data.

Agent Tables

If an agent logged off and logged on more than once in a specified interval, another complete set of records will be created for that agent for each logon in the agent tables. If an agent logs into four split/skills, logs out, and then logs back on during a set interval, there will be two sets of four records for that agent, one set per logon.

VDN Tables

The VDN tables store one record per vector on which a VDN terminates. Therefore, if the terminating vector for a specific VDN changes in a set interval, there will be two records for that VDN—one per terminating vector. This logic also applies to the Vector, Trunk, Trunk Group, and Split/Skill tables. If information is required from these tables, a sum structured query language (SQL) query may be necessary to access complete data for each VDN, vector, trunk, and so on.

Circular Structure

The exceptions, call record, and agent trace tables are circular files. These tables will continuously populate until the table capacity plus ten percent has been reached; at which point, the oldest ten percent of the records will be deleted. For example, if an agent trace table has a capacity of 100 rows, and the total rows populated equals 110, the oldest ten rows will automatically be deleted. Therefore, the data in that table will change continuously as the table is updated.

Terminology

For definitions of terms used in database item descriptions, see Appendix A, "Database Items and Calculations," the Glossary in the *CentreVu® Call Management System R3V6 Reports User Guide* Lucent Technologies (585-215-851) document, and the Glossary in this document.

CMS Feature Interactions

- **Dictionary Names:** Clients can access CMS Dictionary names. The client must map the synonym to the report.
 - **Permissions:** Applications which access *Informix* externally, such as database access scripts, may not work if the table permission script tries to access a table to which permission is denied.
 - **Field Display:** The time and date data you receive from the database may not be formatted. Generally, times may be shown in seconds or in military format. You will need to review data for formatting when you import it into your software application. The data returned from your SQL queries will be formatted in the manner described in your database interface specifications. See your software's documentation for further information on formatting data.
-

Interactions with ECS Features and Tracking of ECS Capabilities

Some of the ECS features and capabilities have an impact on *CentreVu* CMS open database items. For more information on these features and capabilities and their relationship with the *CentreVu* CMS open database, see Appendix A, "Database Items and Calculations" in the *CentreVu® Call Management System R3V6 Reports User Guide* Lucent Technologies (585-215-851) document.

Database Table Names

To select data for external use, you must use the names listed in the following table in your queries. The following table contains all the tables that are ODBC-accessible in *CentreVu* CMS and a brief description of the data in each. If a table is not listed here, it is not accessible through ODBC, but will still be accessible by **root**. All tables included in this chapter have read-only external user access. The database items are described later in this chapter.

Name	Data Stored
hsplit	Split/Skill data for each intrahour interval.
dsplit	Split/Skill data summarized by day.
wsplit	Split/Skill data summarized by week.
msplit	Split/Skill data summarized by month.
hagent	Agent data for each intrahour interval.
dagent	Agent data summarized by day.
wagent	Agent data summarized by week.
magent	Agent data summarized by month.
htkgrp	Trunk group data for each intrahour interval.
dtkgrp	Trunk group data summarized by day.
wtkgrp	Trunk group data summarized by week.
mtkgrp	Trunk group data summarized by month.
htrunk	Trunk data for intrahour interval.
dtrunk	Trunk data summarized by day.
wtrunk	Trunk data summarized by week.
mtrunk	Trunk data summarized by month.
hvector	Vector data for each intrahour interval.
dvector	Vector data summarized by day.
wvector	Vector data summarized by week.
mvector	Vector data summarized by month.
hvdn	VDN data for each intrahour interval.
dvdn	VDN data summarized by day.

Database Table Names

Name	Data Stored
wvdn	VDN data summarized by week.
mvdn	VDN data summarized by month.
hcwc	Call work code data for each intrahour interval.
dcwc	Call work code data summarized by day.
wcwc	Call work code data summarized by week.
mcwc	Call work code data summarized by month.
call_rec	Call record data.
d_secs	The number of seconds in the daily data collection period.
m_secs	The number of seconds in the monthly data collection period.
w_secs	The number of seconds in the weekly data collection period.
arch_stat	The status of archiver executions.
customer_log	The customer error log data.
agroups	Agent group definitions.
synonyms	Dictionary synonyms.
acd_shifts	Shift times and maximum agents logged in for each shift.
dbitems	Dictionary standard and custom database items, constants, and calculations.
f_cday	Forecast current day configuration data by split/skill.
f_cdayrep	Current day forecast data by split/skill.
haglog	Agent login and logout information.
ag_actv	Agent activity trace data.
agex	Agent exceptions.
fullex	Disk full exceptions.
spex	Split/skill exceptions.
tgex	Trunk group exceptions.
vecex	Vector exceptions.
vdnex	VDN exceptions.
linkex	Link down exceptions.
mctex	Malicious call trace exceptions.

Split/Skill Database Items

Overview

The Split/Skill database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical split/skill database items apply to the Intrahour Split/Skill (`hsplit`), Daily Split/Skill (`dsplit`), Weekly Split/Skill (`wsplit`), and Monthly Split/Skill (`msplit`) tables. All items listed in the following table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four split/skill tables are also noted in the Column Type column. The historical indexes are **SPLIT** and **ROW_DATE**.

Split/Skill Database Items

The following table describes the data items in the Split/Skill CMS database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	Number of CALLSOFFERED that were abandoned while in queue or ringing at an agent position. For Generic 3 Version 1 switches with the Vectoring feature, this also includes calls that were queued to the split/skill and abandoned while listening to a forced disconnect announcement.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
ABNCALLS (contd)	<p>NOTE: When a call abandons while queued to multiple splits/skills and abandons from queue, only the primary split/skill increments ABNCALLS (calls that are ringing an agent and then abandon peg as abandons for the split/skill they were ringing). This also includes calls with talk times less than the phantom-abandoned call timer value, if it is set.</p> <p>$ABNCALLS = ABNCALLS1 + ABNCALLS2 + ABNCALLS3 + ABNCALLS4 + ABNCALLS5 + ABNCALLS6 + ABNCALLS7 + ABNCALLS8 + ABNCALLS9 + ABNCALLS10$</p> <p>ABNCALLS includes ABNCALLS1-10, ABNRINGCALLS, O_ABNCALLS, PHANTOMABNS, and SLVLABNS.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ABNCALLS1-10	<p>Number of ABNCALLS that were abandoned during the collection interval in each of the service level increments PERIOD1 through PERIOD9 (as defined on the ACD Administration: Call Profile window). ABNCALLS10 counts calls that abandoned after PERIOD9.</p> <p>NOTE: If call profiles are not set, then the data gets stored into the first interval (ABNCALLS1).</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ABNRINGCALLS	<p>Number of split/skill or direct agent ABNCALLS that abandoned while ringing at an agent position. Available for ring tracking with Generic 2 and Generic 3 switches.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ABNTIME	<p>Time callers spent waiting in queue and ringing at an agent's voice terminal before abandoning the call. For phantom abandons, ABNTIME includes the time until the agent releases the call.</p>	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
ACCEPTABLE	Number of ACDCALLS answered by an agent within the predefined acceptable service level (SERVICELEVEL), as defined on the ACD Administration: Split/Skill Call Profile window.	C	integer	4 bytes
ACD (index)	The ACD number for which data was collected.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACDAUXOUTCALLS	Number of AUXOUTCALLS agents in the split/skill made with at least one split/skill ACD call for this split/skill on hold. For agents in multiple skills with multiple call handling (Generic 3 Version 3 switch and later), the call is recorded for the skill of the last ACD call the agent put on hold. ACDAUXOUTCALLS includes calls made to transfer or conference the ACD call. Available with Generic 2.2 and Generic 3 switches.	A	smallint	2 bytes
ACDCALLS	Number of CALLSOFFERED calls that were answered by an agent in the split/skill. $ACDCALLS = ACDCALLS1 + ACDCALLS2 + ACDCALLS3 + ACDCALLS4 + ACDCALLS5 + ACDCALLS6 + ACDCALLS7 + ACDCALLS8 + ACDCALLS9 + ACDCALLS10$. ACDCALLS includes ACCEPTABLE , ACDCALLS1-10 , BACKUPCALLS , CONFERENCE , HIGHCALLS , HOLDCALLS , LOWCALLS , MEDCALLS , O_ACDCALLS , TOPCALLS , and TRANSFERRED .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
ACDCALLS1-10	Number of ACDCALLS during the collection interval that were answered in each of the service level increments PERIOD1 through PERIOD9 (as defined on the ACD Administration: Call Profile window). ACDCALLS10 is the number of calls answered after the last increment PERIOD9 . NOTE: If call profiles are not set, then the data gets stored into the first interval (ACDCALLS1).	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACDTIME	Talk time of all ACDCALLS . ACDTIME includes O_ACDTIME , but does not include HOLDTIME .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACWINCALLS	Number of inbound extension calls received by agents while in ACW for split/skill ACD calls or in ACW .	C	integer	4 bytes
ACWINTIME	Talk time of all ACWINCALLS . ACWINTIME does not include hold time on Generic 2.2 and Generic 3 switches. It does include time spent on calls received while in ACW not associated with an ACD call.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACWOUTADJCALLS	Number of ACWOUTCALLS that were placed by an adjunct on behalf of an agent (keyboard-dialed). If such calls are placed to off-switch destinations, then they are also counted as ACWOUTOFFCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer	4 bytes
ACWOUTCALLS	Number of outbound extension calls made by agents or on behalf of agents while in ACW . This includes ACW for split/skill ACD calls and ACW not associated with a call. ACWOUTCALLS includes ACWOUTADJCALLS and ACWOUTOFFCALLS .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
ACWOUTOFFCALLS	Number of ACWOUTCALLS that were made to a an off-switch destination—a destination outside the switch. If such calls are placed by an adjunct on behalf of an agent while in ACW, they are also counted as ACWOUTADJCALLS . Available for external calls with Generic 2.2 and Generic 3 switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACWOUTOFFTIME	Talk time of all ACWOUTOFFCALLS (does not include time on hold). ACWOUTOFFTIME includes ACWOUTTIME . Available for external calls with Generic 2.2 and Generic 3 switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
ACWOUTTIME	Talk time of all ACWOUTCALLS . ACWOUTTIME does not include hold time on Generic 2.2 and Generic 3 switches. It does include time spent on calls made while in ACW not associated with an ACD call and on ACWOUTADJCALLS and on ACWOUTOFFCALLS .	C	integer	4 bytes
ACWTIME	Duration of all after call work associated with ACDCALLS . NOTE: ACWTIME does not include time spent in ACW not associated with an ACD call (that is, the agent pressed the ACW button while not on an ACD call). However, both ACWINTIME and ACWOUTTIME do include time spent on calls made or received while in ACW not associated with an ACD call. Therefore, the sum of ACWINTIME and ACWOUTTIME may be greater than ACWTIME . ACWTIME includes ACWINTIME , ACWOUTTIME , and O_ACWTIME .	C	integer	4 bytes
ANSTIME	Time spent by callers in queue or ringing before being answered by an agent.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
ASSISTS	The number of times the supervisor was called (supervisor assists) by agents on split/skill calls, direct agent ACD calls, or in call-related ACW for this split/skill.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
AUXINCALLS	The number of inbound extension calls received by agents while in AUX (auxiliary work), AVAILABLE ; or, for Generic 2.2 and Generic 3 switches, with an ACD or AUXIN/AUXOUT call on hold. AUXINCALLS are recorded in the SPLIT that is OLDEST_LOGON for agents in multiple splits/skills.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
AUXINTIME	The talk time of all AUXINCALLS (does not include hold time on Generic 2.2 and Generic 3 switches).	C	integer	4 bytes
AUXOUTADJCALLS	The number of AUXOUTCALLS that were placed by an adjunct on behalf of an agent (keyboard-dialed). If such calls are placed to off-switch destinations, then they are also counted as AUXOUTOFFCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
AUXOUTCALLS	The number of outbound extension calls made by agents while in AUX (auxiliary work), AVAILABLE ; or, for Generic 2.2 and Generic 3 switches with an ACD or AUXIN/AUXOUT call on hold. AUXOUTCALLS are recorded for the SPLIT which is the OLDEST_LOGON , unless the agent made the call with an ACD call on hold. In this case, they are recorded for the split/skill of the ACD call. AUXOUTCALLS includes ACDAUXOUTCALLS , AUXOUTADJCALLS , and AUXOUTOFFCALLS .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
AUXOUTOFFCALLS	The number of AUXOUTCALLS that were made to a destination outside the switch. If such calls are placed by an adjunct on behalf of an agent, they are also counted as AUXOUTADJCALLS . Available for external calls with Generic 2.2 and Generic 3 switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
AUXOUTOFFTIME	The talk time of all AUXOUTOFFCALLS (does not include AUXOUTOFFCALLS spent on hold). AUXOUTOFFTIME is included in AUXOUTTIME . Available for external calls on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes
AUXOUTTIME	Talk time of all AUXOUTCALLS . AUXOUTTIME does not include hold time on Generic 2.2 and Generic 3 switches. AUXOUTTIME includes AUXOUTOFFTIME .	C	integer	4 bytes
BACKUPCALLS	Number of ACDCALLS that were delivered to and answered by this split/skill by a vector command other than "queue to main". This includes calls delivered by "messaging split/skill", "check backup", "route to" split/skill, and "redirect on no answer" vector routing. Calls that are redirected back to the split/skill from ringing by the "redirect on no answer" feature that are subsequently answered by an agent in the split/skill are also counted as backup calls. Available on Generic 2 and Generic 3 switches with the Vectoring feature. NOTE: The Redirect on No Answer VDN routing feature is also available on the <i>DEFINITY</i> ECS.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
BUSYCALLS	Number of CALLSOFFERED calls that were given a busy signal by the switch. This happens when a “busy” vector command is executed while the call is queued to this split/skill (and this is the primary split/skill the call is queued to) or if a call queued to this split/skill forwards to another split/skill whose queue is full. On Generic 3 and later switches, a busy is given because a non-vector controlled split has a full queue, no queue and no available agents, or no agents that are staffed.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
BUSYTIME	Time callers waited in queue until hearing a busy tone for all BUSYCALLS .	C	integer	4 bytes
CALLSOFFERED	Number of calls that queued to the split/skill and that completed during the interval. This does NOT include calls on the Generic 3 switch that could not queue to the split/skill because the queue was full or there was no queue. CALLSOFFERED = ACDCALLS + ABNCALLS + BUSYCALLS + DISCCALLS + OUTFLOWCALLS + DEQUECALLS. CALLSOFFERED includes ABNCALLS, RINGCALLS, OTHERCALLS, and INFLOWCALLS.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
CONFERENCE	Number of ACDCALLS that were conferenced at least once. Available on Generic 2.2 and Generic 3 switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
DA_ACWINCALLS	Number of inbound extension calls agents answered while in after call work mode for direct agent ACD calls that were queued through this split/skill. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
DA_ACWINTIME	Talk time of inbound extension calls agents answered while in the after call work mode for direct agent ACD calls queued through this split/skill. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer	4 bytes
DA_ACWOCALLS	Number of outbound extension calls agents made while in the after call work mode for direct agent call ACD calls queued through this split/skill. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
DA_ACWOTIME	Talk time of outbound extension calls the agent made while in the after call work mode for a direct agent ACD call. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer	4 bytes
DEQUECALLS	Number of calls that queued to this split/skill as a nonprimary split/skill, but whose disposition was recorded in another split/skill (as answered, abandoned, outflowed, busy, or forced disconnect). Requires vectoring for multiple split/skill queueing on a Generic 2.2 switch with EAS or a Generic 3 switch.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
DEQUETIME	Amount of time DEQUECALLS waited in this split/skill queue before dequeuing. Requires vectoring for multiple split/skill queueing on a Generic 2.2 switch with EAS or a Generic 3 switch.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
DISCCALLS	<p>For Generic 2.2 switches, this is the number of CALLSOFFERED that were disconnected by the switch via the “disconnect” vector command.</p> <p>For the Generic 3 Version 2 and newer switch releases, this also includes the number of CALLSOFFERED that were disconnected by the switch when the vector disconnect timer expired.</p> <p>For the vectoring feature on Generic 3 Version 1 switches, this is the number of CALLSOFFERED that were given a forced disconnect announcement, then were disconnected by the switch. The disconnect announcement is for a “disconnect” vector command.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
DISCTIME	<p>Time all DISCCALLS spent in this split’s/skill’s queue.</p> <p>For Generic 2.2 switches, this is the time until the trunk drops, in the case where the caller hangs up without listening to the entire announcement.</p> <p>For the Generic 3 Version 2 and newer switches (if the call is disconnected due to the expiration of the vector disconnect timer), this is the time until the call is disconnected by the switch.</p> <p>For the vectoring feature on Generic 2.1 or Generic 3 Version 1 switches, this is the time until the announcement ends and the caller is disconnected by the switch.</p>	C	integer	4 bytes
EVENT1-9	<p>Number of times each event (stroke count) feature button (feature button 1 to 9) was pressed by agents on split/skill or direct agent ACD calls or in after call work associated with an ACD call for this split/skill. Available with Generic 2 and Generic 3 switches.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
HIGHCALLS	Number of ACDCALLS with high priority that were answered by agents in this split/skill (for example, answered calls that were queued to the split/skill with high priority by a “queue to main” or “check backup” vector command). For Generic 3 switches with the Vectoring feature, this includes calls that were queued to a split/skill with priority using the “route to” or “messaging split/skill” vector commands, and calls that queued directly to a split/skill with priority. (Priority in these cases is determined by the class of restriction of the originator, which is an agent, an extension, a trunk group or a VDN.) Available on Generic 2 and Generic 3 switches with the Vectoring feature.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
HOLDABNCALLS	Number of times split/skill ACD callers abandoned the call while on hold. Available on Generic 2 and Generic 3 switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
HOLDCALLS	Number of split/skill ACD calls that were placed on hold at least once. Available on Generic 2 and Generic 3 switches. HOLDCALLS includes HOLDABNCALLS .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
HOLDTIME	Time spent by split/skill ACD callers on hold. Available on Generic 2 and Generic 3 switches.	C	integer	4 bytes
I_ACDAUXINTIME	Time during the collection interval that POSITIONS were talking on AUXIN calls with a split/skill ACD call on hold where SPLIT is OLDEST_LOGON . Available on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_ACDAUX_OUTTIME	Time during the collection interval that POSITIONS spent dialing (Generic 2.2) and talking on AUXOUT calls with a split/skill ACD call for this split/skill on hold. NOTE: In a multiple call handling environment with agents in multiple skills, the ACD call for this skill must have been the last ACD call to have been put on hold before the agent made the AUXOUT call. Available on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes
I_ACDOTHERTIME	Time during the collection interval that POSITIONS spent in the OTHER state (dialing an outgoing call with a Generic 3 switch, with a ringing extension call with Generic 3 switch, or with calls on hold and with no other state selected) with a split/skill ACD call on hold. Available on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes
I_ACDTIME	Time during the collection interval that POSITIONS were on split/skill ACD calls. This includes time on O_ACDCALLS as well as on ACDCALLS .	C	integer	4 bytes
I_ACWINTIME	The time during the collection interval that POSITIONS were in ACW for this split/skill, either associated with a split/skill ACD call or not associated with a call, and on inbound extension calls. This does not include time inbound extension calls spent on hold. Available on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_ACWOUTTIME	The time during the collection interval that POSITIONS were in ACW for this split/skill, either associated with this split/skill ACD call or not associated with a call, and on outbound extension calls. This does not include time outbound extension calls spent on hold. Available on Generic 2.2 and Generic 3 switches.	C	integer	4 bytes
I_ACWTIME	The time during the collection interval that POSITIONS were in ACW for this split/skill, either associated with a split/skill ACD call or not associated with a call. This I_ACWTIME includes I_ACWINTIME and I_ACWOUTTIME .	C	integer	4 bytes
I_ARRIVED	The number of calls that queued to this split/skill during this interval.	C	integer	4 bytes
I_AUXINTIME	The time during the collection interval that POSITIONS were in AUX work, AVAILABLE , or, for Generic 2.2 and Generic 3 switches, had an ACD or AUXIN/AUXOUT call on hold and were on inbound extension calls. I_AUXINTIME includes I_ACDAUXINTIME .	C	integer	4 bytes
I_AUXOUTTIME	The time during the collection interval that POSITIONS were in AUX work, AVAILABLE ; or, for Generic 2.2 and Generic 3 switches, had an ACD or AUXIN/AUXOUT call on hold and were on outbound extension calls. I_AUXOUTTIME includes I_ACDAUX_OUTTIME .	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_AUXTIME	The time during the collection interval that POSITIONS were in AUX in this split/skill. I_AUXTIME = I_AUXTIME0 + I_AUXTIME1 + I_AUXTIME2 + I_AUXTIME3 + I_AUXTIME4 + I_AUXTIME5 + I_AUXTIME6 + I_AUXTIME7 + I_AUXTIME8 + I_AUXTIME9. I_AUXTIME includes I_AUXTIME0 , I_AUXTIME1-9 , I_AUXINTIME , I_AUXOUTTIME , and I_TAUXTIME .	C	integer	4 bytes
I_AUXTIME0	The time during the collection interval that POSITIONS were in AUX for reason code 0 in this split/skill. This includes time on extension calls from this AUX state. For switches with AUX reason codes active, this represents time agents spent in “system” AUX . For switches without AUX reason codes active, I_AUXTIME0 is the same as I_AUXTIME .	C	integer	4 bytes
I_AUXTIME1-9	Time during the collection interval that POSITIONS were in AUX for each reason code 1-9 in this skill. This includes time on extension calls from each AUX state. Available for Generic 3 Version 5 and later Generic 3 switches with EAS.	C	integer	4 bytes
I_AVAILTIME	Time during the collection interval that POSITIONS were available for calls from this split/skill. I_AVAILTIME includes I_TAVAILTIME .	C	integer	4 bytes
I_DA_ACDTIME	Time during the collection interval that the agent spent talking on direct agent ACD calls queued through this split/skill. I_DA_ACDTIME is a subset of I_OTHERTIME . Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_DA_ACWTIME	<p>The time that POSITIONS spent in ACW for direct agent ACD calls queued through this split/skill. I_DA_ACWTIME is a subset of I_OTHERTIME.</p> <p>Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.</p>	C	integer	4 bytes
I_OTHERTIME	<p>The time during the collection interval that POSITIONS were doing other work. I_OTHERTIME is collected for the time period after the link to the switch comes up or after the agent logs in and before the CMS receives notification of the agent's state from the switch.</p> <p>For Generic 3 switches, other work includes: while in Auto-In or Manual-In mode, an agent put any call on hold and performed no further action; the agent had a direct agent call ringing, was on a direct agent call or in ACW for a direct agent call; the agent dialed to place a call or activate a feature; or an extension-in call rang at the agent's voice terminal with no other activity.</p> <p>For Generic 2.2 with EAS and Generic 3 switches, the other work includes the time agents were logged into multiple splits/skills and doing work for a split/skill other than this one (with an ACD call ringing, talking on an ACD call, or in ACW for a split/skill other than this one).</p> <p>For Generic 3 switches with EAS and multiple call handling, agents are available in other, multiple call handling skills, but not in this skill.</p> <p>I_OTHERTIME includes I_ACDOTHERTIME, I_DA_ACDTIME, and I_DA_ACWTIME.</p>	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_RINGTIME	The time during the collection interval that agents were in the ringing state for calls to this split/skill. If the agent changes work modes or answers/makes another call instead of answering the ringing call, I_RINGTIME will stop accumulating. RINGTIME is the time the caller spends ringing and is independent of agent activity. NOTE: With forced multiple call handling (Generic 3 Version 4 and later), if an ACD call rings at the agent's voice terminal while the agent is talking on another call, I_RINGTIME does not accumulate. Available on Generic 2 and Generic 3 switches for ring tracking.	C	integer	4 bytes
I_STAFFTIME	The time during the collection interval that POSITIONS were staffed (logged in). I_STAFFTIME = I_AVAILTIME + I_ACDDTIME + I_ACWTIME + I_AUXTIME + I_RINGTIME + I_OTHERTIME. I_STAFFTIME includes I_ACDDTIME , I_ACWTIME , I_AUXTIME , I_AVAILTIME , I_OTHERTIME , and I_RINGTIME .	C	integer	4 bytes
I_TAUXTIME	The time top agents in this split/skill were in AUX mode. This includes time on AUXIN/AUXOUT calls, received or made without an ACD call on hold. (Time on AUXIN/AUXOUT calls with an ACD call on hold is tracked in I_ACDAUXINTIME and I_ACDAUX_OUTTIME .) Available with a Generic 3 switch with the EAS feature for top skills.	C	integer	4 bytes
I_TAVAILTIME	The time top agents in this split/skill were available to receive calls for this split/skill. Available with a Generic 3 switch with the EAS feature for top skills.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
INCOMPLETE	Indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2 switches), the trunk going maintenance busy with a call active (Generic 3 switches), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month.	C	smallint	2 bytes
INFLOWCALLS	Number of calls that were redirected to the split's/skill's queue from another queue. When a call leaves the VDN (for example, by routing to another VDN) or leaves vector processing, (for example, by routing to a split/skill) the next split/skill to which a call queues will not be credited with an inflow. Calls that ring at an agent and are then requeued to the same split/skill by the Redirect on No answer feature are counted as inflows to that split/skill. On Generic 2 and Generic 3 switches with vectoring, an inflow is counted for calls that intraflow from one split's queue to another (that is, call that queue to a split after having previously been queued to another split).	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
INFLOWCALLS (contd)	<p>For Generic 2 with vectoring, the calls that queue to a split and then subsequently queue to a new split by a “queue to main” or “check backup” split vector command and are counted as inflows to the new split.</p> <p>On Generic 2.2 switches with EAS, an inflow is counted for this skill when a call is answered by an agent in a non-primary skill. Calls that are queued to one skill group and are subsequently queued to another are not counted as inflows to the subsequent skill group. Calls that queue to the “zero” skill after having been queued to a “nonzero” skill are not counted as inflows to the “zero” skill. Similarly, calls that queue to a “nonzero” skill after having been queued to a “zero” skill are not counted as inflows to the “nonzero” skill.</p> <p>For Generic 3 with vectoring and Generic 2.2 with EAS and multiple split/skill queueing, calls answered by an agent in a non-primary split/skill are counted as inflows to that split/skill. Calls that abandon from ringing at an agent’s voice terminal in a non-primary split/skill are also counted as inflows to that skill.</p> <p>On Generic 3 Version 2 and newer switches, calls that ring at an agent in this split/skill and then requeue to the same split/skill by the Redirection on No Answer to a Split/Skill feature are counted as inflows.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
INTERFLOWCALLS	Number of OUTFLOWCALLS that were redirected to a destination outside the switch.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
INTRVL	Number of minutes in the intrahour interval (15, 30, or 60). INTRVL applies to intrahour tables only.	A	smallint; only in hsplit table	2 bytes

Database Item	Description	Data Type	Column Type	Length
LOWCALLS	For switches with vectoring, this is the number of ACDCALLS with low priority that were answered by this split/skill. For switches without vectoring, this is the number of ACDCALLS with no priority that were answered by this split/skill.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
MAXINQUEUE	Maximum number of simultaneous calls in this split's/skill's queue during the collection interval.	M	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
MAXOCWTIME	Maximum amount of time that a call, recorded during the collection interval, waited in queue and ringing before an agent answered in this split/skill, the caller abandoned, or the call was redirected, received a busy signal, or was disconnected.	M	integer	4 bytes
MAXSTAFFED	Maximum number of agent POSITIONS that were simultaneously staffed during the collection interval. MAXSTAFFED includes MAXTOP .	M	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
MAXTOP	Maximum number of top agents that were staffed during the collection interval in this split/skill.	M	integer	4 bytes
MEDCALLS	For switches with vectoring, the number of ACDCALLS with medium priority that were answered by agents in the split/skill. For example, answered calls that were queued to the split/skill with medium priority by a "queue to main" or "check backup" vector command. For Generic 3 switches with vectoring, MEDCALLS includes calls that were queued to a split/skill with no priority using the "route to" or "messaging split" vector commands, calls that queued directly to a non-vector-controlled split with no priority, and calls that intraflowed to a split/skill with no priority. For switches without vectoring, the number of ACDCALLS with "yes" priority that were answered by agents in the split/skill.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
NOANSREDIR	<p>The number of split/skill ACD calls that rang at agent positions in the split/skill and then were automatically redirected back to the split/skill queue or to a VDN by the Redirection on No Answer feature because they were not answered.</p> <p>Redirection On No Answer to a split/skill is available on <i>DEFINITY</i> ECS or Generic 3 Version 2 or later ECS. Redirection On No Answer to a VDN is available only on <i>DEFINITY</i> ECS R5 and later.</p> <p>NOTE: When a call is requeued to the same split/skill using the Redirection on No Answer feature, it is counted as an outflow from the split/skill and an inflow to the same split/skill. This is NOT true for calls that are redirected to a VDN using the Redirection on No Answer feature, rather than redirecting the call back to the same split/skill. Such calls count as outflows from the original split or skill, but do not count as inflows to the next split/skill to which they are queued through the new VDN. They are also counted as NOANSREDIR calls and can be subtracted out from the split/skill outflows and inflows to calculate the number of outflows and inflows that were not due to requeuing the call to the same split. This does not apply to Redirection on No Answer calls to a VDN.</p> <p>Such calls count as outflows from the original split or skill, but do not count as inflows to the next split/skill to which they are queued through the new VDN. It is also counted as a NOANSREDIR call and so can be subtracted out from the outflows and from the inflows to calculate the number of outflows and inflows that were not due to requeuing the call to the same split.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
O_ABNCALLS	The number of ABNCALLS that were placed by an adjunct, that is, the number of outbound predictive dialing calls that were abandoned by the far end. O_ABNCALLS is a subset of ABNCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
O_ACDCALLS	The number of ACDCALLS that were placed by an adjunct (outbound predictive dialing). O_ACDCALLS includes DA_ACDCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
O_ACDTIME	The talk time of all O_ACDCALLS (does not include time calls spent on hold). Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature. ACDTIME includes O_ACDTIME .	C	integer	4 bytes
O_ACWTIME	The duration of all after call work associated with O_ACDCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature. O_ACWTIME is included in ACWTIME .	C	integer	4 bytes
O_OTHERCALLS	The number of outbound calls queued to this split/skill that were not answered or abandoned as ACD split/skill calls. These include forced busy calls and calls with unknown dispositions. Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
OTHERCALLS	The number of calls offered to this split/skill that did not abandon and were not answered by an ACD agent for this split/skill. OTHERCALLS = BUSYCALLS + DISCCALLS + OUTFLOWCALLS + DEQUEUECALLS.	C	integer	4 bytes
OTHERTIME	The time OTHERCALLS waited in queue until the disposition was known and the call left the split/skill. NOTE: OTHERTIME relates to time for OTHERCALLS and is not related to I_OTHERTIME , which is the time agents spent in the OTHER state. OTHERTIME = BUSYTIME + DEQUETIME + DISCTIME + OUTFLOWTIME.	C	integer	4 bytes
OUTFLOWCALLS	Number of CALLSOFFERED that were redirected to another destination while queued to this split/skill. This can happen under different circumstances, depending on the switch release and on whether vectoring is active or not.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
OUTFLOWCALLS (contd)	<p>For Generic 3 switches without vectoring, this is the number of CALLSOFFERED that were redirected to another destination while queued to this split/skill. This can happen if:</p> <ul style="list-style-type: none"> • the call intraflowed or interflowed. • the split/skill call forwarding was active. • a ringing ACD call was answered using call pickup. • a ringing ACD call redirects on no answer. <p>For Generic 3 switches with vectoring, the number of outflow calls are counted if:</p> <ul style="list-style-type: none"> • a ringing ACD call redirects on no answer. • the call rang at an agent in this split/skill and was answered using call pickup. • the call was routed to another VDN. • the call routed to a number or digits. • the call queued to a messaging split/skill. • the call queued to this split/skill as the primary split/skill and was answered by an agent in another split/skill, rang at an agent in another split/skill and then abandoned or was redirected by the Redirection on No Answer feature (for Generic 3 Version 2 and later switch releases). <p>For Generic 3 Version 2 switches, this is the number of CALLSOFFERED that were redirected to another destination while queued to this split/skill. This can happen by requeueing to the same split/skill via the Redirect on No Answer feature.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

Database Item	Description	Data Type	Column Type	Length
OUTFLOWCALLS (contd)	<p>For Generic 2 switches without vectoring, outflows are counted when calls intraflow to another split or extension.</p> <p>For Generic 2 switches with vectoring (except for Generic 2.2 with EAS), outflows are counted if:</p> <ul style="list-style-type: none"> • the call was routed to another VDN. • the call is routed to a number. • the call queued to another split (using “queue to main” or “check backup”). <p>For Generic 2 switches with vectoring, without EAS, this is the number of CALLSOFFERED that were redirected to another destination while queued to this split/skill. This can happen if:</p> <ul style="list-style-type: none"> • the call intraflowed or interflowed. • the split/skill call forwarding was active. • the call queued to another split. <p>For Generic 2.2 switches with EAS, outflows are counted if:</p> <ul style="list-style-type: none"> • the call was routed to another VDN. • the call is routed to a number. • the call is queued to this skill which is a “nonzero” skill as primary and then is queued to the “zero” skill. • a call is queued to this skill which is a “zero” skill and then is queued to a “nonzero” skill. • the call is queued to this skill as primary and then is subsequently queued to another skill. <p>OUTFLOWCALLS includes INTERFLOWCALLS, NOANSREDIR, and SLVLOUTFLOWS.</p>	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
OUTFLOWTIME	The time all OUTFLOWCALLS waited in queue or ringing before being redirected.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
PERIOD1-9	The length, in seconds, of each service level increment as defined in the ACD Administration: Split/Skill Call Profile window. Each increment represents a progressively longer wait time. <i>CentreVu</i> CMS counts answered or abandoned calls that wait beyond the last increment (PERIOD9) in either ACDCALLS10 or ABNCALLS10 .	A	smallint	2 bytes
PERIODCHG	Indicates whether or not service level increments PERIOD1-9 (as defined on the ACD Administration: Split/Skill Call Profile window) changed during the collection interval. Valid values for PERIODCHG are 1 = YES and 0= NO.	A	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
PHANTOMABNS	The number of split/skill ACD calls with talk time less than the value of the phantom-abandoned call timer. Available on Generic 3 and newer switches.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
RINGCALLS	The number of this split's/skill's calls that rang at agent positions. Available on a Generic 2 and Generic 3 switch for ring tracking. RINGCALLS includes ACDCALLS and NOANSREDIR .	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
RINGTIME	The time this split's/skill's calls spent ringing at agent positions independent of final disposition and other agent activity. I_RINGTIME is the time the agent spends with ringing calls and is affected by other agent activity. RINGTIME is the time the caller spends ringing and is independent of agent activity. Available on a Generic 2 and Generic 3 switch for ring tracking.	C	integer	4 bytes
ROW_DATE (index)	Date on which data was collected.	I	date	<i>Informix</i> date, 4 bytes

Database Item	Description	Data Type	Column Type	Length
SERVICELLEVEL	Number of seconds within which calls must be answered/connected in order to be considered acceptable (as defined on the ACD Administration: Split/Skill Call Profile window).	A	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
SLVLABNS	Number of ABNCALLS whose time to abandon was less than or equal to this split's/skill's SERVICELLEVEL .	C	integer	4 bytes
SLVLOUTFLOWS	Number of OUTFLOWCALLS whose time to outflow was less than or equal to this split's/skill's SERVICELLEVEL .	C	integer	4 bytes
SPLIT (index)	Split/skill number for which data was collected.	I	smallint	2 bytes
STARTTIME	Start time of the interval for which data was collected. STARTTIME applies only to the interval table.	I	smallint; only in hsplit table	2 bytes
SVCLEVELCHG	Indicates whether or not the service level was changed during the collection interval. Valid values for SVCLEVELCHG are 1 = YES and 0 = NO.	A	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
TOPCALLS	The number of ACDCALLS with top priority that were answered by agents in this split/skill. Available with Generic 2 and Generic 3 switches with vectoring.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table
TRANSFERRED	The number of ACDCALLS that were transferred to another destination. For Generic 2.1 switches, includes transfers to a measured VDN or split. For Generic 2.2 or Generic 3 switches, includes all split/skill calls transferred.	C	integer; smallint in hsplit table	4 bytes, 2 bytes in hsplit table

ECS Cross-Reference

The following table lists which ECS release supports each Split/Skill database item.

⇒ NOTE:

The following is a key to the database items tables:

- Items marked “X” indicate that the database item is supported by the specified ECS release.
- Items marked “EAS” require that the EAS feature be active on the ECS for the items to be populated.
- Items marked “e” are populated for the releases shown, but the values are only meaningful for EAS releases.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	<i>DEFINITY</i> ECS R5	<i>DEFINITY</i> ECS R6
ABNCALLS	X	X	X	X	X	X	X
ABNCALLS1-10	X	X	X	X	X	X	X
ABNRINGCALLS	X	X	X	X	X	X	X
ABNTIME	X	X	X	X	X	X	X
ACCEPTABLE	X	X	X	X	X	X	X
ACD (index)	X	X	X	X	X	X	X
ACDAUXOUTCALLS		X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
ACDCALLS1-10	X	X	X	X	X	X	X
ACDTIME	X	X	X	X	X	X	X
ACWINCALLS	X	X	X	X	X	X	X
ACWINTIME	X	X	X	X	X	X	X
ACWOUTADJCALLS		X	X	X	X	X	X
ACWOUTCALLS	X	X	X	X	X	X	X
ACWOUTOFFCALLS		X	X	X	X	X	X
ACWOUTOFFTIME		X	X	X	X	X	X
ACWOUTTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ACWTIME	X	X	X	X	X	X	X
ANSTIME	X	X	X	X	X	X	X
ASSISTS	X	X	X	X	X	X	X
AUXINCALLS	X	X	X	X	X	X	X
AUXINTIME	X	X	X	X	X	X	X
AUXOUTADJCALLS		X	X	X	X	X	X
AUXOUTCALLS	X	X	X	X	X	X	X
AUXOUTOFFCALLS		X	X	X	X	X	X
AUXOUTOFFTIME		X	X	X	X	X	X
AUXOUTTIME	X	X	X	X	X	X	X
BACKUPCALLS	X	X	X	X	X	X	X
BUSYCALLS	X	X	X	X	X	X	X
BUSYTIME	X	X	X	X	X	X	X
CALLSOFFERED	X	X	X	X	X	X	X
CONFERENCE		X	X	X	X	X	X
DA_ACWINCALLS			X	X	X	X	X
DA_ACWINTIME			X	X	X	X	X
DA_ACWOCALLS			X	X	X	X	X
DA_ACWOTIME			X	X	X	X	X
DEQUECALLS		EAS	X	X	X	X	X
DEQUETIME		EAS	X	X	X	X	X
DISCCALLS	X	X	X	X	X	X	X
DISCTIME	X	X	X	X	X	X	X
EVENT1-9	X	X	X	X	X	X	X
HIGHCALLS	X	X	X	X	X	X	X
HOLDABNCALLS	X	X	X	X	X	X	X
HOLDCALLS	X	X	X	X	X	X	X
HOLDTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
I_ACDAUXINTIME		X	X	X	X	X	X
I_ACDAUX_OUTTIME		X	X	X	X	X	X
I_ACDOTHERTIME		X	X	X	X	X	X
I_ACDTIME	X	X	X	X	X	X	X
I_ACWINTIME	X	X	X	X	X	X	X
I_ACWOUTTIME	X	X	X	X	X	X	X
I_ACWTIME	X	X	X	X	X	X	X
I_ARRIVED	X	X	X	X	X	X	X
I_AUXINTIME	X	X	X	X	X	X	X
I_AUXOUTTIME	X	X	X	X	X	X	X
I_AUXTIME	X	X	X	X	X	X	X
I_AUXTIME0	X	X	X	X	X	X	X
I_AUXTIME1-9						EAS	X
I_AVAILTIME	X	X	X	X	X	X	X
I_DA_ACDTIME			X	X	X	X	X
I_DA_ACWTIME			X	X	X	X	X
I_OTHERTIME	X	X	X	X	X	X	X
I_RINGTIME	X	X	X	X	X	X	X
I_STAFFTIME	X	X	X	X	X	X	X
I_TAUXTIME	X	X	X	X	X	X	X
I_TAVAILTIME	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INFLOWCALLS	X	X	X	X	X	X	X
INTERFLOWCALLS	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
LOWCALLS	X	X	X	X	X	X	X
MAXINQUEUE	X	X	X	X	X	X	X
MAXOCWTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
MAXSTAFFED	X	X	X	X	X	X	X
MAXTOP	*	*	*	*	*	*	X
MEDCALLS	X	X	X	X	X	X	X
NOANSREDIR				X	X	X	X
O_ABNCALLS		X	X	X	X	X	X
O_ACDCALLS		X	X	X	X	X	X
O_ACDTIME		X	X	X	X	X	X
O_ACWTIME		X	X	X	X	X	X
O_OTHERCALLS		X	X	X	X	X	X
OTHERCALLS	X	X	X	X	X	X	X
OTHERTIME	X	X	X	X	X	X	X
OUTFLOWCALLS	X	X	X	X	X	X	X
OUTFLOWTIME	X	X	X	X	X	X	X
PERIOD 1-9	X	X	X	X	X	X	X
PERIODCHG	X	X	X	X	X	X	X
PHANTOMABNS			X	X	X	X	X
RINGCALLS	X	X	X	X	X	X	X
RINGTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SERVICELEVEL	X	X	X	X	X	X	X
SLVLABNS	X	X	X	X	X	X	X
SLVLOUTFLOWS	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X
STARTTIME	X	X	X	X	X	X	X
SVCLEVELCHG	X	X	X	X	X	X	X
TOPCALLS	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X

*This item is populated for this release, but the values are only meaningful if you are using a Generic 3 EAS release, with Skill Level call distribution.

Agent Database Items

Overview

The Agent database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items typically apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical agent database items apply to the Intrahour Agent (*hagent*), Daily Agent (*dagent*), Weekly Agent (*wagent*), and Monthly Agent (*magent*) tables. All items listed in the following table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four agent tables are also noted in the Column Type column. The historical indexes are **LOGID**, **SPLIT**, and **ROW_DATE**.

Agent Database Items

The following table describes the data items in the CMS Agent database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	The number of split/skill ACD calls that were abandoned while ringing the agent's voice terminal (after being directed to the agent voice terminal, but before being answered). This includes calls considered abandoned because their talk time was less than the phantom-abandoned call timer. For Generic 3 switches, ABNCALLS includes PHANTOMABNS . Available on Generic 2 and Generic 3 switches.	C	smallint (<i>dagent</i> , <i>hagent</i>); integer (<i>magent</i> , <i>wagent</i>)	2 bytes (<i>dagent</i> , <i>hagent</i>); 4 bytes (<i>magent</i> , <i>wagent</i>)

Database Item	Description	Data Type	Column Type	Length
ABNTIME	The time split/skill ACD callers waited while ringing the agent's voice terminal before the call was abandoned. For Generic 3 switches, ABNTIME includes the time until the agent releases the call for phantom-abandoned calls. Also available on Generic 2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ACD (index)	The ACD number for which data was collected.	I	smallint	2 bytes
ACD_RELEASE	The number of split/skill ACD calls that the agent released or dropped before the far end released. NOTE: The transfers and conferences are always recorded as agent-released calls. Available for Generic 3 switches.	C	integer; not in hagent table	4 bytes
ACDAUXOUTCALLS	The number of AUXOUTCALLS the agent made with at least one split/skill or direct agent ACD call on hold. This includes calls made to transfer or conference the ACD call. Available on Generic 2.2 and Generic 3 switches.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
ACDCALLS	The number of calls that were queued to SPLIT and answered by this agent in this SPLIT . ACDCALLS includes O_ACDCALLS and ACD_RELEASE .	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
ACDTIME	The talk time of all ACDCALLS . ACDTIME includes O_ACDTIME . It does not include HOLDTIME except on Generic 2.1 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
ACWINCALLS	The number of inbound extension calls received by the agent while in ACW. This includes ACW for split/skill and direct agent ACD calls and ACW not associated with a call.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
ACWINTIME	Talk time of all ACWINCALLS . ACWINTIME includes DA_ACWINCALLS , but does not include HOLDTIME except on Generic 2.1 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ACWOUTADJCALLS	The number of ACWOUTCALLS that were placed by an adjunct on behalf of an agent (keyboard-dialed). If such calls are placed to off-switch destinations, then they are also counted as ACWOUTOFFCALLS . Available on the Generic 2.2 switch with the ASAI Gateway Interface feature and on the Generic 3 switch with the ASAI feature.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
ACWOUTCALLS	The number of outbound extension calls made by the agent or on behalf of the agent while in ACW. This includes ACW for split/skill ACD calls and ACW not associated with a call. ACWOUTCALLS includes ACWOUTADJCALLS , ACWOUTOFFCALLS , and DA_ACWOCALLS .	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
ACWOUTOFFCALLS	The number of ACWOUTCALLS that were made to an off-switch destination—a destination outside the switch. If these calls were placed by an adjunct on behalf of the agent (keyboard-dialed), then they are counted as ACWOUTADJCALLS . Available for external calls on Generic 2.2 and Generic 3 switches.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

Database Item	Description	Data Type	Column Type	Length
ACWOUTOFFTIME	The talk time of all ACWOUTOFFCALLS (does not include time on hold). ACWOUTTIME includes ACWOUTOFFTIME . Available for external calls on Generic 2.2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ACWOUTTIME	The talk time of all ACWOUTCALLS . ACWOUTTIME does not include HOLDTIME (except for Generic 2.1 switches). ACWOUTTIME includes time spent on calls made while in ACW that was not associated with an ACD call and on ACWOUTADJCALLS and on ACWOUTOFFCALLS . For Generic 2.2 and generic 3 switches, ACWOUTTIME does not include time ACWOUTCALLS spent on hold. It does include time spent on calls made while in ACW not associated with an ACD call.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ACWTIME	The duration of all after call work associated with ACDCALLS , including ACWINTIME and ACWOUTCALLS received/made during call-associated ACW. NOTE: ACWTIME does not include the time spent in ACW not associated with an ACD call (that is, the agent pressed the ACW button while not on an ACD call). However, both ACWINTIME and ACWOUTTIME do include time spent on calls made or received while in ACW not associated with an ACD call. Therefore, the sum of ACWINTIME and ACWOUTTIME may be greater than ACWTIME . ACWTIME includes ACWINTIME , ACWOUTTIME , DA_ACWTIME , and O_ACWTIME .	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
ANSRINGTIME	The time split/skill and direct agent ACD calls spent ringing at the agent's voice terminal before being answered. Available for ring-tracking on Generic 2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ASSISTS	The number of times the supervisor was called (supervisor assists) by agents on a split/skill direct agent ACD calls, or in call-related ACW for this split/skill.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
AUXINCALLS	The number of inbound extension calls received by agents while in AUX (auxiliary work), AVAILABLE ; or, for Generic 2.2 and Generic 3 switches, with an ACD or AUXIN/AUXOUT call on hold.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
AUXINTIME	The talk time of all AUXINCALLS (does not include HOLDTIME except on Generic 2.1 switches).	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
AUXOUTADJCALLS	The number of AUXOUTCALLS that were placed by an adjunct on behalf of an agent (keyboard dialed). If such calls are placed to off-switch destinations, then they are also counted as AUXOUTOFFCALLS . Available for outbound calls on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

Database Item	Description	Data Type	Column Type	Length
AUXOUTCALLS	<p>The number of outbound extension calls that were made by the agent or on behalf of the agent while in AUX (auxiliary work), AVAILABLE; or, for Generic 2.2 and Generic 3 switches with an ACD or AUXIN/AUXOUT call on hold.</p> <p>NOTE: Calls the agent makes to transfer or conference an ACD call are included as AUXOUT calls.</p> <p>AUXOUTCALLS includes AUXOUTADJCALLS, AUXOUTOFFCALLS, and ACDAUXOUTCALLS.</p>	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
AUXOUTOFFCALLS	<p>The number of AUXOUTCALLS that were made to a destination outside the switch. If such calls were placed by an adjunct on behalf of the agent (keyboard-dialed), then they are also counted as AUXOUTADJCALLS.</p> <p>Available for external calls on Generic 2.2 and Generic 3 switches.</p>	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
AUXOUTOFFTIME	<p>The talk time of all AUXOUTOFFCALLS (does not include HOLDTIME). This time is included in AUXOUTTIME.</p> <p>Available for external calls on Generic 2.2 and Generic 3 switches.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
AUXOUTTIME	<p>The talk time of all AUXOUTCALLS. AUXOUTTIME does not include HOLDTIME except on Generic 2.1 switches).</p> <p>AUXOUTTIME includes AUXOUTOFFTIME, AUXOUTOFFCALLS, and AUXADJCALLS.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
CONFERENCE	<p>The number of times the agent completed a conference; that is, pushed the conference key a second time.</p> <p>Available on Generic 2.2 and Generic 3 switches.</p>	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

Database Item	Description	Data Type	Column Type	Length
DA_ABNCALLS	The number of direct agent ACD calls that were abandoned by callers while in queue or ringing the agent's voice terminal. Includes calls considered abandoned because their talk time was less than the phantom abandon call timer. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_ABNTIME	The time DA_ABNCALLS were waiting in queue or ringing before being abandoned. Includes the time until the agent releases the call for phantom abandoned calls. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_ACDCALLS	The number of direct agent ACD calls that the agent answered. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling. DA_ACDCALLS includes DA_RELEASE .	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_ACDTIME	The talk time of all DA_ACDCALLS (does not include HOLDTIME). Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_ACWINCALLS	The number of inbound extension calls answered by the agent while in ACW for direct agent ACD calls. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

Database Item	Description	Data Type	Column Type	Length
DA_ACWINTIME	The talk time of all DA_ACWINCALLS (does not include HOLDTIME). Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_ACWOADJCALLS	The number of DA_ACWOCALLS that were placed by an ASAI adjunct on behalf of the agent (keyboard-dialed). If these calls were placed to off-switch destinations, they are also counted as DA_ACWOFFFCALLS . Available on Generic 3 switches with ASAI.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_ACWOCALLS	The number of outbound extension calls agents made while in ACW for direct agent ACD calls queued through this split/skill. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling. DA_ACWOCALLS includes DA_ACWOADJCALLS and DA_ACWOFFFCALLS .	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_ACWOFFFCALLS	The number of DA_ACWOCALLS that were made to an off-switch location. If these calls were placed by an adjunct on behalf of the agent (keyboard-dialed), they are also counted as DA_ACWOADJCALLS . Requires a Generic 3 switch with ASAI.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_ACWOFFFTIME	The talk time of all DA_ACWOFFFCALLS (does not include HOLDTIME). DA_ACWOFFFTIME is included in DA_ACWOTIME . Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
DA_ACWOTIME	The talk time of all DA_ACWOCALLS (does not include HOLDTIME). DA_ACWOTIME includes DA_ACWOOFFTIME . Available on Generic 3 switches with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_ACWTIME	The duration of ACW associated with DA_ACDCALLS , including time on DA_ACWINCALLS and DA_ACWOCALLS . Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling. DA_ACWTIME includes DA_ACWINTIME and DA_ACWOTIME .	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_ANSTIME	The time spent by callers in direct agent queue and ringing before being answered. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
DA_OTHERCALLS	The number of direct agent calls that were redirected to another destination before being answered; for example, by call pickup, coverage or Redirection on No Answer. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
DA_OTHERTIME	The time spent in queue or ringing by DA_OTHERCALLS before being redirected. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
DA_RELEASE	The number of direct agent ACD calls released or dropped by the agent before the far end released. Requires an ECS with the ASAI or EAS feature for direct agent calling.	C	integer	4 bytes
EVENT1-9	The number of times each event (stroke count) feature button (1 to 9) was pressed while the agent was on an ACD call or in call-related after call work. Available on Generic 2 and Generic 3 switches.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
EXTENSION	The extension number for which data were collected.	A	char(6)	6 byte ASCII text string
HOLDABNCALLS	The number of times callers abandoned from hold. For Generic 2.2 and Generic 3 switches, HOLDABNCALLS applies to all calls the agent put on hold. For Generic 2.1 switches, HOLDABNCALLS applies to split ACD calls held.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
HOLDACDTIME	The time split/skill and direct agent ACD calls spent on hold at the agent's voice terminal. This includes time on AUXIN or AUXOUT calls with the ACD calls on hold.	C	integer	4 bytes
HOLDCALLS	The number of calls that were placed on hold at least once. HOLDCALLS includes HOLDABNCALLS . For Generic 2.1 switches, HOLDCALLS applies to split ACD calls held. For Generic 3 and Generic 2.2 switches, HOLDCALLS applies to all calls the agent put on hold. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

Database Item	Description	Data Type	Column Type	Length
HOLDTIME	The time spent by callers on hold. For Generic 2.1 switches, HOLDTIME applies to split ACD calls held. For Generic 3 and Generic 2.2 switches, HOLDTIME applies to all calls the agent put on hold. HOLDTIME includes HOLDACD TIME.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACD_AUXINTIME	The time during the collection interval that the agent spent talking on AUXIN calls with at least one split/skill or direct agent ACD call on hold. For agents in multiple splits/skills, this time is recorded in the record in which SPLIT is OLDEST_LOGON . Available on Generic 2.2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACD_AUX_OUTTIME	The time during the collection interval that the agent spent dialing (Generic 2.2) and talking on AUXOUT calls with at least one split/skill or direct agent ACD call for this split/skill with the call on hold. Available on Generic 2.2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACD_OTHERTIME	The time during the collection interval that the agent spent in the OTHER state (dialing an outgoing call, with a ringing personal call [Generic 3 switches], or with calls on hold and with no other state selected) with at least one split/skill or direct agent ACD call on hold. Available on Generic 2.2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACDTIME	The time during the collection interval that the agent was talking on ACD calls for SPLIT or the time ACD calls spent on hold. I_ACDTIME includes time spent on O_ACD CALLS, but does not include HOLDTIME (except for Generic 2.1 switches).	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
I_ACWINTIME	The time during the collection interval that the agent was in ACW and on inbound extension calls. I_ACWINTIME includes ACW for split/skill ACD calls and ACW not associated with a call, but does not include the time inbound ACW calls spent on hold (except for Generic 2.1 switches).	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACWOUTTIME	The time during the collection interval that the agent was in ACW and on outbound extension calls. I_ACWOUTTIME includes ACW for split/skill ACD calls and ACW not associated with a call, but does not include the time ACWOUT calls spent on hold (except for Generic 2.1 switches).	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_ACWTIME	The time during the collection interval that the agent was in ACW . This includes ACW for split/skill ACD calls and ACW not associated with a call. NOTE: I_ACWINTIME and I_ACWOUTTIME include time in ACW for direct agent calls, but I_ACWTIME does not include this time. Therefore, the sum of I_ACWINTIME and I_ACWOUTTIME may be greater than I_ACWTIME . I_ACWTIME includes I_ACWINTIME and I_ACWOUTTIME .	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_AUXINTIME	The time during the collection interval that the agent was in AUX work, AVAILABLE ; or, for Generic 2.2 and Generic 3 switches, including time when an ACD or AUXIN/AUXOUT call is on hold and on inbound extension calls and SPLIT was the OLDEST_LOGON . I_AUXINTIME includes I_ACDAUXINTIME but does not include time calls spent on hold except for Generic 2.1 switches, which include time spent on hold unless the agent makes an outgoing call with an AUXIN call on hold.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
I_AUXOUTTIME	The time during the collection interval that the agent was in AUX work, AVAILABLE, or, for Generic 2.2 and Generic 3 switches, including time when an ACD or AUXIN/AUXOUT call on hold and on outbound extension calls. In the cases where the agent was in AUX work, AVAILABLE or had an AUXIN/AUXOUT call on hold, the AUXOUT time and calls are recorded for the SPLIT that is the OLDEST_LOGON . In cases where the agent had an ACD call on hold, SPLIT is the split or skill associated with the last ACD call put on hold. I_AUXOUTTIME includes I_ACDAUX_OUTTIME , but does not include time calls spent on hold (except for Generic 2.1 switches).	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_AUXTIME	The time the agent spent in AUX work in SPLIT. When an agent is in AUX work in multiple splits/skills, this time is recorded in each split or skill in which the agent is in AUX. I_AUXTIME includes I_AUXINTIME and I_AUXOUTTIME .	C	integer	4 bytes
I_AVAILTIME	The time during the collection interval that the agent was available for ACD calls in this split/skill.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_DA_ACDTIME	The time during the collection interval that the agent spent talking on direct agent calls. Does not include HOLDTIME . Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
I_DA_ACWTIME	The time during the collection interval that the agent was doing after call work associated with direct agent ACD calls. Requires a Generic 3 switch with the ASAI or EAS feature for direct agent calling.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_OTHERTIME	<p>The time during the collection interval that POSITIONS were doing other work. I_OTHERTIME is collected for the time period after the link to the switch comes up or after the agent logs in and before the CMS receives notification of the agent's state from the switch.</p> <p>For Generic 3 switches, other work includes: while in Auto-In or Manual-In mode, an agent put any call on hold and performed no further action; the agent had a direct agent call ringing, was on a direct agent call or in ACW for a direct agent call; the agent dialed to place a call or activate a feature; or an extension-in call rang at the agent's voice terminal with no other activity.</p> <p>For Generic 2.2, agents were in AUTO-IN or MANUAL-IN, put a call on hold using the Hold button or switchhook flash and performed no further action.</p> <p>For Generic 2.2 with EAS and Generic 3 switches, the other work includes the time agents were logged into multiple splits/skills and doing work for a split/skill other than this one (with an ACD call ringing, talking on an ACD call, or in ACW for a split/skill other than this one).</p> <p>For Generic 3 switches with EAS and multiple call handling, agents are available in other, multiple call handling skills, but not in this skill.</p> <p>I_OTHERTIME includes I_ACDOTHERTIME, I_DA_ACDTIME, and I_DA_ACWTIME.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
I_RINGTIME	The time during the collection interval that the agent had split/skill and direct agent ACD calls ringing. If the agent changes work modes or makes/receives another call instead of answering the ringing call, I_RINGTIME will stop accumulating. RINGTIME is the time the caller spends ringing and is independent of agent activity. Available on a Generic 2 and Generic 3 switch for ring tracking.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
I_STAFFTIME	The time during the collection interval that the agent was staffed (logged in) in this split/skill. I_STAFFTIME includes I_AUXTIME , I_AVAILABLE , I_ACDTIME , I_ACWTIME , I_DA_ACDTIME , I_DA_ACWTIME , I_OTHERTIME , and I_RINGTIME .	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
INCOMPLETE	This indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month. Changing split/skill or VDN call profile data while data collection is active only affects the respective split/skill or VDN data.	C	smallint	2 bytes

Database Item	Description	Data Type	Column Type	Length
INTRVL	The number of minutes in the intrahour interval (15, 30, or 60). INTRVL applies to intrahour intervals only.	A	smallint; only in hagent table	2 bytes
LOGID (index)	The Login ID that was used to staff the EXTENSION . Agents in multiple splits/skills have one LOGID .	A	char(10)	10 byte ASCII text string
NOANSREDIR	The number of split/skill and direct agent ACD calls that rang at this agent's voice terminal and then were automatically redirected by the Redirection on No Answer feature because they were not answered. Split/skill ACD calls are queued to the split/skill or VDN; direct agent ACD calls are redirected to the agent's coverage path. Redirection On No Answer to a split/skill is available on <i>DEFINITY</i> ECS or Generic 3 Version 2 or later switches. Redirection On No Answer to a VDN is only available on <i>DEFINITY</i> ECS.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
O_ACDCALLS	The number of ACDCALLS and DA_ACDCALLS that were placed by an adjunct (predictive dialing). Available for outbound calling on a Generic 2.2 switch with the ASAI Gateway Interface feature and on a Generic 3 switch with the ASAI feature.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
O_ACDTIME	The talk time of all O_ACDCALLS (does not include time calls spent on hold). This time is included in ACDTIME . Available for outbound calling on Generic 2.2 switches with the ASAI Gateway Interface feature and on Generic 3 switches with the ASAI feature.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
O_ACWTIME	The duration of all ACW associated with O_ACDCALLS . O_ACWTIME is included in ACWTIME . Available for outbound calling on Generic 2.2 switches with the ASAI Gateway Interface feature and on Generic 3 switches with the ASAI feature.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
PHANTOMABNS	The number of ACD calls with talk time less than the value of the phantom-abandoned call timer. Available on Generic 3 switches.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
RINGCALLS	The number of split/skill (Generic 2 and Generic 3 switches) and direct agent ACD calls (Generic 3 switches) that rang at the agent's position. RINGCALLS includes NOANSREDIR . Available on Generic 2 and Generic 3 switches for ring tracking.	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)
RINGTIME	The time split/skill and direct agent ACD calls spent ringing at the agent's position (independent of disposition or other agent activity). RINGTIME is the time the caller spends ringing and is independent of agent activity. I_RINGTIME is the time the agent spends in the ringing state and is affected by other agent activity. RINGTIME includes ANSRINGTIME . Available on Generic 2 and Generic 3 switches.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
ROW_DATE (index)	The day for which the data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
SPLIT (index)	The split number to which the EXTENSION is assigned to or the skill number that the agent logged into.	I	smallint	2 bytes

Database Item	Description	Data Type	Column Type	Length
STARTTIME	The start time for the interval for which data was collected. STARTTIME applies only to the Interval table.	I	smallint; only in hagent table	2 bytes
TI_AUXTIME	TI_AUXTIME includes TI_AUXTIME0 , TI_AUXTIME1-9 , I_AUXINTIME , and I_AUXOUTTIME . sum(TI_AUXTIME) = sum(TI_AUXTIME0 + TI_AUXTIME1 + TI_AUXTIME2 + TI_AUXTIME3 + TI_AUXTIME4 + TI_AUXTIME5 + TI_AUXTIME6 + TI_AUXTIME7 + TI_AUXTIME8 + TI_AUXTIME9) , over all splits/skills the agent was logged into. Requires an ECS with the EAS feature.	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
TI_AUXTIME0	The time the agent spent in AUX with reason code 0 (zero). This is time in “system” AUX for switches with AUX reason codes active. It is the same as TI_AUXTIME for switches without AUX reason codes active. “TI_” time is only stored for the skill logged in to the longest. “TI_” time needs to be summed across the skills the agents may log in to, in case the login order changes during the collection interval. Requires an ECS with the EAS feature.	C	integer	4 bytes
TI_AUXTIME1-9	The time the agent spent in AUX with reason codes 1-9. “TI_” time is only stored for the skill logged in to the longest. “TI_” time needs to be summed across the skills the agents may log in to, in case the login order changes during the collection interval. Requires an ECS with the EAS feature.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
TI_AVAILTIME	<p>The time during the collection interval that the agent was in the available state for split/skill or direct agent ACD calls in any split/skill. TI_AVAILTIME is recorded for the split/skill that was the OLDEST_LOGON.</p> <p>For a non-EAS operation, if an agent logged into multiple splits and is in AUX mode in one split and is available for ACD calls in another split, the agent will accrue I_AVAILTIME for the split in which the agent is available and TI_AVAILTIME in the split logged into the longest.</p> <p>NOTE: "TI_" time is only stored for the split/skill the agent has been logged into the longest. "TI_" time needs to be summed across the splits/skills the agents may log in to, in case the logon order changes during the collection interval.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
TI_OTHERTIME	<p>The time during the collection interval that the agent was doing other work in all splits/skills.</p> <p>For all switches, TI_OTHERTIME is collected for the time period after the link to the switch comes up or after the agent logs in and before the <i>CentreVu</i> CMS receives notification of the agent's state from the switch.</p> <p>For Generic 3 switches, other work includes: while in AUTO-IN or MANUAL-IN mode, the agent put any call on hold and performed no further action, the agent dialed to place a call or to activate a feature, or an extension call rang with no other activity.</p> <p>For Generic 2.2 switches, the agent pushed the Hold button or flashed the switchhook from auto-in or manual-in mode and performed no further action.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table

Database Item	Description	Data Type	Column Type	Length
TI_OTHERTIME (contd)	<p>NOTE: "TI_" time is only stored for the split/skill the agent has been logged into the longest. "TI_" time needs to be summed across the splits/skills the agents may log in to, in case the logon order changes during the collection interval.</p> <p>TI_OTHERTIME includes I_ACDOTHERTIME.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
TI_STAFFTIME	<p>The time during the collection interval that the agent was staffed in any split/skill. "TI_" time is only stored for the split/skill logged into the longest. "TI_" time needs to be summed across the splits/skills the agents may log in to, in case the login order changes during the collection interval.</p> <p>sum(TI_STAFFTIME) = sum(I_ACDTIME + I_ACWTIME + I_DA_ACDTIME + I_DA_ACWTIME + I_RINGTIME + TI_AUXTIME + TI_AVAILTIME + TI_OTHERTIME), over all splits/skills the agent was logged into.</p>	C	integer; smallint in hagent table	4 bytes; 2 bytes in hagent table
TRANSFERRED	<p>Number of calls the agent transferred to another destination. Note that TRANSFERRED calls include both inbound and outbound calls. Therefore, OTHERCALLS and O_OTHERCALLS may each include some SHORTCALLS. For Generic 2.1 switches, includes transfers to measured VDNs or splits/skills. For Generic 2.2 and Generic 3 switches, this includes transferring all calls.</p>	C	smallint (dagent, hagent); integer (magent, wagent)	2 bytes (dagent, hagent); 4 bytes (magent, wagent)

ECS Cross-Reference

The following table lists which ECS releases support each Agent database item.

The following is a key to the database items tables:

- Items marked “X” indicate that the database item is supported by the specified ECS release.
- Items marked “EAS” require that the EAS feature be active on the ECS for the items to be populated.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ABNCALLS	X	X	X	X	X	X	X
ABNTIME	X	X	X	X	X	X	X
ACD	X	X	X	X	X	X	X
ACD_RELEASE			X	X	X	X	X
ACDAUXOUTCALLS		X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
ACDTIME	X	X	X	X	X	X	X
ACWINCALLS	X	X	X	X	X	X	X
ACWINTIME	X	X	X	X	X	X	X
ACWOUTADJCALLS		X	X	X	X	X	X
ACWOUTCALLS	X	X	X	X	X	X	X
ACWOUTOFFCALLS		X	X	X	X	X	X
ACWOUTOFFTIME		X	X	X	X	X	X
ACWOUTTIME	X	X	X	X	X	X	X
ACWTIME	X	X	X	X	X	X	X
ANSRINGTIME	X	X	X	X	X	X	X
ASSISTS	X	X	X	X	X	X	X
AUXINCALLS	X	X	X	X	X	X	X
AUXINTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
AUXOUTADJCALLS		X	X	X	X	X	X
AUXOUTCALLS	X	X	X	X	X	X	X
AUXOUTOFFCALLS		X	X	X	X	X	X
AUXOUTOFFTIME		X	X	X	X	X	X
AUXOUTTIME	X	X	X	X	X	X	X
CONFERENCE		X	X	X	X	X	X
DA_ABNCALLS			X	X	X	X	X
DA_ABNTIME			X	X	X	X	X
DA_ACDCALLS			X	X	X	X	X
DA_ACDDTIME			X	X	X	X	X
DA_ACWINCALLS			X	X	X	X	X
DA_ACWINTIME			X	X	X	X	X
DA_ACWOADJCALLS			X	X	X	X	X
DA_ACWOCALLS			X	X	X	X	X
DA_ACWOFFCALLS			X	X	X	X	X
DA_ACWOFFTIME			X	X	X	X	X
DA_ACWOTIME			X	X	X	X	X
DA_ACWTIME			X	X	X	X	X
DA_ANSTIME			X	X	X	X	X
DA_OTHERCALLS			X	X	X	X	X
DA_OTHERTIME			X	X	X	X	X
DA_RELEASE			X	X	X	X	
EVENT1-9	X	X	X	X	X	X	X
EXTENSION	X	X	X	X	X	X	X
HOLDABNCALLS	X	X	X	X	X	X	X
HOLDACDDTIME	X	X	X	X	X	X	X
HOLDCALLS	X	X	X	X	X	X	X
HOLDTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
I_ACDAUXINTIME		X	X	X	X	X	X
I_ACDAUX_OUTTIME		X	X	X	X	X	X
I_ACDOTHERTIME		X	X	X	X	X	X
I_ACDTIME	X	X	X	X	X	X	X
I_ACWINTIME	X	X	X	X	X	X	X
I_ACWOUTTIME	X	X	X	X	X	X	X
I_ACWTIME	X	X	X	X	X	X	X
I_AUXINTIME	X	X	X	X	X	X	X
I_AUXOUTTIME	X	X	X	X	X	X	X
I_AUXTIME	X	X	X	X	X	X	X
I_AVAILTIME	X	X	X	X	X	X	X
I_DA_ACDTIME			X	X	X	X	X
I_DA_ACWTIME			X	X	X	X	X
I_OTHERTIME	X	X	X	X	X	X	X
I_RINGTIME	X	X	X	X	X	X	X
I_STAFFTIME	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
LOGID	X	X	X	X	X	X	X
NOANSREDIR				X	X	X	X
O_ACDCALLS		X	X	X	X	X	X
O_ACDTIME		X	X	X	X	X	X
O_ACWTIME		X	X	X	X	X	X
PHANTOMABNS			X	X	X	X	X
RINGCALLS	X	X	X	X	X	X	X
RINGTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X

Agent Database Items

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
STARTTIME	X	X	X	X	X	X	X
TI_AUXTIME	X	X	X	X	X	X	X
TI_AUXTIME0	X	X	X	X	X	X	X
TI_AUXTIME1-9						EAS	X
TI_AVAILTIME	X	X	X	X	X	X	X
TI_OTHERTIME	X	X	X	X	X	X	X
TI_STAFFTIME	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X

Trunk Group Database Items

Overview

The Trunk Group database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical trunk group database items apply to the Intrahour Trunk Group (*htkgrp*), Daily Trunk Group (*dtkgrp*), Weekly Trunk Group (*wtkgrp*), and Monthly Trunk Group (*mtkgrp*) tables. All items listed in the following table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four trunk group tables are also noted in the Column Type column. The historical indexes are **ROW_DATE** and **TKGRP**.

Trunk Group Database Items

The following table describes the data items in the CMS Trunk Group database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	<p>The number of calls carried by this trunk that were abandoned by the caller before being answered by an agent. Calls directly to unmeasured stations that did not go through a measured VDN or split/skill are not recorded.</p> <p>For Generic 2.2 and Generic 3 switches, ABNCALLS includes all calls abandoned by the caller that were carried by this trunk, except for calls directly to unmeasured stations that did not go through a measured VDN or split/skill. This includes ACD calls and calls that routed to an agent or extension with talk times less than the phantom-abandoned call timer value.</p> <p>For Generic 2.1 switches, this is ACD calls that abandon from the split queue or from ringing, and calls that abandon from vector processing. Calls that abandon while listening to a forced disconnect are also included in ABNCALLS.</p> <p>For Generic 3 Version 1 switches, ABNCALLS includes all calls abandoned by the caller that were carried by this trunk, except for calls that direct to unmeasured stations that did not go through a VDN or split/skill. It also includes calls that abandon while listening to a forced disconnect. ABNCALLS includes ABNVECCALLS, ABNQUEUECALLS, and ABNRINGCALLS.</p>	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
ABNQUEECALLS	<p>The number of ABNCALLS that abandoned while in a split/skill or direct agent ACD queue.</p> <p>Available on Generic 2 and Generic 3 switches.</p>	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
ABNRINGCALLS	<p>The number of split/skill or direct agent ABNCALLS that abandoned by the caller while ringing at an agent position.</p> <p>Available on Generic 2 and Generic 3 switches.</p>	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
ABNVECCALLS	The number of ABNCALLS that abandoned while in vector processing. This includes vector calls that abandoned while in queue or while ringing at an agent position. Available on Generic 2 and Generic 3 switches with vectoring. ABNVECCALLS includes ABNQUEECALLS and ABNRINGCALLS .	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
ACD (index)	ACD number for which data was collected	I	smallint	2 bytes
ACDCALLS	The number of INCALLS that were answered by an agent as a split/skill or direct agent ACD call. ACDCALLS includes BACKUPCALLS .	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
ALLINUSETIME	The length of time during the interval that all trunks in the trunk group are in use (on calls or maintenance busy).	C	integer	4 bytes
AUDIO	The number of calls for which audio difficulty problems were reported for a trunk or for trunks in this trunk group. Available on Generic 2 and Generic 3 switches.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BACKUPCALLS	The number of ACDCALLS that were delivered to and answered by this split/skill by a vector command other than "queue to main". This includes calls delivered by "messaging split/skill", "check backup", and "route to split/skill" vector commands, direct agent calls, and redirect on no answer routing. Calls answered in a main split/skill can be calculated as ACDCALLS - BACKUPCALLS . Available on Generic 2 and Generic 3 switches with vectoring. NOTE: The Redirect on No Answer to VDN routing feature is available on the <i>DEFINITY</i> ECS.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table

Database Item	Description	Data Type	Column Type	Length
BH_ABNCALLS	The number of incoming calls carried by the trunk group that abandoned during the busy hour.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_ACDCALLS	The number of incoming calls carried by this trunk group during the busy hour that were answered by an agent as split/skill or direct agent ACD calls.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_ALLINUSETIME	The length of time during the busy hour that all trunks in the trunk group were in use.	B	integer	4 bytes
BH_BUSYCALLS	The number of incoming calls carried by the trunk group during the busy hour that were given a busy signal by the switch.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_DISCCALLS	The number of incoming calls carried by the trunk group during the busy hour that were forced to disconnect by the switch.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_INCALLS	The number of incoming calls carried by this trunk group that completed during the busy hour. BH_INCALLS includes BH_ABNCALLS , BH_ACDCALLS , and BH_OTHERCALLS .	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_INTERVAL	The defined interval length, with the start time, of the busy hour.	B	integer; only in dtkgrp table	4 bytes
BH_INTIME	The trunk holding time of all incoming calls carried by this trunk group that completed during the busy hour.	B	integer	4 bytes
BH_OABNCALLS	The number of outgoing adjunct-originated calls carried by the trunk group that abandoned during the busy hour. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table

Database Item	Description	Data Type	Column Type	Length
BH_OACDCALLS	The number of outgoing adjunct-originated ACD calls carried by the trunk group and answered by an agent as split/skill or direct agent ACD calls that completed during the busy hour. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_OOTHERCALLS	The number of outgoing calls carried by the trunk group during the busy hour that were not answered or abandoned as ACD calls. BH_OOTHERCALLS include extension out calls, outbound call management calls forced busy or forced disconnect, short outgoing calls, and outgoing calls with unknown disposition.	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_OTHERCALLS	The number of incoming calls carried by the trunk group during the busy hour that were not answered or abandoned. BH_OTHERCALLS include extension in calls, calls forced busy or disconnected, calls that outflowed off the switch, short inbound calls, and inbound calls of unknown disposition. BH_OTHERCALLS includes BH_BUSYCALLS and BH_DISCCALLS .	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_OUTCALLS	The number of outgoing calls carried by the trunk group that completed during the busy hour. BH_OUTCALLS includes BH_OABNCALLS , BH_OACDCALLS , and BH_OOTHERCALLS .	B	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BH_OUTTIME	The trunk holding time of all outgoing calls carried by the trunk group that completed during the busy hour.	B	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
BH_STARTTIME	The starting time of the hour for which busy hour data was collected. The busy hour is that set of contiguous intervals during the day totaling an hour in which the trunk holding time for the trunk group was a maximum.	B	integer	4 bytes
BLOCKAGE	The number of outbound call attempts that were blocked because all trunks were busy. Available on Generic 2 switches.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
BUSYCALLS	The number of INCALLS that were given a busy signal by the switch. This can occur on all switches via the "busy" vector command. On Generic 3 switches without vectoring, BUSYCALLS can occur if a call is routed to a split/skill with coverage set to "yes" where there are no agents available, the queue is full (or there is no queue), there is no coverage, and an announcement has played or the trunk is not a CO trunk. Also on Generic 3 switches, BUSYCALLS can occur if a call is routed to a direct agent with coverage set to "yes", the agent is not logged in and there is no coverage path administered, and an announcement has played or the trunk is not a CO trunk. BUSYCALLS can occur on Generic 3 switches without vectoring when a split queue is full or there are no queue slots, no busy coverage is administered and an announcement has played or the trunk is not a CO trunk.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
COMPLETED	The number of OUTCALLS that were completed (far end answered). Available on Generic 3 switches.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
CONNECTCALLS	The number of INCALLS that were answered at a station and were not split/skill or direct agent ACD calls.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
DISCCALLS	<p>With Generic 2.1, Generic 3 Version 1 and Generic 3 Version 2 (prior to load 100) switches, DISCCALLS is the number of INCALLS that were given a forced disconnect announcement by the “disconnect” vector command, listened to the entire announcement, then were disconnected by the switch.</p> <p>With Generic 2.2 switches, this is the number of INCALLS that were disconnected by the switch by the “disconnect” vector command.</p> <p>With Generic 3 Version 2 and later Generic 3 switches, this is the number of INCALLS that were disconnected by the switch by the “disconnect” vector command. DISCCALLS also includes calls that were disconnected by the switch when the vector disconnect timer expired or that reached the end of vector processing without being queued. DISCCALLS includes VDISCCALLS.</p>	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
FAILURES	<p>The number of trunk failures for this TKGRP. No time or call is recorded in any of the CMS tables. Trunk failures can be due to hardware problems on the trunk, incompatible trunk types on either end of a call, or internal switch errors (such as errors in call processing or vectoring translations).</p> <p>This item does not include calls with short holding times.</p> <p>Available on Generic 2 switches. The FAILURES database item is not populated for Generic 3 switches because trunks that fail are automatically placed in the maintenance busy state.</p>	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
I_INOCC	<p>The total time during the collection interval that ALL trunks in the trunk group were occupied by incoming calls. If an incoming call on a measured trunk is transferred off the switch, the incoming trunk remains in use for the call and accrues trunk holding time until the caller drops or the call is released.</p>	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
I_OUTOCC	The time during the collection interval that trunks in this trunk group were occupied by outgoing calls.	C	integer	4 bytes
INCALLS	The number of inbound calls that were carried by this TKGRP and that completed during the collection interval. INCALLS includes ABNCALLS , ACDCALLS , CONNECTCALLS , and TRANSFERRED . INCALLS = ACDCALLS + ABNCALLS + OTHERCALLS .	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
INCOMPLETE	This indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month.	C	smallint	2 bytes
INTIME	The trunk holding time for all INCALLS carried by trunks in this trunk group that completed during the collection interval. Trunk holding time is the time from the initial trunk seizure until the trunk goes idle (that is, until the caller drops, the agent releases the call, or the switch disconnects the call). If an incoming call on a measured trunk is transferred off the switch, the incoming trunk remains in use for the call and accrues trunk holding time until the caller drops or the call is released.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
INTRVL	The number of minutes in the intrahour interval (15, 30, or 60). INTRVL applies to intrahour tables only.	A	smallint; only in htkgrp table	2 bytes
MBUSYTIME	The total time (in seconds) during the collection interval that trunks in the trunk group were maintenance busy.	C	integer	4 bytes
O_ABNCALLS	The number of OUTCALLS on this trunk group that were offered by an adjunct as split/skill or direct agent ACD calls and were answered then abandoned by the far end. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
O_ACDCALLS	The number of OUTCALLS from this trunk group that were offered by an adjunct to one or more splits/skills and were answered by an agent. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
O_OTHERCALLS	The number of OUTCALLS on this trunk group that were not answered or abandoned as ACD split/skill calls. These include extension out calls, calls forced busy and forced disconnected, short outgoing calls, and calls with unknown dispositions. O_OTHERCALLS includes SHORTCALLS .	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table

Database Item	Description	Data Type	Column Type	Length
OTHERCALLS	The number of INCALLS carried by this trunk group that were not answered as split/skill or direct agent ACD calls or abandoned. These include forced busy calls, forced disconnect calls, calls that were connected to a non-ACD destination, short inbound calls, calls that outflowed off the switch, and calls with unknown dispositions. OTHERCALLS includes BUSYCALLS , DISCCALLS , SHORTCALLS , and CONNECTCALLS . OTHERCALLS = INCALLS - ACDCALLS - ABNCALLS	C	Integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
OUTCALLS	The number of outbound calls that were carried by this TKGRP and were completed during the collection interval. OUTCALLS includes COMPLETED , O_ABNCALLS , O_ACDCALLS , O_OTHERCALLS , TRANSFERRED , and SHORTCALLS . OUTCALLS = O_ACDCALLS + O_ABNCALLS + O_OTHERCALLS .	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
OUTTIME	The trunk holding time for all OUTCALLS carried by trunks in this trunk group that completed during the collection interval. Trunk holding time is the time from the initial trunk seizure until the trunk goes idle (that is, until the far end drops, the agent releases the call, or the switch disconnects the call). OUTTIME includes SETUPTIME .	C	integer	4 bytes
ROW_DATE (index)	The day for which data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
SETUPTIME	Amount of time from trunk seizure until OUTCALLS completed at the far end. Available on Generic 3 switches.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
SHORTCALLS	The number of inbound and outbound calls that occupied a trunk in the trunk group for less than 2 seconds and that did not queue to a split/skill, forward to a split/skill, get answered by an agent, get a forced busy or forced disconnect from the switch, or produce a trunk failure or maintenance busy. Note that SHORTCALLS includes both inbound and outbound calls. Therefore, OTHERCALLS and O_OTHERCALLS may each include some SHORTCALLS .	C	integer	4 bytes
SPLIT	The split/skill to which this TKGRP terminates.	A	smallint	2 bytes
TKGRP (index)	The trunk group number for which data was collected. This will be zero if the trunk group carrying the call is not measured.	I	smallint	2 bytes
TRANSFERRED	The number of calls that the agent transferred to another destination. Note that TRANSFERRED includes both inbound and outbound calls. Therefore, OTHERCALLS and O_OTHERCALLS may each include some TRANSFERRED . For Generic 2.1 switches, TRANSFERRED includes transfers to measured VDNs or splits/skills. For Generic 3 and Generic 2.2 switches, TRANSFERRED includes all calls that transferred.	C	integer; smallint in htkgrp table	4 bytes; 2 bytes in htkgrp table
TRUNKS	The current number of trunks assigned to this TKGRP .	A	smallint	2 bytes
VDN	The VDN to which the TKGRP terminates. Available on Generic 2 and Generic 3 switches with vectoring.	A	char(6)	6 byte ASCII text string
VECTOR	The vector to which this trunk group's VDN terminates. Available on Generic 2 and Generic 3 switches with vectoring.	A	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each of the Trunk Group database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ABNCALLS	X	X	X	X	X	X	X
ABNQUECALLS	X	X	X	X	X	X	X
ABNRINGCALLS	X	X	X	X	X	X	X
ABNVECCALLS	X	X	X	X	X	X	X
ACD (index)	X	X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
ALLINUSETIME	X	X	X	X	X	X	X
AUDIO	X	X	X	X	X	X	X
BH_ABNCALLS	X	X	X	X	X	X	X
BH_ACDCALLS	X	X	X	X	X	X	X
BH_ALLINUSETIME	X	X	X	X	X	X	X
BH_BUSYCALLS	X	X	X	X	X	X	X
BH_DISCCALLS	X	X	X	X	X	X	X
BH_INCALLS	X	X	X	X	X	X	X
BH_INTERVAL	X	X	X	X	X	X	X
BH_INTIME	X	X	X	X	X	X	X
BH_OABNCALLS		X	X	X	X	X	X
BH_OACDCALLS		X	X	X	X	X	X
BH_OOTHERCALLS	X	X	X	X	X	X	X
BH_OTHERCALLS	X	X	X	X	X	X	X
BH_OUTCALLS	X	X	X	X	X	X	X
BH_OUTTIME	X	X	X	X	X	X	X
BH_STARTTIME	X	X	X	X	X	X	X
BACKUPCALLS	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
BLOCKAGE	X	X					X
BUSYCALLS	X	X	X	X	X	X	X
COMPLETED			X	X	X	X	X
CONNECTCALLS	X	X	X	X	X	X	X
DISCCALLS	X	X	X	X	X	X	X
FAILURES	X	X					X
I_INOCC	X	X	X	X	X	X	X
I_OUTOCC	X	X	X	X	X	X	X
INCALLS	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INTIME	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
MBUSYTIME	X	X	X	X	X	X	X
O_ABNCALLS		X	X	X	X	X	X
O_ACDCALLS		X	X	X	X	X	X
O_OTHERCALLS	X	X	X	X	X	X	X
OUTCALLS	X	X	X	X	X	X	X
OTHERCALLS	X	X	X	X	X	X	X
OUTTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SETUPTIME			X	X	X	X	X
SHORTCALLS	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X
TKGRP	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X
TRUNKS	X	X	X	X	X	X	X
VDN	X	X	X	X	X	X	X
VECTOR	X	X	X	X	X	X	X

Trunk Database Items

Overview

The Trunk database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items typically apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical trunk database items apply to the Intrahour Trunk (*htrunk*), Daily Trunk (*dtrunk*), Weekly Trunk Group (*wtrunk*), and Monthly Trunk (*mttrunk*) tables. All items listed in the following Trunk Database Items table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four trunk tables are also noted in the Column Type column. The historical indexes are **EQLOC**, **ROW_DATE** and **TKGRP**.

Trunk Database Items

The following table describes the data items in the CMS Trunk database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	<p>The number of calls carried by this trunk that were abandoned by the caller before being answered by an agent. Calls directly to unmeasured stations that did not go through a measured VDN or split/skill are not recorded. For Generic 2.2 and Generic 3 switches, ABNCALLS includes all calls abandoned by the caller that were carried by this trunk, except for calls directly to unmeasured stations that did not go through a measured VDN or split/skill. This includes ACD calls and calls that routed to an agent or extension with talk times less than the phantom-abandoned call timer value. For Generic 2.1 switches, this is ACD calls that abandon from the split queue or from ringing, and calls that abandon from vector processing. Calls that abandon while listening to a forced disconnect are also included in ABNCALLS. For Generic 3 Version 1 switches, ABNCALLS includes all calls abandoned by the caller that were carried by this trunk, except for calls that directed to unmeasured stations that did not go through a VDN or split/skill. It also includes calls that abandon while listening to a forced disconnect.</p> <p>ABNCALLS includes ABNVECCALLS, ABNQUEUECALLS, and ABNRINGCALLS.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
ACD (index)	The ACD number for which data was collected.	I	smallint	2 bytes
ACDCALLS	The number of INCALLS that were answered by an agent as a split/skill or direct agent ACD call.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
AUDIO	The number of calls for which audio difficulty problems were reported for this trunk. Available on Generic 2 and Generic 3 switches.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
EQLOC (index)	The physical equipment location (trunk number) for which data was collected.	A	char(9)	9 byte ASCII text string

Database Item	Description	Data Type	Column Type	Length
FAILURES	<p>The number of trunk failures for this trunk. No time or call is recorded in any of the CMS tables. Trunk failures can be due to hardware problems on the trunk, incompatible trunk types on either end of a call, or to internal switch errors (such as errors in call processing or vectoring translations).</p> <p>This item does not include calls with short holding times.</p> <p>Available on Generic 2 switches. The FAILURES database item is not populated for Generic 3 switches because trunks that fail are automatically placed in the maintenance busy state.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
I_INOCC	<p>The total time during the collection interval that the trunk was occupied by incoming calls. If an incoming call on a measured trunk is transferred off the switch, the incoming trunk remains in use for the call and accrues trunk holding time until the caller drops or the call is released.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
I_OUTOCC	<p>The total time during the collection interval that this trunk was occupied by outbound calls.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
INCALLS	<p>The number of inbound calls carried by this trunk that completed during the collection interval. This includes calls with short holding times (SHORTCALLS) but does not include calls that had a trunk failure (FAILURES).</p> <p>INCALLS = ABNCALLS + ACDCALLS + OTHERCALLS.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table

Database Item	Description	Data Type	Column Type	Length
INCOMPLETE	<p>This indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month.</p> <p>Changing split/skill or VDN call profile data while data collection is active only affects the respective split/skill or VDN data.</p>	C	smallint	2 bytes
INTIME	<p>The trunk holding time for all INCALLS carried by this trunk that completed during the collection interval.</p> <p>Trunk holding time is the time from the initial trunk seizure until the trunk goes idle (that is, until the caller drops, the agent releases the call, or the switch disconnects the call). If an incoming call on a measured trunk is transferred off the switch, the incoming trunk remains in use for the call and accrues trunk holding time until the caller drops or the call is released.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
INTRVL	<p>The number of minutes in the intrahour interval (15, 30, or 60).</p> <p>INTRVL applies to intrahour intervals only.</p>	A	smallint; only in htrunk table	2 bytes
MBUSYTIME	<p>The total time during the collection interval that this trunk was maintenance busy.</p>	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table

Database Item	Description	Data Type	Column Type	Length
O_ABNCALLS	The number of OUTCALLS on this trunk that were offered by an adjunct as split/skill or direct agent ACD calls and were answered then abandoned by the far end before talking to an agent. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
O_ACDCALLS	The number of OUTCALLS from this trunk that were offered by an adjunct as split/skill or direct agent ACD calls and were answered by an agent. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
O_OTHERCALLS	The number of OUTCALLS on this trunk that were not answered as ACD split/skill calls or abandoned. These include extension out calls, forced busy calls, short outgoing calls, and calls with unknown dispositions. O_OTHERCALLS includes SHORTCALLS .	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
OUTCALLS	The number of outbound calls that were carried by the trunk and were completed during the collection interval. OUTCALLS = O_ACDCALLS + O_ABNCALLS + O_OTHERCALLS.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
OTHERCALLS	Number of INCALLS on this trunk that were not answered or abandoned as split/skill or direct agent ACD calls. These include forced busy calls, forced disconnect calls, calls that outflowed off the switch, calls that were connected to a non- ACD destination, short inbound calls, and calls with unknown dispositions.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table

Database Item	Description	Data Type	Column Type	Length
OUTTIME	The trunk holding time for all OUTCALLS carried by this trunk that completed during the collection interval. Trunk holding time is the time from the initial trunk seizure until the trunk goes idle (that is, until the far end drops, the agent releases the call, or the ECS disconnects the call). The format is the number of seconds that have accumulated.	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
ROW_DATE (index)	The day for which data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
SHORTCALLS	The number of inbound and outbound calls that occupied a trunk for less than 2 seconds and that did not queue to a split/skill, forward to a split/skill, get answered by an agent, get a forced busy or forced disconnect from the switch, or produce a trunk failure or maintenance busy. Note that SHORTCALLS includes both inbound and outbound calls. Therefore, OTHERCALLS and O_OTHERCALLS may each include some SHORTCALLS .	C	integer; smallint in htrunk table	4 bytes; 2 bytes in htrunk table
TKGRP (index)	The trunk group number to which the trunk is assigned.	A	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each Trunk database item.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ABNCALLS	X	X	X	X	X	X	X
ACD	X	X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
AUDIO	X	X	X	X	X	X	X
EQLOC	X	X	X	X	X	X	X
FAILURES	X	X					X
I_INOCC	X	X	X	X	X	X	X
I_OUTOCC	X	X	X	X	X	X	X
INCALLS	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INTIME	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
ITN	X	X	X	X	X	X	X
MBUSYTIME	X	X	X	X	X	X	X
O_ABNCALLS		X	X	X	X	X	X
O_ACDCALLS		X	X	X	X	X	X
O_OTHERCALLS	X	X	X	X	X	X	X
OUTCALLS	X	X	X	X	X	X	X
OTHERCALLS	X	X	X	X	X	X	X
OUTTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SHORTCALLS	X	X	X	X	X	X	X
TKGRP	X	X	X	X	X	X	X

Vector Database Items

Overview

⇒ NOTE:

Vector database items are available only if the Vectoring feature has been purchased and authorized for you to use.

The Vector database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical vector database items apply to the Intrahour Vector (`hvector`), Daily Vector (`dvector`), Weekly Vector (`wvector`), and Monthly Vector (`mvector`) tables. All items listed in the following Vector Database Items table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four vector tables are also noted in the Column Type column. The historical indexes are **ROW_DATE** and **VECTOR**.

Vector Database Items

The following table describes the data items in the CMS Vector database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	The number of INCALLS that were abandoned while INPROGRESS for this vector. This includes split/skill and direct agent ACD calls that abandon from queue or from ringing, calls that abandon from vector processing, and for the Generic 2.1 and Generic 3 Version 1 switches, calls that abandoned while listening to a forced disconnect announcement. ABNCALLS includes ABNQUECALLS , ABNRINGCALLS , and PHANTOMABNS .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
ABNQUECALLS	The number of ABNCALLS that hung up while in a split/skill or direct agent ACD queue. Available on Generic 2 and Generic 3 switches.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
ABNRINGCALLS	The number of split/skill or direct agent ABNCALLS that were abandoned while ringing at an agent position. Available on Generic 2 and Generic 3 switches and on the ECS.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
ABNTIME	The time caller waited while vector steps were executed, the call was queued, and ringing, before abandoning. For phantom abandons, ABNTIME includes the total time until the agent releases the call.	C	integer	4 bytes
ACD (index)	The ACD number for which data was collected.	I	smallint	2 bytes
ACDCALLS	The number of split/skill and direct agent ACD calls that were answered by an agent from “queue to main”, “check backup”, “messaging split/skill”, “route to” split/skill or direct agent, and “adjunct routing” to a split/skill or direct agent. ACDCALLS includes BACKUPCALLS .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table

Database Item	Description	Data Type	Column Type	Length
ADJATTEMPTS	The number of adjunct routing attempts for calls in this VECTOR . Available on the ECS and Generic 3 switches with the ASAI feature. ADJATTEMPTS includes ADJROUTED .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
ADJROUTED	The number of adjunct-routing calls that were redirected by an adjunct processor or host computer. Available on the ECS and Generic 3 switches with vectoring and the ASAI feature.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
ANSTIME	The time that split/skill and direct agent ACD calls waited while executing steps in this vector, queuing, and ringing before being answered by an agent. ANSTIME includes RINGTIME .	C	integer	4 bytes
BACKUPCALLS	The number of ACDCALLS that were queued to the answering split/skill using a vector command other than “queue to main.” Calls answered in a main split/skill (MAINCALLS) can then be calculated as ACDCALLS - BACKUPCALLS . However, MAINCALLS does not include direct agent calls. BACKUPCALLS includes “messaging split/skill” calls, “check backup” calls, and calls that route to a split/skill or direct agent, either by the “route to” vector command or by adjunct routing. Calls that are redirected back to the split/skill using the redirection on no answer feature and are subsequently answered are also counted as BACKUPCALLS . NOTE: The Redirect to No Answer to VDN routing feature is available on the <i>DEFINITY</i> ECS. Available on Generic 2 and Generic 3 switches and on the ECS with the vectoring feature.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table

Database Item	Description	Data Type	Column Type	Length
BUSYCALLS	<p>The number of INCALLS that were given a busy signal by the switch.</p> <p>This can occur on all switches when the "busy" vector command is executed.</p> <p>On Generic 3 switches, BUSYCALLS can occur if a call is routed to a split with coverage set to "yes" where there are no agents available, the queue is full (or there is no queue), there is no coverage, and an announcement has played or the trunk is not a Central Office (CO) trunk.</p> <p>Also on Generic 3 switches, BUSYCALLS can occur if a call is routed to a direct agent with coverage set to "yes", the agent is not logged in and there is no coverage path administered and an announcement has played or the trunk is not a CO trunk.</p>	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
BUSYTIME	The time callers waited in queue until hearing a busy tone for all BUSYCALLS .	C	integer	4 bytes
DEFLECTCALLS	<p>The number of calls deflected to the network through Best Service Routing (BSR).</p> <p>Requires the <i>DEFINITY ECS R6</i>.</p>	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table

Database Item	Description	Data Type	Column Type	Length
DISCCALLS	<p>With Generic 2.1 and Generic 3 Version 1 switches, DISCCALLS is the number of INCALLS that were given a forced disconnect announcement by the “disconnect” vector command, listened to the entire announcement, then were disconnected by the switch.</p> <p>With the Generic 2.2 switch, Generic 3 Version 2, and later Generic 3 switches and with the ECS, the number of INCALLS that executed the “disconnect” vector command.</p> <p>With Generic 3 Version 2 and newer switches (and with the ECS), DISCCALLS also includes calls disconnected by the switch when the vector disconnect timer expired or that reached the end of vector processing without being queued.</p> <p>DISCCALLS includes VDISCCALLS.</p>	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
DISCTIME	<p>The time all DISCCALLS spent in this VECTOR. The time until the trunk drops following the forced disconnect command for those calls recorded as DISCCALLS.</p> <p>For Generic 2.1 and Generic 3 Version 1 switches, this is the time until the announcement ends and the caller is disconnected by the switch.</p> <p>For Generic 2.2, Generic 3 Version 2 and later Generic 3 switches, and for the ECS, this is the time until the trunk drops, in the case where the caller hangs up without listening to the entire announcement.</p> <p>For Generic 3 Version 2 and newer switches, and for the ECS, this is the time when the call is disconnected due to the expiration of the vector disconnect timer or the time until the caller is disconnected by the switch.</p>	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
GOTOCALLS	The number of OUTFLOWCALLS that were redirected to another vector by way of a “go to vector” command. Available on Generic 2.2 and Generic 3 switches and on the ECS.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
GOTOTIME	The time all GOTOCALLS spent in this vector before being redirected to another vector. Available on Generic 2.2 and Generic 3 switches and on the ECS.	C	integer	4 bytes
INCALLS	The number of inbound calls that were processed by this vector. INCALLS includes ABNCALLS , RINGCALLS , INFLOWCALLS , and OTHERCALLS . INCALLS = ACDCALLS + ABNCALLS + OTHERCALLS.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
INCOMPLETE	This indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3 and the ECS), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month.	C	smallint	2 bytes
INFLOWCALLS	The number of calls that were redirected to this vector by way of a “go to vector” or a “route to” VDN command, or by redirection on no answer to a VDN.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table

Database Item	Description	Data Type	Column Type	Length
INTERFLOWCALLS	Number of OUTFLOWCALLS that were directed to an off-switch location. INTERFLOWCALLS includes LOOKFLOWCALLS .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
INTIME	The time all DISCCALLS spent in this VECTOR . The time until the trunk drops following the forced disconnect command for those calls recorded as DISCCALLS . For Generic 2.1 and Generic 3 Version 1 switches, this is the time until the announcement ends and the caller is disconnected by the switch. For Generic 2.2, Generic 3 Version 2 and later Generic 3 switches, and for the ECS, this is the time until the trunk drops, in the case where the caller hangs up without listening to the entire announcement. For Generic 3 Version 2 and newer switches, and for the ECS, this is the time when the call is disconnected due to the expiration of the vector disconnect timer or the time until the caller is disconnected by the switch.	C	integer	4 bytes
INTRVL	The number of minutes in the timed period (15, 30, or 60). INTRVL applies to intrahour tables only.	A	smallint; only in hvdn table	2 bytes
LOOKATTEMPTS	The number of look-ahead interflow attempts for calls in this vector. Available on Generic 2.2 and Generic 3 switches and on the ECS with the vectoring and look-ahead interflow features. LOOKATTEMPTS includes LOOKFLOWCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
LOOKFLOWCALLS	Number of INTERFLOWCALLS that were redirected by way of the Lookahead Interflow feature. Available on Generic 2.2 and Generic 3 switches with the Lookahead Interflow feature.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table

Database Item	Description	Data Type	Column Type	Length
NETDISCCALLS	Number of disconnected calls for the reply step in BSR. Requires the <i>DEFINITY ECS R6</i> .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
NETPOLLS	Number of network polls for the consider steps in BSR. Requires the <i>DEFINITY ECS R6</i> .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
OTHERCALLS	The number of INCALLS that were redirected out of the vector, given a busy signal, or were disconnected. OTHERCALLS includes BUSYCALLS , DISCCALLS , and OUTFLOWCALLS . OTHERCALLS = INCALLS - ACDCALLS - ABNCALLS	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
OTHERTIME	The time OTHERCALLS spent in the vector until the disposition was known and the call left the vector. OTHERTIME includes BUSYTIME , DISCTIME , and OUTFLOWTIME .	C	integer	4 bytes
OUTFLOWCALLS	The number of INCALLS that were redirected to another destination by way of a "go to vector" command or by a "route to" or "adjunct routing" command to a destination other than a split/skill or direct agent. (Calls that route to a split/skill or direct agent by way of a "route to," "adjunct routing," or "messaging split/skill" command are still tracked in the vector.) OUTFLOWCALLS includes GOTOCALLS and INTERFLOWCALLS .	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
OUTFLOWTIME	Time all OUTFLOWCALLS spent in the VECTOR before being redirected. OUTFLOWTIME includes GOTOTIME .	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
PHANTOMABNS	The number of split/skill and direct agent ACD calls and calls that routed to an agent or extension with talk time less than the value of the phantom abandoned call timer. Available on Generic 3 and newer switches and the ECS.	C	integer	4 bytes
RINGCALLS	The number of split/skill and direct agent ACD calls that rang at agent positions. RINGCALLS includes ACDCALLS . Available on Generic 2 and Generic 3 switches and the ECS.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvector table
RINGTIME	The time split/skill and direct agent ACD that rang at agent positions. Available on Generic 2 and Generic 3 switches and the ECS.	C	integer	4 bytes
ROW_DATE (index)	The date for which data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
VDISCCALLS	The number of calls forced to disconnect because the vector disconnect timer timed out or because the call reached a vector stop without being queued. "Vector stop" means an explicit "stop" vector command, the end of the vector, or the call executed 1000 vector steps. Available on Generic 3 Version 2 and newer switches and the ECS.	C	integer; smallint in hvector table	4 bytes; 2 bytes in hvdn table
VECTOR (index)	The vector number that this row represents. Available on Generic 2 and Generic 3 switches and on the ECS with vectoring.	I	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support the Vector database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ABNCALLS	X	X	X	X	X	X	X
ABNQUECALLS	X	X	X	X	X	X	X
ABNRINGCALLS	X	X	X	X	X	X	X
ABNTIME	X	X	X	X	X	X	X
ACD	X	X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
ADJATTEMPTS			X	X	X	X	X
ADJROUTED			X	X	X	X	X
ANSTIME	X	X	X	X	X	X	X
BACKUPCALLS	X	X	X	X	X	X	X
BUSYCALLS	X	X	X	X	X	X	X
BUSYTIME	X	X	X	X	X	X	X
DEFLECTCALLS							X
DISCCALLS	X	X	X	X	X	X	X
DISCTIME	X	X	X	X	X	X	X
GOTOCALLS		X	X	X	X	X	X
GOTOTIME		X	X	X	X	X	X
INCALLS	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INFLOWCALLS	X	X	X	X	X	X	X
INTERFLOWCALLS	X	X	X	X	X	X	X
INTIME	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
LOOKATTEMPTS		X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
LOOKFLOWCALLS		X	X	X	X	X	X
NETDISCCALLS							X
NETPOLLS							X
OTHERCALLS	X	X	X	X	X	X	X
OTHERTIME	X	X	X	X	X	X	X
OUTFLOWCALLS	X	X	X	X	X	X	X
OUTFLOWTIME	X	X	X	X	X	X	X
PHANTOMABNS			X	X	X	X	X
RINGCALLS	X	X	X	X	X	X	X
RINGTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
VDISCCALLS				X	X	X	X
VECTOR	X	X	X	X	X	X	X

VDN Database Items

Overview

⇒ NOTE:

VDN database items are available only if the Vectoring feature has been purchased and authorized for you to use.

The VDN Database Item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical VDN database items apply to the Intrahour VDN (*hvdn*), Daily VDN (*dvdn*), Weekly VDN (*wvdn*), and Monthly VDN (*mvdn*) tables. All items listed in the following VDN Database Items table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four VDN tables are also noted in the Column Type column. The historical indexes are **ROW_DATE** and **VDN**.

VDN Database Items

The following table describes the data items in the CMS VDN database tables.

Database Item	Description	Data Type	Column Type	Length
ABNCALLS	<p>The number of INCALLS that were abandoned while INPROGRESS for this VDN. This includes split/skill and direct agent ACD calls that abandon from queue or from ringing, calls that abandon from vector processing, calls that abandon after being routed to an extension via the "route to" vector command, and for the Generic 2.1 and Generic 3 (prior to Generic 3 Version 2 load 100) switches, calls that abandoned while listening to a forced disconnect announcement.</p> <p>ABNCALLS includes ACD calls and calls routed to an agent or extension with talk times less than the value of the phantom abandoned call timer.</p> <p>ABNCALLS includes ABNCALLS1 through ABNCALLS10, ABNQUECALLS, ABNRINGCALLS, PHANTOMABNS, and SLVLABNS are pegged as ABNCALLS.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ABNCALLS1-10	<p>The number of INCALLS that abandoned in each of the service level increments PERIOD1 through PERIOD9 (as defined IN the ACD Administration: VDN Call Profile Setup window). ABNCALLS10 counts calls that abandoned after PERIOD9.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ABNQUECALLS	<p>The number of ABNCALLS that were abandoned while in a split/skill or direct agent ACD queue.</p> <p>Available on Generic 2 and Generic 3 switches and the ECS.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ABNRINGCALLS	<p>The number of split/skill and direct agent ABNCALLS that were abandoned by the caller while ringing at an agent position.</p> <p>Available on Generic 2 and Generic 3 switches and on the ECS.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
ABNTIME	The time caller waited while vector steps were executed, the call was queued, and ringing before abandoning. For phantom-abandon calls, ABNTIME is the total time from entering the VDN until the agent released the call.	C	integer	4 bytes
ACCEPTABLE	The number of ACDCALLS and CONNECTCALLS that were answered within the acceptable service level (SERVICELEVEL) as defined on the ACD Administration: VDN Call Profile Setup window.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ACD (index)	The ACD number for which data was collected.	I	smallint	2 bytes
ACDCALLS	The number of split/skill and direct agent ACD calls that were answered by an agent from “queue to main,” “check backup,” “messaging split/skill,” “route to” split/skill or direct agent, and “adjunct routing” to a split/skill or direct agent. ACDCALLS includes ACDCALLS1-10 , ACCEPTABLE , ANSCONNCALLS1-10 , BACKUPCALLS , and TRANSFERRED .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ACDTIME	The talk time of all ACDCALLS , not including HOLDTIME . ACDTIME includes SKILLTIME1 , SKILLTIME2 , and SKILLTIME3 .	C	integer	4 bytes
ACWTIME	The time that agents spent in ACW associated with ACDCALLS . ACWTIME includes SKILLACWTIME1-3 .	C	integer	4 bytes
ADJATTEMPTS	The number of adjunct-routing attempts for calls in this VDN. ADJATTEMPTS includes ADJROUTED . Available on Generic 3 switches and on the ECS with the ASAI gateway.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
ADJROUTED	The number of adjunct routing calls that were redirected by an adjunct processor or host computer. Available on Generic 3 switches and on the ECS with vectoring and the ASAI feature.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ANSCONNCALLS 1-10	The number of times that callers were answered (ACDCALLS) and connected (CONNECTCALLS) during each of the service level increments PERIOD1 through PERIOD9 as defined in the ACD Administration: VDN Call Profile Setup window. ANSCONNCALLS10 counts calls answered or connected after PERIOD9 . Answered/connected calls include split/skill and direct agent ACD calls and extension calls by a "route to" or "adjunct routing" vector command.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
ANSTIME	The time split/skill and direct agent ACD calls spent waiting to be answered in vector processing, in queue, and while ringing. For extension calls on Generic 2.1 switches, ANSTIME is the time until ringing starts. ANSTIME includes RINGTIME .	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
BACKUPCALLS	<p>The number of ACDCALLS that were queued to the answering split/skill by a vector command other than “queue to main”. Calls answered in the main split/skill can then be calculated as ACDCALLS - BACKUPCALLS. However, this calculation does not include direct agent calls. BACKUPCALLS includes “messaging split/skill” calls, “check backup” calls, and calls that route to a split/skill or direct agent, either by the “route to” vector command or by adjunct routing. Calls that are redirected back to the split/skill using the Redirection on No Answer feature and then answered are also counted as BACKUPCALLS.</p> <p>NOTE: The Redirect on No Answer to VDN routing feature is available on the <i>DEFINITY</i> ECS.</p> <p>Available on Generic 2 and Generic 3 switches and the ECS with the vectoring feature.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_ABNCALLS	The number of INCALLS that were abandoned by callers during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_ACDCALLS	The number of ACDCALLS that were completed during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_ACDTIME	The talk time of ACDCALLS that were completed during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_BUSYCALLS	The number of INCALLS that were given a busy signal by the switch during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
BH_DISCCALLS	The number of INCALLS that were disconnected by the switch during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_INTERVAL	The defined interval length, with the start time, of the busy hour.	B	integer; only in dvdn table	4 bytes
BH_OTHERCALLS	The number of OTHERCALLS that completed during the busy hour. BH_OTHERCALLS includes extension-in calls, calls forced busy or disconnected, calls that outflowed off the switch, short inbound calls, and inbound calls of unknown disposition.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_STARTTIME	The starting time of the hour for which busy hour data was collected. The busy hour is that set of contiguous intervals comprising a total of one hour in which the number of INCALLS to the VDN was a maximum.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BH_VDNCALLS	The number of INCALLS to the VDN that completed during the busy hour. BH_VDNCALLS includes answered calls that completed during the busy hour, calls that abandoned, were forced busy, forced disconnected or outflowed from the VDN during the busy hour.	B	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BSRPLAN	Information for the specified Best Service Routing (BSR) plan. Available on the R6 ECS.	A	smallint	2 bytes

Database Item	Description	Data Type	Column Type	Length
BUSYCALLS	<p>The number of INCALLS that were given a busy signal by the ECS.</p> <p>This can occur on all ECS releases through the “busy” vector command.</p> <p>On the Generic 3 ECS, BUSYCALLS can occur if a call is routed to a split/skill with coverage set to “yes” where there are no agents available, the queue is full (or there is no queue), there is no coverage, and an announcement has played or the trunk is not a CO trunk.</p> <p>Also on the Generic 3 ECS, BUSYCALLS can occur if a call is routed to a direct agent with coverage set to “yes,” the agent is not logged in and there is no coverage path administered, an announcement has played, or the trunk is not a CO trunk.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
BUSYTIME	The duration of all BUSYCALLS (until the trunk goes idle). The format is the number of seconds that have accumulated.	C	integer	4 bytes
CONNECTCALLS	<p>The number of non-ACD INCALLS that were delivered to a station extension (other than a VDN or direct agent login ID) by a “route to” or “adjunct routing” vector command and were not abandoned by the callers.</p> <p>For Generic 2.1 ECS, non-ACD abandons are not tracked, so all calls that route to a station extension (other than a VDN) are included in CONNECTCALLS for that ECS release.</p> <p>CONNECTCALLS includes ANSCONNCALLS1-10.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
CONNECTTIME	The time CONNECTCALLS waited before being answered (for the Generic 3 ECS). For Generic 2 ECS, CONNECTTIME is the time before ringing starts. The format is the number of seconds that have accumulated.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
CONNTALKTIME	The talk time for all CONNECTCALLS , not including HOLDTIME (except on Generic 2.1). The format is the number of seconds that have accumulated.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
DEFLECTCALLS	Number of calls that were deflected to the network by BSR. Requires the R6 ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
DISCCALLS	With Generic 2.1 and Generic 3 Version 1 switches, DISCCALLS is the number of INCALLS that were given a forced disconnect announcement by the "disconnect" vector command, listened to the entire announcement, then were disconnected by the switch. With Generic 2.2, Generic 3 Version 2 and later Generic 3 switches, and with the ECS, this is the number of INCALLS that were disconnected by the switch by the "disconnect" vector command. With Generic 3 Version 2 and newer Generic 3 switches and with the ECS, DISCCALLS also includes calls disconnected by the switch when the vector disconnect timer expired or that reached the end of vector processing without being queued. DISCCALLS includes VDISCCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
DISCTIME	The time all DISCCALLS spent in this VDN. The time until the trunk drops following the forced disconnect command for those calls recorded as DISCCALLS . For Generic 2.2, Generic 3 Version 2, and newer Generic 3 switches, and for the ECS, if the caller hangs up during the forced disconnect announcement, this is the time until the caller hangs up. For Generic 2.1 and Generic 3 switches, this is the time until the announcement ends and the caller is disconnected by the switch.	C	integer	4 bytes

Database Item	Description	Data Type	Column Type	Length
HOLDABNCALLS	The number of times that callers abandoned from on hold. For Generic 2.1 switches, HOLDABNCALLS applies to split ACD calls held. For Generic 2.2 and Generic 3 switches, and for the ECS, HOLDABNCALLS applies to all calls the agent put on hold.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
HOLDACDCALLS	The number of split/skill or direct agent ACD calls placed on hold at least one time.	C	integer	4 bytes
HOLDACDTIME	The time spent by split/skill or direct agent ACD callers spent on hold.	C	integer	4 bytes
HOLDCALLS	The number of calls that were placed on hold at least once. HOLDCALLS also includes HOLDABNCALLS and HOLDACDCALLS . For Generic 2.1 switches, HOLDCALLS applies to split ACD calls held. For Generic 2.2 and Generic 3 switches, and for the ECS, HOLDCALLS applies to all calls the agent put on hold.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
HOLDTIME	Time spent by callers on hold. HOLDTIME includes HOLDACDTIME . For Generic 2.1 switches, HOLDTIME applies to split ACD calls. For Generic 2.2 and Generic 3 switches and for the ECS, HOLDTIME applies to all calls the agent put on hold.	C	integer	4 bytes
I_ARRIVED	Number of calls that reached this VDN during this interval.	C	integer	4 bytes
INCALLS	Number of inbound calls that were directed to this VDN. INCALLS includes ABNCALLS , INFLOWCALLS , OTHERCALLS , RETURNCALLS , and RINGCALLS (which includes ACDCALLS). INCALLS = ABNCALLS + ACDCALLS + OTHERCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
INCOMPLETE	<p>Indicates whether or not data is complete for this collection interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3 and the ECS), protocol failures with data collection active, or when the split/skill or VDN call profile is changed with data collection active.</p> <p>The value for interval tables indicates whether data is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month.</p>	C	smallint	2 bytes
INFLOWCALLS	<p>Number of calls that were redirected to the split's/skill's queue from another queue. When a call leaves the VDN (for example, by routing to another VDN) or leaves vector processing, (for example, by routing to a split/skill) the next split/skill to which a call queues will not be credited with an inflow. Calls that ring at an agent and are then requeued to the same split/skill by the Redirect on No answer feature are counted as inflows to that split/skill.</p> <p>On Generic 2 and Generic 3 switches with vectoring, an inflow is counted for calls that intraflow from one split's queue to another (that is, call that queue to a split after having previously been queued to another split).</p>	C	integer; smallint in hsplit table	4 bytes; 2 bytes in hsplit table
INTERFLOWCALLS	<p>The number of OUTFLOWCALLS that were redirected to a destination outside the switch. INTERFLOWCALLS includes LOOKFLOWCALLS.</p>	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
INTIME	The time spent by INCALLS in this VDN. INTIME = ACDTIME + ABNTIME + ANSTIME + HOLDTIME + OTHERTIME However, on Generic 2.1 switches, there are multiple call handling scenarios in which call-based ACDTIME is stopped before the call ends. In these scenarios, INTIME does not add up to ACDTIME + ABNTIME + ANSTIME + OTHERTIME + HOLDTIME . (The scenarios occur when an agent puts an ACD call on hold using the HOLD key when another ACD call is already on hold, or when a call is dropped while an ACD call is on hold.)	C	integer	4 bytes
INTRVL	The number of minutes in the timed period (15, 30, or 60). INTRVL applies to intrahour tables only.	A	smallint; only in hvdn table	2 bytes
LOOKATTEMPTS	The number of look-ahead interflow attempts for calls in this VDN. Available on Generic 2.2 and Generic 3 switches and on the ECS with the vectoring and look-ahead interflow features. LOOKATTEMPTS includes LOOKFLOWCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
LOOKFLOWCALLS	Number of INTERFLOWCALLS that were redirected by way of the Lookahead Interflow feature. Available on Generic 2.2 and Generic 3 switches with the Lookahead Interflow feature.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
MAXOCWTIME	The maximum time that a call, recorded during the collection interval, waited in the VDN before being answered (ACD calls) or connected (non-ACD calls), abandoning, being redirected, receiving a busy signal or being disconnected. This applies only to the first disposition of the call.	M	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
MAXWAITING	Maximum number of calls simultaneously in progress in the VDN during the collection interval.	M	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
NETDISCCALLS	Number of calls that disconnected from the BSR reply step. Requires the R6 ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
NETINCALLS	Calls that interflowed in from the network in BSR. Requires the R6 ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
NETINTIME	Time, in seconds, that the call was in a VDN somewhere else in the network. Requires the R6 ECS.	C	integer	4 bytes
NETPOLLS	Number of network polls for BSR consider steps. Requires the R6 ECS.	C	integer	4 bytes
NOANSREDIR	Number of split/skill and direct agent ACD calls that rang at agent stations and then were automatically redirected by the Redirection on No Answer feature because they were not answered. Available on Generic 3 Version 2 and newer switches and on the ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
NUMTGS	Number of trunk groups assigned to this VDN.	A	integer	4 bytes
OTHERCALLS	Number of calls that were given a forced busy, forced disconnect, or outflowed from the switch, and non-ACD calls that were answered (CONNECTCALLS). OTHERCALLS includes BUSYCALLS , CONNECTCALLS , DISCCALLS , and OUTFLOWCALLS . OTHERCALLS = INCALLS - ACDCALLS - ABNCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
OTHERTIME	Duration of all OTHERCALLS until the calls leave the VDN (the calls drop, are sent to another VDN, are transferred, or are sent outside the switch). OTHERTIME includes BUSYTIME , CONNECTTIME , CONNTALKTIME , DISCTIME , and OUTFLOWTIME .	C	integer	4 bytes
OUTFLOWCALLS	Number of INCALLS that were redirected to another VDN or to a destination outside the switch by way of a “route to” or “adjunct routing” command, or were redirected to another VDN by the Redirect on No Answer feature. Note that calls are only counted as outflows from the VDN when they are redirected to another VDN or to an off-switch destination. Calls in the VDN that route to other destinations, such as split/skills or extensions, are not counted as outflows from the VDN. OUTFLOWCALLS includes INTERFLOWCALLS and SLVLOUTFLOWS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
OUTFLOWTIME	Time all OUTFLOWCALLS spent in the VDN before being redirected.	C	integer	4 bytes
PERIOD1-9	Length, in seconds, of each service level increment as defined in the ACD administration. Each increment represents a progressively longer wait time. <i>CentreVu</i> CMS counts answered or abandoned calls that wait beyond the last increment (PERIOD9) in ANSCONNCALLS10 .	A	smallint	2 bytes
PERIODCHG	This indicates whether or not service level increments PERIOD1 through PERIOD9 (as defined on the ACD Administration: VDN Call Profile window) changed during the data collection interval. Valid values for PERIODCHG are 1 = YES and 0 = NO.	A	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
PHANTOMABNS	Number of split/skill and direct agent ACD calls and calls that routed to an agent or extension with talk time of less than the value set for the phantom abandoned call timer. Available on Generic 3 and newer switches and on the ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
RETURNCALLS	Number of calls that reached this VDN via the VDN return destination feature. Available on Generic 3 Version 3 and newer switches and on the ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
RINGCALLS	Number of split/skill and direct agent ACD calls that rang at agent positions. Available on Generic 2 and Generic 3 switches and on the ECS. RINGCALLS includes ACDCALLS .	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
RINGTIME	Time split/skill and direct agent ACD calls spent ringing at agent positions, independent of final disposition. Available on Generic 2 and Generic 3 switches and on the ECS.	C	integer	4 bytes
ROW_DATE (index)	Date for which the data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
SERVICELLEVEL	Number of seconds within which calls must be answered/connected to be considered acceptable (as defined on the ACD Administration: VDN Call Profile Setup window).	A	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
SKILLACWTIME1-3	ACW time spent by agents for calls answered in each VDN skill preference. Available on Generic 2.2 and Generic 3 switches with EAS.	C	integer	4 bytes
SKILLCALLS1-3	Number of calls answered by agents in each VDN skill preference. Available on Generic 2.2 and Generic 3 Version 2 switches with EAS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
SKILLTIME1-3	Time agents spent talking on calls they answered in each VDN skill preference. Available on Generic 2.2 and Generic 3 switches with EAS.	C	integer	4 bytes
SKILL1-3	First, second, and third VDN skill assigned to this VDN. Available on Generic 2.2 and Generic 3 and newer switches and on the ECS with EAS.	A	smallint	2 bytes
SLVLABNS	Number of ABNCALLS whose time to abandon was less than or equal to this VDN's SERVICELEVEL .	C	integer	4 bytes
SLVLOUTFLOWS	Number of OUTFLOWCALLS whose time to outflow was less than or equal to this VDN's SERVICELEVEL .	C	integer	4 bytes
SVCLEVELCHG	Indicates whether the service level was changed during the data collection interval. Valid values for SVCLEVELCHG are 1 = YES and 0 = NO.	A	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
TRANSFERRED	Number of calls that were transferred to another destination. For Generic 2.1 switches, TRANSFERRED includes all VDN calls blind transferred to a measured VDN or split. For Generic 2.2 and Generic 3 switches and for the ECS, TRANSFERRED includes all VDN calls transferred.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table
VDISCCALLS	Number of calls forced to disconnect because the vector disconnect timer timed out, or because the call reached a vector stop without being queued. ("Vector stop" means a "stop" vector command, the end of the vector, or the call executed 1000 vector steps.) Available on Generic 3 Version 2 and newer switches and on the ECS.	C	integer; smallint in hvdn table	4 bytes; 2 bytes in hvdn table

Database Item	Description	Data Type	Column Type	Length
VDN (index)	Vector directory number associated with this VDN.	I	char(6)	6 byte ASCII text string
VECTOR (index)	Vector number associated with this VDN.	A	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each VDN database item.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ABNCALLS	X	X	X	X	X	X	X
ABNCALLS1-10	X	X	X	X	X	X	X
ABNQUECALLS	X	X	X	X	X	X	X
ABNRINGCALLS	X	X	X	X	X	X	X
ABNTIME	X	X	X	X	X	X	X
ACCEPTABLE	X	X	X	X	X	X	X
ACD	X	X	X	X	X	X	X
ACDCALLS	X	X	X	X	X	X	X
ACDTIME	X	X	X	X	X	X	X
ACWTIME	X	X	X	X	X	X	X
ADJATTEMPTS			X	X	X	X	X
ADJROUTED			X	X	X	X	X
ANSCONNCALLS1-10	X	X	X	X	X	X	X
ANSTIME	X	X	X	X	X	X	X
BH_ABNCALLS	X	X	X	X	X	X	X
BH_ACDCALLS	X	X	X	X	X	X	X
BH_ACDTIME	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
BH_BUSYCALLS	X	X	X	X	X	X	X
BH_DISCCALLS	X	X	X	X	X	X	X
BH_INTERVAL	X	X	X	X	X	X	X
BH_OTHERCALLS	X	X	X	X	X	X	X
BH_STARTTIME	X	X	X	X	X	X	X
BH_VDNCALLS	X	X	X	X	X	X	X
BACKUPCALLS	X	X	X	X	X	X	X
BUSYCALLS	X	X	X	X	X	X	X
BUSYTIME	X	X	X	X	X	X	X
CONNECTCALLS	X	X	X	X	X	X	X
CONNECTTIME	X	X	X	X	X	X	X
CONNTALKTIME	X	X	X	X	X	X	X
DEFLECTCALLS							X
DISCCALLS	X	X	X	X	X	X	X
DISCTIME	X	X	X	X	X	X	X
HOLDABNCALLS	X	X	X	X	X	X	X
HOLDACDCALLS	X	X	X	X	X	X	X
HOLDACDTIME	X	X	X	X	X	X	X
HOLDCALLS	X	X	X	X	X	X	X
HOLDTIME	X	X	X	X	X	X	X
I_ARRIVED	X	X	X	X	X	X	X
INCALLS	X	X	X	X	X	X	X
INCOMPLETE	X	X	X	X	X	X	X
INFLOWCALLS	X	X	X	X	X	X	X
INTERFLOWCALLS	X	X	X	X	X	X	X
INTIME	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
LOOKATTEMPTS		X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
LOOKFLOWCALLS		X	X	X	X	X	X
MAXOCWTIME	X	X	X	X	X	X	X
MAXWAITING	X	X	X	X	X	X	X
NETDISCCALLS							X
NETINCALLS							X
NETINTIME							X
NETPOLLS							X
NOANSREDIR				X	X	X	X
NUMTGS							X
OTHERCALLS	X	X	X	X	X	X	X
OTHERTIME	X	X	X	X	X	X	X
OUTFLOWCALLS	X	X	X	X	X	X	X
OUTFLOWTIME	X	X	X	X	X	X	X
PERIOD1-9	X	X	X	X	X	X	X
PERIODCHG	X	X	X	X	X	X	X
PHANTOMABNS			X	X	X	X	X
RETURNCALLS				X (V3)	X	X	X
RINGCALLS	X	X	X	X	X	X	X
RINGTIME	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SERVICELEVEL	X	X	X	X	X	X	X
SKILL1-3		(EAS)		(EAS)	(EAS)	(EAS)	X
SKILLACWTIME1-3		(EAS)		(EAS)	(EAS)	(EAS)	X
SKILLCALLS1-3		(EAS)		(EAS)	(EAS)	(EAS)	X
SKILLTIME1-3		(EAS)		(EAS)	(EAS)	(EAS)	X
SLVLABNS	X	X	X	X	X	X	X
SLVLOUTFLOWS	X	X	X	X	X	X	X
STARTTIME	X	X	X	X	X	X	X

VDN Database Items

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
SVCLEVELCHG	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X
VDISCCALLS				X	X	X	X
VDN	X	X	X	X	X	X	X
VECTOR	X	X	X	X	X	X	X

Call Work Codes Database Items

Overview

⇒ NOTE:

Call work codes are only available with Generic 3 and Generic 2.2 ECS. The Call Work Codes database item descriptions apply to historical items.

The **Data Type** column refers to **Cumulative (C)**, **Administrative (A)**, **Status (S)**, **Row Identifier (I)**, **Busy Hour (B)**, **Special Table (N)**, or **Maximum Value (M)** data. Cumulative, Administrative, Row Identifier, Busy Hour, and Maximum Value items apply to both the current and previous interval real-time tables. Special Table items are historical, and apply only to the table in which they are stored.

Historical call work codes database items apply to the Intrahour Call Work Codes (*hcwc*), Daily Call Work Codes (*dcwc*), Weekly Call Work Codes (*wcwc*), and Monthly Call Work Codes (*mcwc*) tables. All items listed in the following Call Work Codes Database Items table are included in all four tables, unless otherwise noted in the Column Type column. Any differences in the data format between the four call work codes tables are also noted in the Column Type column. The indexes are **ACD**, **ROW_DATE** and **CWC**.

Call Work Codes Database Items

The following table describes the data items in the CMS Call Work Codes database table.

Database Item	Description	Data Type	Column Type	Length
ACD (index)	The ACD number for which data was collected.	I	smallint	2 bytes
ACDCALLS	Number of times this call work code was entered while an agent was on a split/skill or direct agent ACD call or in call-related ACW.	C	integer; smallint in hcwc table	4 bytes; 2 bytes in hcwc table

Database Item	Description	Data Type	Column Type	Length
ACDTIME	Talk time of all ACDCALLS (not including HOLDTIME) associated with this call work code.	C	integer	4 bytes
ACWTIME	Time that the agent spent in ACW for ACDCALLS that were associated with this call work code.	C	integer	4 bytes
CWC (index)	Call work code for which data was collected.	I	char(16)	16 byte ASCII text string
INCOMPLETE	This indicates whether or not data is complete for this interval. Data is incomplete whenever the link goes down and whenever tracking is aborted for calls, due to trunk failures (Generic 2), the trunk going maintenance busy with a call active (Generic 3), protocol failures with data collection active, or when split/skill or VDN call profile is changed with data collection active. The value for interval tables indicates whether data collection is incomplete for the interval (0 = NO, 1 = YES). The value in the daily, weekly, and monthly tables indicates the number of incomplete intervals in the day, week, or month. Changing split/skill or VDN call profile data while data collection is active only affects the respective split/skill or VDN data.	C	smallint	2 bytes
INTRVL	The number of minutes in the intrahour interval (15, 30, or 60). INTRVL applies to intrahour tables only.	A	smallint; in hcwc table only	2 bytes
ROW_DATE (index)	The day for which data was collected or the exception occurred.	I	date	4 byte <i>Informix</i> date
STARTTIME	Start time of the interval for which data was collected. STARTTIME applies to the only interval table.	I	smallint; in hcwc table only	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each Call Work Code database item.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	<i>DEFINITY</i> ECS R5	<i>DEFINITY</i> ECS R6
ACD		X	X	X	X	X	X
ACDCALLS		X	X	X	X	X	X
ACDTIME		X	X	X	X	X	X
ACWTIME		X	X	X	X	X	X
CWC		X	X	X	X	X	X
INCOMPLETE		X	X	X	X	X	X
INTRVL		X	X	X	X	X	X
ROW_DATE		X	X	X	X	X	X
STARTTIME		X	X	X	X	X	X

Agent Login/Logout Database Items

Overview

The Agent Login/Logout database item descriptions are **historical** items that apply specifically to the Agent Login/Logout (`haglog`) table. The indexes are **ACD**, **SPLIT** and **ROW_DATE**.

Agent Login/Logout Database Items

The following table describes the data items in the CMS Agent Login/Logout database table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
EXTN	Extension number of the station that the agent staffed.	char(6)	6 byte ASCII text string
INFLAG	If not null, indicates that agent was already logged in when the link came up. Values are NULL and "<."	char(1)	1 byte ASCII text string
LOGID	Login ID that was used to staff the EXTENSION . Agents in multiple splits/skills have one LOGID .	char(10)	10 byte ASCII text string
LOGIN	Time at which the agent logged into this extension and split/skill with the given login ID. This field is a standard UNIX time field; that is, the time is stored as the number of seconds since January 1, 1970.	integer	4 bytes

Database Item	Description	Column Type	Length
LOGONSKILL2-20	Second through twentieth skills the agent logged in with. NOTE: The number of skills per agent depends on the type of switch. For example, prior to the ECS, LOGONSKILL5 is only available on Generic 2.2 switches. Available on the ECS with the EAS feature and the Generic 2.2 and later Generic 3 switches with the EAS feature.	smallint	2 bytes
LOGOUT	Time at which the agent logged out. This field is a standard UNIX time field; that is, the time is stored as the number of seconds since January 1, 1970.	integer	4 bytes
LOGOUT_DATE	Date on which the agent logged out.	date	4 byte <i>Informix</i> date
LOGOUT_REASON	Reason code (0 through 9) associated with the agent's logout. For switch releases earlier than the ECS or switch releases that do not have the EAS feature and reason codes active, this field will always contain a 0 when the agent has logged out.	smallint	2 bytes
OUTFLAG	If not null, indicates that the agent logged out while the link was down. Values are NULL and ">."	char(1)	1 byte ASCII text string
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
SKILLTYPE	Type ("p" for primary or "s" for secondary) of the second, third, and fourth skills the agent logged into. NOTE: For the ECS and later switch releases with the EAS feature, skill level 1 will be represented by "p," skill level 2 will be represented by "s," and skill levels 3 through 16 by blank. Users of more than 2 skill levels should use SKLEVEL instead of SKILLTYPE items. Available for Generic 3 switches with EAS and requires Generic 2.2 or Generic 3 Version 2 switches with EAS.	char(1)	1 byte ASCII text string

Database Item	Description	Column Type	Length
SKILLTYPE2-4	The type (“p” for primary or “s” for secondary) of the second, third, and fourth skills the agent logged into. Requires Generic 2.2 or Generic 3 Version 2 switches with EAS. NOTE: For the ECS and newer EAS releases, skill level 1 will be represented by “p”, skill level 2 by “s” and skill levels 3-16 by blank. Users of more than 2 skill levels should use SKLEVEL items instead of SKILLTYPE items.	char(1)	1 byte ASCII text string
SKLEVEL	Skill level (from 1 through 16) associated with the first skill the agent logged into.	smallint	2 bytes
SKLEVEL2-20	Skill levels (from 1 through 20) associated with the second through twentieth skills (LOGONSKILL2 through LOGONSKILL20) the agent logged into.	smallint	2 bytes
SPLIT (index)	Split number to which the extension is assigned or skill number the agent logged into.	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each Agent Login/Logout database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ACD	X	X	X	X	X	X	X
EXTN	X	X	X	X	X	X	X
INFLAG	X	X	X	X	X	X	X
LOGID	X	X	X	X	X	X	X
LOGIN	X	X	X	X	X	X	X
LOGONSKILL2-4		(EAS)		(EAS)	(EAS)	(EAS)	X
LOGONSKILL5		(EAS)				(EAS)	X
LOGONSKILL6-20						(EAS)	X
LOGOUT	X	X	X	X	X	X	X

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
LOGOUT_DATE	X	X	X	X	X	X	X
LOGOUTREASON						(EAS)	X
OUTFLAG	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SKILLTYPE				X (EAS)	X (EAS)	*	X
SKILLTYPE2-4				X (EAS)	X (EAS)	*	X
SKLEVEL				X (EAS)	X (EAS)	X (EAS)	X
SKLEVEL2-4				X (EAS)	X (EAS)	X (EAS)	X
SKLEVEL5-20						X (EAS)	X
SPLIT	X	X	X	X	X	X	X

* Not recommended.

Agent Trace Database Items

Overview

The Agent Trace database item descriptions are **historical** items that apply specifically to the Agent Trace (`ag_actv`) table. The indexes are **ACD**, **LOGID**, and **ROW_DATE**.

Agent Trace Database Items

The following table describes the data items in the CMS Agent Trace database table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
AGT_RELEASED	Agent released or dropped the split/skill or direct agent ACD call. This is always true for ACD calls the agent transferred or conferenced. Available on Generic 3 and newer switches.	char(1)	1 byte ASCII text string
ASSIST_ACTV	Agent requested supervisor assistance (pressed the ASSIST button).	char(1)	1 byte ASCII text string
AUXREASON	Reason code associated with the agent's state. This is blank if the agent is not in the AUX state. For agents in AUX on switch releases that are earlier than the ECS or that do not have EAS and reason codes active, this will be 0 (zero).	smallint	2 bytes
CALLER_HOLD	Agent put the current call on hold. For Generic 2.1 switches, CALLER_HOLD applies to split ACD calls. For Generic 2.2 and Generic 3 switches, CALLER_HOLD applies to all calls the agent put on hold.	char(1)	1 byte ASCII text string
CALLING_II	Information Indicator (II) digits associated with the call. These digits supply information about the originator location; for example, pay phone, hospital, or prison. Available on the ECS and newer switches.	char(2)	2 byte ASCII text string

Database Item	Description	Column Type	Length
CALLING_PTY	Calling party identification, which is the Automatic Number Identification (ANI)/Station Identification (SID) (for Generic 2.2 and Generic 3 Version 4 and later switches with ISDN ANI delivery), extension or trunk equipment location identifying the originator of the call. The field is blank if the trunk is not measured or, for internal calls, if the originating extension is not measured.	char(12)	12 byte ASCII text string
CONFERENCE	Agent activated a conference. Available on Generic 3 and Generic 2.2 switches.	char(1)	1 byte ASCII text string
DIGITS_DIALED	Digits the agent dialed to originate a call. Trunk access codes, feature access codes, account and authorization codes are not included. Available on Generic 2.2 and Generic 3 switches.	char(16)	16 byte ASCII text string
DIRECTION	Direction of the call the agent is currently handling for any split/skill. Valid values are IN, OUT, or as defined in Dictionary. If the agent is not on a call, the value is blank (NULL).	smallint	2 bytes
DURATION	Duration of current WORKMODE and DIRECTION for this split (for example, length of time in current AGSTATE for this split).	integer	4 bytes
EVENT_TIME	Time of day (hour, minute, and second) the WORKMODE or DIRECTION changed.	integer	4 bytes
EXT_CALL_ORIG	Agent originated an external (off-switch) call. Available on Generic 2.2 and Generic 3 switches.	char(1)	1 byte ASCII text string
KEYBD_DIALED	Call was keyboard dialed. Available on Generic 2.2 switches with the ASAI Gateway Interface feature and Generic 3 switches with the ASAI feature.	char(1)	1 byte ASCII text string
LOGID (index)	Login ID that was used to staff the EXTENSION . Agents in multiple splits/skills have one LOGID .	char(10)	10 byte ASCII text string

Database Item	Description	Column Type	Length
LOGOUTREASON	Reason code (0 through 9) associated with the agent's logout. For switch releases earlier than the ECS or switch releases that do not have the EAS feature and reason codes active, this field will always contain 0 when the agent has logged out.	smallint	2 bytes
MCT	The agent activated a malicious call trace. Available on Generic 2 and Generic 3 (except for Generic 3i Version 1) switches.	char(1)	1 byte ASCII text string
RECONNECT	This event represents the agent reconnecting to the call after putting it on hold. Available on Generic 2 and Generic 3 (except for Generic 3 Version 1) switches.	char(1)	1 byte ASCII text string
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
SPLIT	The split number to which the EXTENSION is assigned or the skill number the agent logged into.	smallint	2 bytes
STARTTIME	The time of day (hour and minute) for which the agent trace is being ordered. This is the time of day you enter to request the report.	smallint	2 bytes
TRANSFERRED	Whether or not an answering agent initiated a transfer for this call. Valid values are YES and NO. For Generic 2.1 switches, TRANSFERRED includes transfers to measured VDNs or splits/skills. For Generic 3 and Generic 2.2 switches, TRANSFERRED includes all calls that are transferred.	char(1)	1 byte ASCII text string
UCID	The UCID is the Universal Call Identifier—a unique number assigned to this call segment within the customer network. Requires the <i>DEFINITY</i> ECS R6.	char(20)	20 byte ASCII text string
WMODE_SEQ	Sequence number for events that occur in the same second.	smallint	2 bytes

Database Item	Description	Column Type	Length
WORKCODE	Call work code the agent entered for the call. Available on Generic 3 and Generic 2.2 switches with call work codes.	char(16)	16 byte ASCII text string
WORKMODE	Work mode in which the agent was working during the trace. Agent work modes include: AVAIL, ACD, ACW, AUX, DACD, DACW, RING, UNKNOWN, OTHER, and UNSTAFF. If the agent has not been logged in during the collection interval, the value is blank.	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each Agent Trace database item.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ACD	X	X	X	X	X	X	X
AGT_RELEASED			X	X	X	X	X
ASSIST_ACTV	X	X	X	X	X	X	X
AUXREASON	X	X	X	X	X	X	X
CALLER_HOLD	X	X	X	X	X	X	X
CALLING_II						X	X
CALLING_PTY	X	X	X	X	X	X	X
CONFERENCE		X	X	X	X	X	X
DIGITS_DIALED		X	X	X	X	X	X
DIRECTION	X	X	X	X	X	X	X
DURATION	X	X	X	X	X	X	X
EVENT_TIME	X	X	X	X	X	X	X
EXT_CALL_ORIG		X	X	X	X	X	X
KEYBD_DIALED		X	X	X	X	X	X

Agent Trace Database Items

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	<i>DEFINITY</i> ECS R5	<i>DEFINITY</i> ECS R6
LOGID (index)	X	X	X	X	X	X	X
LOGOUTREASON	X	X	X	X	X	X	X
MCT	X	X	X	X	X	X	X
RECONNECT	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X
STARTTIME	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X
UCID							X
WMODE_SEQ	X	X	X	X	X	X	X
WORKCODE		X	X	X	X	X	X
WORKMODE	X	X	X	X	X	X	X

Current Day Configuration Forecast Database Items

Overview

The Current Day Configuration database item descriptions are **historical** items used specifically To collect values entered in the Forecast: Current Day window. They apply to the Current Day (f_cday) table. The indexes are **ACD**, **ROW_DATE** and **SPLIT**.

Current Day Configuration Forecast Database Items

The following table describes the data items in the CMS Current Day Configuration Forecast database table.

Database Item	Description	Column Type	Length
ACD (index)	ACD number for which data was collected.	smallint	2 bytes
CHANGE	Additional change factor (percent).	smallfloat	4 byte <i>Informix</i> floating point type
CHPROF	Number of the call handling profile to use.	smallint	2 bytes
FMETHOD	Type of trending to use for forecast. Values are 0 = none, 1 = seasonal, 2 = current trending.	smallint	2 bytes
HDATE1-4	Date of first (HDATE1), second (HDATE2), third (HDATE3), and fourth (HDATE4) days of historical data to be used.	date	4 byte <i>Informix</i> date
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
SPLIT (index)	Split/skill number for which data was collected	smallint	2 bytes

Database Item	Description	Column Type	Length
TRENDBASE	Base date for seasonal trending.	date	4 byte <i>Informix</i> date
WT1-4	Weight given to date 1 (WT1), date 2 (WT2), date 3 (WT3), and date 4 (WT4), respectively.	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each of the Current Day Configuration database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	<i>DEFINITY</i> ECS R5	<i>DEFINITY</i> ECS R6
ACD	X	X	X	X	X	X	X
CHANGE	X	X	X	X	X	X	X
CHPROF	X	X	X	X	X	X	X
FMETHOD	X	X	X	X	X	X	X
HDATE1-4	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X
TRENDBASE	X	X	X	X	X	X	X
WT1-4	X	X	X	X	X	X	X

Current Day Forecast Report Database Items

Overview

The Current Day Forecast Report database item descriptions are **historical** items used specifically to collect values displayed in the Current Day Forecast report, and they apply to the Current Day Report (`f_cdayrep`) table. The indexes are **ACD**, **ROW_DATE** and **SPLIT**.

Forecast data for a split/skill are automatically generated when the Forecast Manager runs (if you have also completed a Current Day Configuration for the split/skill).

Current Day Forecast Report Database Items

The following table describes the data items in the CMS Current Day Forecast Report database table.

Database Item	Description	Column Type	Length
ACD (index)	ACD number for which data was collected.	smallint	2 bytes
AGOCC	Objective maximum percentage of time that an agent will be on ACD calls (agent occupancy).	smallfloat	4 byte <i>Informix</i> floating point type
AVGAGSERV	Objective average number of seconds for an agent to service a call.	smallint	2 bytes
AVGSPEEDANS	Objective average speed of answer in seconds for this type of call.	smallint	2 bytes
FCALLS	Number of forecast calls carried.	integer	4 bytes
INTRVL	Length of intrahour interval (15, 30, or 60). INTRVL applies to intrahour tables only.	smallint	2 bytes
NUMAGREQ	Number of agents required to handle FCALLS .	smallint	2 bytes

Database Item	Description	Column Type	Length
RAGOCC	Resulting maximum percentage of time that an agent will be on ACD calls.	smallfloat	4 byte <i>Informix</i> floating point type
RAVGSPEEDANS	Resulting average speed of answer in seconds for this type of call.	smallint	2 bytes
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
RSERVLEVELP	Resulting percentage of calls to be handled within SERVLEVELT seconds.	smallfloat	4 byte <i>Informix</i> floating point type
SERVLEVELP	Objective percentage of calls to be handled within SERVLEVELT seconds.	smallfloat	4 byte <i>Informix</i> floating point type
SERVLEVELT	Number of seconds within which SERVLEVELP percent of calls are to be answered (service level time).	smallint	2 bytes
SPLIT (index)	Split/skill number for which data was collected.	smallint	2 bytes
STARTTIME	Start time of the intrahour interval for which data was collected. STARTTIME applies only to the interval table.	smallint	2 bytes

ECS Cross-Reference

The following table lists which of the ECS releases support each Current Day Report database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	<i>DEFINITY</i> ECS R5	<i>DEFINITY</i> ECS R6
ACD	X	X	X	X	X	X	X
AGOCC	X	X	X	X	X	X	X
AVGAGSERV	X	X	X	X	X	X	X
AVGSPEEDANS	X	X	X	X	X	X	X
FCALLS	X	X	X	X	X	X	X
INTRVL	X	X	X	X	X	X	X
NUMAGREQ	X	X	X	X	X	X	X
RAGOCC	X	X	X	X	X	X	X
RAVGSPEEDANS	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
RSERVLEVELP	X	X	X	X	X	X	X
SERVLEVELP	X	X	X	X	X	X	X
SERVLEVELT	X	X	X	X	X	X	X
SPLIT	X	X	X	X	X	X	X
STARTTIME	X	X	X	X	X	X	X

Call Record Database Items

Overview

The Call Record database item descriptions are **historical** items that apply specifically to the Call Record (`call_rec`) table. The indexes are **ACD** and **ROW_DATE**.

Call Record Database Items

The following table describes the database items in the CMS Call Record database table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
ACWTIME	The time spent, in seconds, in After Call Work (ACW) associated with this call by the answering agent in this segment.	integer	4 bytes
AGT_RELEASED	Agent released or dropped the split/skill or direct agent ACD call. This is always true for ACD calls the agent transferred or conferenced. (0=NO, 1=YES). Available on Generic 3 and newer switches.	smallint	2 bytes
ANSHOLDTIME	The total time, in seconds, the call was put on hold by the answering agent in this call segment. In agent-to-agent calls, ANSHOLDTIME is accrued for the answering agent if the agent puts the call on hold, but not for the other agent (who continues to accrue talk time). For Generic 2.1 switches, hold time is accrued only for ACD calls. For Generic 2.2 and Generic 3 switches, and <i>DEFINITY</i> ECS Release 5, hold time is accrued for any type of call.	integer	4 bytes
ANSLOGIN	Login ID of the agent who answered the call in this segment. This field is blank for unmeasured extensions when EAS is not active.	char(9)	9 byte ASCII text string

Database Item	Description	Column Type	Length
ANSREASON	The reason code (0 through 9) associated with the answering agent's mode, if the agent is in the AUX mode. For agents in AUX on switches with releases prior to the ECS or switches that do not have EAS and reason codes active, ANSREASON is always 0.	smallint	2 bytes
ASSIST	Whether or not the answering agent in this segment requested supervisor assistance on this call. Valid values for ASSIST are 0=NO, 1=YES.	smallint	2 bytes
AUDIO	Whether or not an agent in this segment reported an audio difficulty problem. Valid values for AUDIO are 0=NO, 1=YES.	smallint	2 bytes
CALLID	A unique number assigned to this call and all its call segments. For conferenced/transferred calls, two (or more) calls are tied together. When the entire call is recorded, one call ID is used to tie together all call segments. In "meet-me" conferences, this may result in a "later" segment of the call starting earlier than the first segment. Call IDs are not necessarily strictly sequential, but will be unique for calls over a day.	integer	4 bytes
CALLING_II	Information Indicator (II) digits associated with the call. These digits are a two-digit string provided by the ISDN to indicate the type of originating line of the caller. These digits supply information about the originator location; for example, pay phone, hospital, or prison. The column is blank if the call does not contain II digits. Available on the ECS and newer switches.	char(2)	2 byte ASCII text string
CALLING_PTY	Calling party identification, which is the Automatic Number Identification (ANI)/Station Identification (SID) (for Generic 2.2 or Generic 3 Version 4 switches or the ECS with ISDN ANI delivery), extension or trunk equipment location identifying the originator of the call. This field is blank if the trunk is not measured or, for internal calls, if the originating extension is not measured. (Up to 12 digits in this field.)	char(12)	12 byte ASCII string

Database Item	Description	Column Type	Length
CONFERENCE	Whether or not the answering agent initiated a conference on this segment. Valid values for CONFERENCE are 0=NO, 1=YES. Available on the ECS, Generic 2.2 and Generic 3 switches.	smallint	2 bytes
CONSULTTIME	The time an agent talked on any outbound call while in AUX work, ACW, or in Other with a call on hold. This includes the time the originating agent spent talking to the destination party while establishing a conference or transferring a call. (This is the time between presses of the transfer or conference button.) It includes wait time if the agent is calling a Vector Directory Number (VDN) or split/skill extension, but the wait time can be subtracted out by subtracting the DISPTIME item from CONSULTTIME .	integer	4 bytes
DA_QUEUED	Whether or not the call was queued as a direct agent call Valid values for DA_QUEUED are 0=NO, 1=YES. Applies to the ECS and Generic 3 switches only.	smallint	2 bytes
DIALED_NUM	Number the caller dialed (up to 24 digits). This will be the VDN for inbound vectoring calls, blank for inbound calls without vectoring, and dialed digits for outbound calls.	char(24)	24 byte ASCII text string
DISPIVECTOR	Number of the first vector associated with the disposition VDN (DISPVDN).	smallint	2 bytes
DISPOSITION	Represents the call disposition and indicates whether the call in the segment was: 1=connected (CONN, non-ACD call to a measured agent) 2=answered (ANS, split/skill or direct agent call answered by an agent) 3=abandoned (ABAN) 4=interflowed (IFLOW) 5=forced busy (FBUSY) 6=forced disconnect (FDISC) 7=other (OTHER)	smallint	2 bytes

Database Item	Description	Column Type	Length
DISPOSITION (contd)	<p>A connected call is a non-ACD call to a measured agent for which CMS receives an indication that the call was connected.</p> <p>An answered call is any split/skill or direct agent ACD call for which CMS receives an indication that the call was answered by an agent and was not a phantom abandon.</p> <p>An abandoned call is any ACD call in which a caller hangs up before receiving an answer from an agent and for which CMS receives notification that the caller abandoned. Phantom abandons (PHANTOMABNS) are included as abandoned calls.</p> <p>Interflowed calls are calls that are interflowed to an off-switch destination.</p> <p>Forced busy calls are calls that CMS records as BUSYCALLS for the trunk group that carried them. These calls can be VDN calls that received a forced busy from the vector command or, on the ECS and G3 switches, a split/skill call for a nonvector-controlled split that received a busy indication from the switch because the split queue was full.</p> <p>For Generic 2.2, Generic 3 Version 2 and newer Generic 3 switches, and the <i>DEFINITYECS</i>, forced disconnect calls are VDN calls that are disconnected by the switch due to the execution of a disconnect vector command. For the ECS, and Generic 3 Version 2 and later Generic 3 switches, forced disconnect calls also include calls disconnected because of the vector disconnect timer or because they reached the end of vector processing without being queued. For Generic 3 Version 1 switches, forced disconnect calls are calls that were given a forced disconnect announcement and listening to the entire announcement, then were disconnected by the switch.</p> <p>Other calls include any other calls that do not fall into categories such as answered or abandoned. See definitions for individual tables for OTHERCALLS.</p>	smallint	2 bytes

Database Item	Description	Column Type	Length
DISPPRIORITY	Priority the call had at its disposition in this segment. Priorities can be 1=NO or 2=YES (without vectoring), or 3=LOW, 4=MED, 5=HIGH, or 6=TOP (with vectoring). If the call never gets queued to a split/skill, the priority will not be set. For the ECS, and Generic 3 switches with vectoring, calls directed to split/skills using "route to" or "messaging split/skill" commands and calls directly routed to splits/skills without going through a vector will have MED (no priority) or HIGH (priority) priority, depending on the class of restriction of the originator of the call (agent, extension, trunk group, or VDN).	smallint	2 bytes
DISPSKLEVEL	The skill level (1 through 16) associated with the skill for which the agent answered the call or, for calls that abandoned from ringing or from a direct agent queue, with the agent from whom the call abandoned.	smallint	2 bytes
DISPSPLIT	Number of the split/skill associated with the call at its disposition in this call segment. Calls that were not queued to a split or skill at the time of disposition will have DISPSPLIT set to null. Calls that were queued to an unmeasured split/skill at the time of disposition will have DISPSPLIT set to zero.	smallint	2 bytes
DISPTIME	Wait time (in the vector, in queue, and ringing) until the disposition is recorded in DISPOSITION for the segment. For extension calls made directly to agents (not through a VDN), this will always be zero.	integer	4 bytes
DISPVDN	Number of the VDN associated with the call at its disposition for this call segment. DISPVDN will be blank for calls that are not associated with a VDN at their disposition.	char(5)	5 byte ASCII text string

Database Item	Description	Column Type	Length
DURATION	The total time the trunk was in use. This is the overall trunk holding time from the beginning of the call segment until the caller is disconnected. For the first segment of a call, this will be the trunk holding time for the caller for the entire call (from seized until idle). With a transfer, the original trunk remains associated with both call segments until the call ends.	integer	4 bytes
EQLOC	Physical equipment location (trunk number) for which data was collected or for which the exception occurred. This will be blank if the trunk is not measured.	char(9)	9 byte ASCII text string
EVENT1-9	The number of times each event (stroke count) button (buttons 1 to 9) was entered for this call segment. Available with the ECS, Generic 2, and Generic 3 switches.	smallint	2 bytes
FIRSTVECTOR	Number of the first vector associated with the first VDN for the call segment. This will be blank if no vector is involved.	smallint	2 bytes
FIRSTVDN	The number of the first VDN associated with the call segment. This will be blank for calls not associated with a VDN.	char(5)	5 byte ASCII text string
HELD	The total number of times this call was placed on hold by the answering agent in this call segment. With agent-to-agent calls, this count is incremented for the agent who puts the call on hold, but not for the calling agent. (For the ECS, Generic 2.2, and Generic 3 switches, applies to all calls the agent put on hold.)	smallint	2 bytes
HOLDABN	Whether or not this call abandoned from hold in this call segment. Valid values for HOLDABN are 0=NO, 1=YES. With the ECS, Generic 2.2, and Generic 3 switches, this applies to all calls the agent put on hold.	smallint	2 bytes

Database Item	Description	Column Type	Length
LASTCWC	The last call work code (up to 16 digits) entered by the answering agent in this segment. This database item applies to Generic 2.2 and Generic 3 switches and the ECS only.	char(16)	16 byte ASCII text string
LASTDIGITS	Last set of collected digits sent to the CMS by the switch for this call. These are digits the switch sends to CMS when it executes a “collect” vector command. The digits may be digits the caller was prompted to enter, either through the prompting feature on the switch or through network-prompted digits (“caller-entered digits (CED)”), customer database-provided digits (“CDPD” from the network), or digits collected through a “converse” vector command. Available on ECS switches.	char(16)	16 byte ASCII text string
LASTOBSERVER	Login ID of the last agent who service-observed or bridged on to this call.	char(9)	9 byte ASCII text string
MALICIOUS	Indicates whether or not a malicious call trace was activated for this call segment. Valid values for MALICIOUS are 0=NO, 1=YES. Applies to the ECS, Generic 2, and Generic 3 (except Generic 3i Version 1) switches.	smallint	2 bytes
NETINTIME	The time the call spent in a VDN processing at another switch located elsewhere in the network. Requires the <i>DEFINITY ECS R6</i> .	integer	4 bytes
OBSERVINGCALL	Whether this call represents an agent observing or bridging on to an existing call. Valid values for OBSERVINGCALL are 0=NO, 1=YES.	smallint	2 bytes
ORIGHOLDTIME	The total time the call was put on hold by the originating agent. Requires the <i>DEFINITY ECS R6</i> .	integer	4 bytes
ORIGLOGIN	Login ID of the agent originating the call. This is used for calls an agent originates to another agent, to an on-switch extension, or to an external destination.	char(9)	9 byte ASCII text string

Database Item	Description	Column Type	Length
ORIGREASON	The reason code (0 through 9) associated with the originating agent's mode, if the agent is in the AUX mode. For agents in AUX on switches with releases prior to the ECS or switches that do not have EAS and reason codes active, ORIGREASON is always 0.	smallint	2 bytes
ROW_DATE	The date for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
ROW_TIME	The starting time for this segment.	smallint	2 bytes
SEGMENT	Number identifying the call segment. Segment numbers are from 1 up to the number of segments in the call.	smallint	2 bytes
SEGSTART	<i>UNIX</i> time and date when the call segment started. (<i>UNIX</i> time and date is the number of seconds since midnight, 01/01/70.) Call segments start when CMS receives the first message for the call, since each call segment represents a call. (When an agent transfers or conferences a call, the agent makes another call to bring about the transfer/conference.)	integer	4 bytes
SEGSTOP	<i>UNIX</i> time and date when the call segment ended. (<i>UNIX</i> time and date is the number of seconds since midnight, 01/01/70.) A call segment ends when all trunks and agents associated with the call segment have dropped off the call. This means that after call work time for the agent(s) is included when calculating the call segment stop time.	integer	4 bytes
SPLIT1	First split/skill the call queued to in the first VDN with which it was associated in the call segment.	smallint	2 bytes
SPLIT2	Second split/skill the call was also queued to in the first VDN with which it was associated in the call segment. Applies to Generic 2.2 with Expert Agent Selection (EAS), the ECS, and Generic 3 switches with vectoring only.	smallint	2 bytes

Database Item	Description	Column Type	Length
SPLIT3	Third split/skill the call was also queued to in the first VDN with which it was associated in the call segment. Applies to Generic 2.2 with EAS, the ECS, and Generic 3 switches with vectoring only.	smallint	2 bytes
TALKTIME	Total talk time for the answering agent in this segment.	integer	4 bytes
TKGRP	Trunk group number for which data was collected (or for which an exception occurred). This will be null if the trunk group carrying the call is not measured.	smallint	2 bytes
TRANSFERRED	Whether or not an answering agent initiated a transfer for this call segment. Valid values are 0=NO, 1=YES. For the ECS, Generic 2.2, and Generic 3 switches, TRANSFERRED includes all calls that are transferred.	smallint	2 bytes
UCID	The UCID is the Universal Call Identifier—a unique number assigned to this call segment within the customer network. Requires the <i>DEFINITY ECS R6</i> .	char(20)	20-byte ASCII text string

ECS Cross-Reference

The following table lists which of the ECS releases support the Call Record database items.

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
ACD	X	X	X	X	X	X	X
ACWTIME	X	X	X	X	X	X	X
AGT_RELEASED			X	X	X	X	X
ANSHOLDTIME	X	X	X	X	X	X	X
ANSLOGIN	X	X	X	X	X	X	X
ANSREASON	X	X	X	X	X	X	X
ASSIST	X	X	X	X	X	X	X
AUDIO	X	X	X	X	X	X	X
CALLID	X	X	X	X	X	X	X
CALLING_II						X	X
CALLING_PTY	X	X	X	X	X	X	X
CONFERENCE		X	X	X	X	X	X
CONSULTTIME	X	X	X	X	X	X	X
DA_QUEUED			X	X	X	X	X
DIALED_NUM	X	X	X	X	X	X	X
DISPIVECTOR	X	X	X	X	X	X	X
DISPOSITION	X	X	X	X	X	X	X
DISPPRIORITY	X	X	X	X	X	X	X
DISPSKLEVEL						X (EAS)	X
DISPSPLIT	X	X	X	X	X	X	X
DISPTIME	X	X	X	X	X	X	X
DISPVDN	X	X	X	X	X	X	X
DURATION	X	X	X	X	X	X	X
EQLOC	X	X	X	X	X	X	X

Call Record Database Items

Database Item	G2.1	G2.2	G3V1	G3V2/ G3V3	G3V4	DEFINITY ECS R5	DEFINITY ECS R6
EVENT1-9	X	X	X	X	X	X	X
FIRSTVECTOR	X	X	X	X	X	X	X
FIRSTVDN	X	X	X	X	X	X	X
HELD	X	X	X	X	X	X	X
HOLDABN	X	X	X	X	X	X	X
LASTCWC		X	X	X	X	X	X
LASTDIGITS						X	X
LASTOBSERVER			X	X	X	X	X
MALICIOUS	X	X	X (not G3iV1)	X	X	X	X
NETINTIME							X
OBSERVINGCALL			X	X	X	X	X
ORIGHOLDTIME							X
ORIGLOGIN	X	X	X	X	X	X	X
ORIGREASON	X	X	X	X	X	X	X
ROW_DATE	X	X	X	X	X	X	X
ROW_TIME	X	X	X	X	X	X	X
SEGMENT	X	X	X	X	X	X	X
SEGSTART	X	X	X	X	X	X	X
SEGSTOP	X	X	X	X	X	X	X
SPLIT1	X	X	X	X	X	X	X
SPLIT2		X (EAS)	X	X	X	X	X
SPLIT3		X (EAS)	X	X	X	X	X
TALKTIME	X	X	X	X	X	X	X
TKGRP	X	X	X	X	X	X	X
TRANSFERRED	X	X	X	X	X	X	X
UCID							X

Data Collection Period Database Items

Overview

The **Administrative** data collection period database items apply to the Daily Data Collection Period (*d_secs*), Weekly Data Collection Period (*w_secs*), and Monthly Data Collection Period (*m_secs*) tables. All items listed in the following table are included in all three tables, unless otherwise noted in the Column Type column. Any differences in the data format between the three data collection period tables are also noted in the Column Type column. The indexes are **ROW_DATE** and **ACD**. The tables indicate the number of seconds in the data collection period (daily, weekly, monthly).

Because the data contained in the Data Collection periods tables are administrative, the table does not rely on ECS releases.

Data Collection Period Database Items

The following table describes the data items in the CMS Data Collection Periods database tables.

Database Item	Description	Data Type	Column Type	Length
ACD (index)	The ACD number for which data was collected.	A	smallint	2 bytes
ROW_DATE	The day for which data was collected or the exception occurred.	A	date	4 byte <i>Informix</i> date
SECSPERDAY	The number of seconds in the daily data collection period	A	integer; only in <i>d_secs</i> table	4 bytes
SECSPERMN	The number of seconds in the monthly data collection period.	A	integer; only in <i>m_secs</i> table	4 bytes
SECSPERWK	The number of seconds in the weekly data collection period.	A	integer; only in <i>w_secs</i> table	4 bytes

Archiver Execution Status Database Items

Overview

The Archiver Execution Status database item descriptions apply specifically to items in the `arch_stat` table. The table contains status information on recent archiver executions, and displays status and the next run scheduled. The indexes are **ACD** and **ARCH_TYPE**. Because the data contained in the `arch_stat` table are administrative, the table is not dependent on ECS releases.

Archiver Execution Status Database Items

The following table describes the data items in the CMS Archiver Execution Status database table.

Database Item	Description	Data Type	Column Type	Length
ACD (index)	The ACD number for which data was collected.	A	smallint	2 bytes
ARCH_TYPE (index)	The type of archiver executions being run. Values are: 1 = interval, 2 = daily, 3 = weekly, 4 = monthly.	A	char(20)	20 byte ASCII text string
LAST_TIME	The last time the archiver execution was run.	A	char(20)	20 byte ASCII text string

Database Item	Description	Data Type	Column Type	Length
STATUS	<p>The status of the archiver execution. This field indicates if the execution was not run, is currently running, or has finished.</p> <p>Values are:</p> <p>1 = not run</p> <p>2 = finished, the archive was successful</p> <p>3 = finished, the archive had a failure</p> <p>4 = running.</p> <p>If the status field displays 3, for finished, but had a failure, you should consult the error log (<code>customer_log</code> table) and the archive log for the reason prior to troubleshooting.</p>	A	char(9)	9 byte ASCII text string

Customer Log Database Items

Overview

The Customer Log database item descriptions apply specifically to items in the `customer_log` table. The table contains customer error log information on recent archiver executions. The information includes the error code, the date the error occurred, the severity, the associated event, and a description of the error. Because the data contained in the `customer_log` table are administrative, the table is not dependent on ECS releases. The `customer_log` table is not backed up by the CMS Maintenance backup.

Customer Log Database Items

The following table describes the data items in the CMS Customer Log database table.

Database Item	Description	Data Type	Column Type	Length
ACD_ID	The ACD number for which data was collected.	A	integer	4 bytes
COUNTS	The number of occurrences of the error.	A	integer	4 bytes
DATE_OCCURRED	The date that the error occurred.	A	date	4 byte <i>Informix</i> date
DESCRIPTION	A text description of the error.	A	char(256)	256 byte ASCII text string
ERROR_CODE	The error code number.	A	integer	4 bytes
LAST_TIME	The last time the error occurred.	A	integer	4 bytes
SEVERITY	The level of severity of the error.	A	char(10)	10 byte ASCII text string

Agent Group Database Items

Overview

The Agent Group database items descriptions in the following table apply to items in the agroups table. The table contains dictionary information on agent groups. The indexes are ACD_NO, ITEM_TYPE, ITEM_NAME, and VALUE. Because the data contained in the agroups table are administrative and are used to define agent groups, the table is not dependent on ECS releases.

Agent Group Database Items

The following table describes the data items in the CMS Agent Group database dictionary table.

Database Item	Description	Data Type	Column Type	Length
ACD_NO (index)	The ACD number for which data was collected.	A	smallint	2 bytes
ITEM_TYPE (index)	The type "agent group."	A	char(20)	20 byte ASCII text string
ITEM_NAME (index)	The name of the agent group.	A	char(20)	20 byte ASCII text string
VALUE (index)	An agent login ID belonging to the agent group.	A	char(9)	9 byte ASCII text string

Synonyms Database Items

Overview

The Synonyms database item descriptions apply specifically to items in the `synonyms` table. The table contains dictionary synonyms. The indexes are **ACD_NO**, **ITEM_TYPE**, **ITEM_NAME**, and **VALUE**. Because the data contained in the `synonyms` table are administrative, and are used to define dictionary entries, the table is not dependent on ECS releases.

Synonyms Database Items

The following table describes the data items in the CMS Synonyms database dictionary table.

Database Item	Description	Data Type	Column Type	Length												
ACD_NO (index)	The ACD number for which data was collected.	A	smallint	2 bytes												
DESCR	The description, or definition, of the dictionary synonym.	A	char(5)	5 byte ASCII text string												
ITEM_TYPE (index)	The type of synonym. The following synonyms are used in reports: <table border="0"> <thead> <tr> <th>Type</th> <th>Database Item</th> </tr> </thead> <tbody> <tr> <td>acd</td> <td>ACD</td> </tr> <tr> <td>agname</td> <td>LOGID (login ID)</td> </tr> <tr> <td>tkgrp</td> <td>TKGRP (trunk group)</td> </tr> <tr> <td>split</td> <td>SPLIT</td> </tr> <tr> <td>vdn</td> <td>VDN (vector directory number)</td> </tr> </tbody> </table>	Type	Database Item	acd	ACD	agname	LOGID (login ID)	tkgrp	TKGRP (trunk group)	split	SPLIT	vdn	VDN (vector directory number)	A	char(20)	20 byte ASCII text string
Type	Database Item															
acd	ACD															
agname	LOGID (login ID)															
tkgrp	TKGRP (trunk group)															
split	SPLIT															
vdn	VDN (vector directory number)															

Database Item	Description	Data Type	Column Type	Length
ITEM_TYPE (index) (contd)	vector VECTOR aux_rsn AUXREASON (AUX work state reason) logout_rsn LOGOUTREASON (agent logout reason) cwc CWC (call work code) workmode WORKMODE (agent work mode in both Agent and Agent Trace tables) ag_origin ORIGIN ag_dir DIRECTION ag_dest DESTINATION ag_pref PREFERENCE tkstate TKSTATE (trunk state) tk_pri PRIORITY tk_queue QUETYPE (AUX work state reason) tk_vpri PRIORITY (only available with vectoring) tk_dir DIRECTION all_busy ALLINUSE slvl_chg SVCLEVELCHG per_chg PERIODCHG	A	char(20)	20 byte ASCII text string
ITEM_NAME (index)	The name of the synonym. There can be many ITEM_NAMES for a specific ITEM_TYPE .	A	char(20)	20 byte ASCII text string
STANDARD	Indicates if the item is a standard or custom synonym; Values are: 1 = standard, not 1 = custom.	A	smallint	2 bytes
VALUE (index)	The item name's corresponding value. Because each ITEM_TYPE can have many different ITEM_NAMES , a discrete value is assigned to each synonym ITEM_NAME .	A	char(9)	9 byte ASCII text string

ACD Shifts Database Items

Overview

The ACD shifts database item descriptions apply specifically to items in the `acd_shifts` table. The table contains information on ACD shift times and the maximum number of agents logged in for each shift. The indexes are **ACD** and **SHIFT_ID**. Because the data contained in the `acd_shifts` table are administrative, the table is not dependent on ECS releases.

ACD Shifts Database Items

The following table describes the data items in the CMS ACD Shifts database dictionary table.

Database Item	Description	Data Type	Column Type	Length
ACD (index)	The ACD number for which data was collected.	A	smallint	2 bytes
SHIFT_ID	The identification number of the ACD shift. Values are 1 - 4.	A	smallint	2 bytes
START_TIME	The ACD shift start time.	A	smallint	2 bytes
STOP_TIME	The ACD shift stop time.	A	smallint	2 bytes
MAX_AGENTS	The maximum number of agents logged in per shift.	A	smallint	2 bytes

Database Items

Overview

The Database Items descriptions apply specifically to items in the `dbitems` table. The table contains definitions for Dictionary standard and custom database items, constants, and calculations. The index is **ITEM_NAME**. Because the data contained in the `dbitems` table are administrative, and are used to define dictionary entries, the table is not dependent on ECS releases.

Database Items

The following table describes the data items in the CMS Database Items dictionary table.

Database Item	Description	Data Type	Column Type	Length
ITEM_TYPE (index)	The type of data for the row. Valid values are: dbase = database item calc = calculation constant = constant cust_def = customer-defined database item	A	char(8)	8 byte ASCII text string
ITEM_NAME (index)	The name of the data item. There can be many ITEM_NAMES for a specific ITEM_TYPE .	A	char(20)	20 byte ASCII text string
FORMULA	The formula for the database constant or calculation.	A	char(70)	70 byte ASCII text string
STANDARD	Indicates if the item is a standard or custom database item. Values are: 1 = standard, not 1 (null) = custom.	A	smallint	2 bytes
DESCR	The description of the database calculation, constant, or standard/custom database item.	A	char(50)	50 byte ASCII text string

Exceptions Historical Database Items

Overview

In the tables in this section, the database item **EXTYPE** lists numerical values associated with exception types. The database item **REASON** in the Database Collection Exceptions Database Items table lists numerical values associated with exception types. *CentreVu* CMS stores exception types using the numerical values, then translates the numbers into the text you see in standard exception reports.

To select specific exception types for a custom report, you must enter the numerical value(s) in the `Select rows where:` statement.

Agent Exceptions Database Items

The following table describes the historical Agent Exceptions database items. They apply to the Agent Exceptions (*agex*) table.

Database Item	Description	Column Type	Length																																
ACD	The ACD number for which data was collected.	smallint	2 bytes																																
EXTYPE	The type of exception that occurred: <table border="0"> <thead> <tr> <th>Numerical Value</th> <th>Type</th> </tr> </thead> <tbody> <tr><td>1</td><td>Time Available</td></tr> <tr><td>2</td><td>Time on inbound ACD call (min)</td></tr> <tr><td>3</td><td>Time on inbound ACD call (max)</td></tr> <tr><td>4</td><td>Time in after call work</td></tr> <tr><td>5</td><td>Time on outbound ACW call</td></tr> <tr><td>6</td><td>Time on inbound ACW call</td></tr> <tr><td>7</td><td>Time in AUX work</td></tr> <tr><td>8</td><td>Time on outbound AUX call</td></tr> <tr><td>9</td><td>Time on inbound AUX call</td></tr> <tr><td>10</td><td>Number of outbound ACW calls/agent</td></tr> <tr><td>11</td><td>Number of inbound ACW calls/agent</td></tr> <tr><td>12</td><td>Number of outbound AUX calls/agent</td></tr> <tr><td>13</td><td>Number of inbound AUX calls/agent</td></tr> <tr><td>14</td><td>Login identification</td></tr> <tr><td>15</td><td>Time ACD call spent on hold*</td></tr> </tbody> </table>	Numerical Value	Type	1	Time Available	2	Time on inbound ACD call (min)	3	Time on inbound ACD call (max)	4	Time in after call work	5	Time on outbound ACW call	6	Time on inbound ACW call	7	Time in AUX work	8	Time on outbound AUX call	9	Time on inbound AUX call	10	Number of outbound ACW calls/agent	11	Number of inbound ACW calls/agent	12	Number of outbound AUX calls/agent	13	Number of inbound AUX calls/agent	14	Login identification	15	Time ACD call spent on hold*	smallint	2 bytes
Numerical Value	Type																																		
1	Time Available																																		
2	Time on inbound ACD call (min)																																		
3	Time on inbound ACD call (max)																																		
4	Time in after call work																																		
5	Time on outbound ACW call																																		
6	Time on inbound ACW call																																		
7	Time in AUX work																																		
8	Time on outbound AUX call																																		
9	Time on inbound AUX call																																		
10	Number of outbound ACW calls/agent																																		
11	Number of inbound ACW calls/agent																																		
12	Number of outbound AUX calls/agent																																		
13	Number of inbound AUX calls/agent																																		
14	Login identification																																		
15	Time ACD call spent on hold*																																		

Database Item		Description	Column Type	Length
EXTYPE (Contd)	16	Number ACD calls placed on hold*	smallint	2 bytes
	17	Number ACD calls abandoned while on hold*		
	18	Time on outbound ACD call (min)		
	19	Time on outbound ACD call (max)†		
	20	Number calls transferred**		
	21	Time on external outbound ACW call†		
	22	Time on external outbound AUX call†		
	23	Time on direct agent call‡		
	24	Number external outbound ACW calls/agent**		
	25	Number external outbound AUX calls/agent**		
	26	Time ACD call spends ringing**		
	27	Multiple logins on same extension		
	28	Ringing call automatically redirected from agent		
	29	Agent logged out with active/held calls		
	30	Number of calls in direct agent queue‡		
	31	Time call waited in direct agent queue‡		
	32	Number calls abandoned from direct agent queue‡		
	34	Number calls outflowed from direct agent queue‡		
	38	Number of calls transferred		
	48	Logout attempt without valid reason code		
	49	Could not be logged in		
	59	AUX attempt without valid reason code		
	60	Time in AUX with reason code 0 (default)		
	61	Time in AUX with reason code 1		
	62	Time in AUX with reason code 2		
	63	Time in AUX with reason code 3		
	64	Time in AUX with reason code 4		
	65	Time in AUX with reason code 5		
	66	Time in AUX with reason code 6		
	67	Time in AUX with reason code 7		
	68	Time in AUX with reason code 8		
	69	Time in AUX with reason code 9		
98	Agent denied login to some skills			
99	Invalid call work code			

Database Item	Description	Column Type	Length
LOGID	Login ID of the agent who had the exception.	char(10)	10 byte ASCII text string
REASON_CODE	The reason code that the agent was in when the exception occurred.	smallint	2 bytes
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
ROW_TIME	Time at which the exception occurred.	smallint	2 bytes
SEQNO	The sequence number of this exception record.	integer	4 bytes
SPLIT	Split/skill in which the agent was doing work when the exception occurred.	smallint	2 bytes
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the agent's activity falls outside of that limit.	smallint	2 bytes
TIME	Limit, as a number of seconds, administered for timed exceptions types. An occurrence is logged against the threshold when the agent's activity falls outside of that limit.	smallint	2 bytes

*Available only on Generic 2 and Generic 3 ECS.

†Available only with Generic 2.2 ECS with the ASAI Gateway Interface feature and Generic 3 ECS with the ASAI feature.

** Available only with Generic 2.1 or later and Generic 3 ECS.

‡ Available only with Generic 3 ECS with the ASAI or EAS feature.

Disk Full Exceptions Database Items

The following table describes the historical Disk Full Exceptions database items. They apply to the Disk Full Exceptions (`fullex`) table.

Database Item	Description	Column Type	Length
PROC_NAME	The name of the process that failed because the disk was full.	char(30)	30 byte ASCII text string
ROW_DATE	The date at which the disk was full.	date	4 byte <i>Informix</i> date
ROW_TIME	The time at which the disk was full.	smallint	2 bytes
SEQNO	The sequence number of this record.	smallint	2 bytes
TASK_GRP	The activity that failed because the disk was full.	char(2)	2 byte ASCII text string

Split/Skill Exceptions Database Items

The following table describes the historical Split/Skill Exceptions database items. They apply to the Split/Skill Exceptions (`spex`) table.

Database Item	Description	Column Type	Length
ACD	The ACD number for which data was collected.	smallint	2 bytes
EXTYPE	The type of exception that occurred: Numerical Value Type 30 Number calls waiting 31 Time call has waited in queue 32 Number calls abandoned 33 Number intraflowed-in calls 34 Number intraflowed-out calls 35 Number interflowed-out calls 36 Number calls offered while queue full* 37 Number calls handled as backup† 38 Number calls transferred** 39 Average speed of answer (seconds) 40 Rolling average speed of answer (seconds) 41 Expected wait time (priority top) 42 Expected wait time (priority high) 43 Expected wait time (priority medium) 44 Expected wait time (priority low)	smallint	2 bytes
ROW_DATE (index)	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
ROW_TIME	Time at which the exception occurred.	smallint	2 bytes
SEQNO	The sequence number of this exception record.	integer	4 bytes
SPLIT	Split/skill in which the exception occurred.	smallint	2 bytes
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the split's/skill's activity falls outside of that limit.	smallint	2 bytes

Database Item	Description	Column Type	Length
TIME	Limit, as a number of seconds, administered for timed exception types. An occurrence is logged against the threshold when the split's/skill's activity falls outside of that limit.	smallint	2 bytes

*Available only on Generic 2 and Generic 3 ECS

†Available only with Generic 2.2 ECS with the ASAI Gateway Interface feature and Generic 3 ECS with the ASAI feature.

** Available only with Generic 2.1 or later and Generic 3 ECS.

Trunk Group Exceptions Database Items

The following table describes the historical Trunk Group Exceptions database s. They apply to the Trunk Group Exceptions (`tgex`) table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
EXTYPE	The type of exception that occurred: Numerical Value Type 50 Time trunk in use (min) 51 Time trunk in use (max) 52 Number of trunks in use 53 Time any trunk maintenance busy 54 Number of trunks maintenance busy 55 Length of time all trunks busy 56 Number trunk failures in group 57 Number failures on a single trunk 58 Audio difficulty on a trunk	smallint	2 bytes
EQLOC	Physical equipment location (trunk number) for which data was collected or the exception occurred.	char(9)	9 byte ASCII text string

Database Item	Description	Column Type	Length
LOGID	Login ID of the agent reporting audio difficulty.	char(10)	10 byte ASCII text string
ROW_DATE	Day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
ROW_TIME	Time at which the exception occurred.	smallint	2 bytes
SEQNO	The sequence number of this exception record.	integer	4 bytes
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the trunk group's activity falls outside of that limit.	smallint	2 bytes
TIME	Limit, as a number of seconds, administered for timed exception types. An occurrence is logged against the threshold when the trunk group's activity falls outside of that limit.	smallint	2 bytes
TKGRP	Trunk group number for which data was collected (or for which an exception occurred). This will be zero if the trunk group carrying the call is not measured.	smallint	2 bytes

VDN Exceptions Database Items

The following table describes the historical VDN Exceptions database items. They apply to the VDN Exceptions (`vdnex`) table. VDN exceptions are available only with the Vectoring feature.

Database Item	Description	Column Type	Length																														
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes																														
EXTYPE	<p>The type of exception that occurred:</p> <table border="1"> <thead> <tr> <th>Numerical Value</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>Time at agent (min)</td> </tr> <tr> <td>3</td> <td>Time at agent (max)</td> </tr> <tr> <td>30</td> <td>Number calls in an ACD split queue</td> </tr> <tr> <td>32</td> <td>Number calls abandoned while in vector</td> </tr> <tr> <td>33</td> <td>Number calls that flowed into VDN</td> </tr> <tr> <td>34</td> <td>Number calls that flowed out of VDN</td> </tr> <tr> <td>35</td> <td>Number calls interflowed out of VDN</td> </tr> <tr> <td>37</td> <td>Number calls handled by backup split</td> </tr> <tr> <td>71</td> <td>Time in vector (max)</td> </tr> <tr> <td>72</td> <td>Number calls forced busy</td> </tr> <tr> <td>73</td> <td>Number calls disconnected</td> </tr> <tr> <td>74</td> <td>Number unsuccessful lookahead attempts</td> </tr> <tr> <td>75</td> <td>Adjunct routing</td> </tr> <tr> <td>76</td> <td>Rolling average speed of answer</td> </tr> </tbody> </table>	Numerical Value	Type	2	Time at agent (min)	3	Time at agent (max)	30	Number calls in an ACD split queue	32	Number calls abandoned while in vector	33	Number calls that flowed into VDN	34	Number calls that flowed out of VDN	35	Number calls interflowed out of VDN	37	Number calls handled by backup split	71	Time in vector (max)	72	Number calls forced busy	73	Number calls disconnected	74	Number unsuccessful lookahead attempts	75	Adjunct routing	76	Rolling average speed of answer	smallint	2 bytes
Numerical Value	Type																																
2	Time at agent (min)																																
3	Time at agent (max)																																
30	Number calls in an ACD split queue																																
32	Number calls abandoned while in vector																																
33	Number calls that flowed into VDN																																
34	Number calls that flowed out of VDN																																
35	Number calls interflowed out of VDN																																
37	Number calls handled by backup split																																
71	Time in vector (max)																																
72	Number calls forced busy																																
73	Number calls disconnected																																
74	Number unsuccessful lookahead attempts																																
75	Adjunct routing																																
76	Rolling average speed of answer																																
ROW_DATE	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date																														
ROW_TIME	Time at which the exception occurred.	smallint	2 bytes																														
SEQNO	The sequence number of this exception record.	integer	4 bytes																														
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the VDN activity falls outside of that limit.	smallint	2 bytes																														
TIME	Limit, as a number of seconds, administered for timed exceptions types. An occurrence is logged against the threshold when the VDN activity falls outside of that limit.	smallint	2 bytes																														

Database Item	Description	Column Type	Length
VDN	VDN for which the exception occurred or that carried the malicious call.	char(6)	6 byte ASCII text string
VECTOR	Vector number associated with this VDN or for which the exception occurred.	smallint	2 bytes

Vector Exceptions Database Items

The following table describes the historical Vector Exceptions database items. They apply to the Vector Exceptions (`vecex`) table. Vector exceptions are available only with the Vectoring feature.

Database Item	Description	Column Type	Length																		
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes																		
EXTYPE	The type of exception that occurred: <table border="0"> <thead> <tr> <th>Numerical Value</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>30</td> <td>Number calls in an ACD split/skill queue</td> </tr> <tr> <td>32</td> <td>Number calls abandoned while in the vector</td> </tr> <tr> <td>72</td> <td>Number calls forced busy</td> </tr> <tr> <td>73</td> <td>Number calls disconnected</td> </tr> <tr> <td>74</td> <td>Number unsuccessful lookahead interflow attempts</td> </tr> <tr> <td>75</td> <td>Number unsuccessful adjunct routing attempts</td> </tr> <tr> <td>80</td> <td>Time in vector (min)</td> </tr> <tr> <td>81</td> <td>Time in vector (max)</td> </tr> </tbody> </table>	Numerical Value	Type	30	Number calls in an ACD split/skill queue	32	Number calls abandoned while in the vector	72	Number calls forced busy	73	Number calls disconnected	74	Number unsuccessful lookahead interflow attempts	75	Number unsuccessful adjunct routing attempts	80	Time in vector (min)	81	Time in vector (max)	smallint	2 bytes
Numerical Value	Type																				
30	Number calls in an ACD split/skill queue																				
32	Number calls abandoned while in the vector																				
72	Number calls forced busy																				
73	Number calls disconnected																				
74	Number unsuccessful lookahead interflow attempts																				
75	Number unsuccessful adjunct routing attempts																				
80	Time in vector (min)																				
81	Time in vector (max)																				
ROW_DATE	The date for which data was collected or which the exception occurred.	date	4 byte <i>Informix</i> date																		
ROW_TIME	Time at which the exception occurred.	smallint	2 bytes																		
SEQNO	The sequence number of this exception record.	integer	4 bytes																		

Database Item	Description	Column Type	Length
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the vector activity falls outside of that limit.	smallint	2 bytes
TIME	Limit, as a number of seconds, administered for timed exceptions types. An occurrence is logged against the threshold when the vector activity falls outside of that limit.	smallint	2 bytes
VECTOR	Vector number that this row represents, or for which the exception occurred.	smallint	2 bytes

Malicious Call Trace Exceptions Database Items

The following table describes the historical Malicious Call Trace Exceptions database items. The exception type will always be 90 = Malicious Call. They apply to the Malicious Call Trace Exceptions (`mctex`) table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
ANI_SID	Billing number or phone number from which the malicious call originated (available only if the switch has ANI/SID service).	char(8)	8 byte ASCII text string
EQLOC	Physical equipment location (trunk number) for which data was collected or for which the exception occurred.	char(9)	9 byte ASCII text string
II_DIGITS	Type of exception that occurred. The value for Malicious Call Trace exceptions is 90.	smallint	2 bytes
LOGID	Information Indicator digits. Indicates type of originating line the call used.	char(10)	10 byte ASCII text string

Database Item	Description	Column Type	Length
ROW_DATE	Login ID of the agent initiating a malicious call trace.	date	4 byte <i>Informix</i> date
ROW_TIME	The date for which data was collected or the exception occurred.	smallint	2 bytes
SEQNO	The sequence number of this exception record.	integer	4 bytes
SPLIT	Split/skill of the agent reporting the malicious call.	smallint	2 bytes
TKGRP	Trunk group number for which data was collected (or for which an exception occurred). This will be zero if the trunk group carrying the call is not measured.	smallint	2 bytes
VDN	The VDN for which the exception occurred or that carried the malicious call. Available on Generic 2, Generic 3, and newer switches with vectoring.	char(6)	6 byte ASCII text string

Data Collection Exceptions Database Items

The following table describes the historical Data Collection Exceptions database items. They apply to the Data Collection Exceptions ([linkex](#)) table.

Database Item	Description	Column Type	Length
ACD (index)	The ACD number for which data was collected.	smallint	2 bytes
DURATION	Length of time for which data collection was off.	integer	4 bytes
REASON	The reason for the interruption of data collection. The reasons may be as follows: Numerical Value Reason 91 Data collection started 92 Data collection of new translations started 93 Data collection turned off 94 Data collection busied out 95 Data collection timed out 96 Data collection clock was reset 97 Data collection session down	smallint	2 bytes
ROW_DATE	The day for which data was collected or the exception occurred.	date	4 byte <i>Informix</i> date
ROW_TIME	Time at which data collection was interrupted.	smallint	2 bytes
SEQNO	The sequence number of this exception record.	integer	4 bytes
THRESHOLD	Limit, as a number of occurrences, administered for the exception type. An exception occurs when the activity falls outside of that limit.	smallint	2 bytes

Glossary

3332 Platform	<i>Intel</i> [*] based CMS platform.
Abandoned Call	A call in which a caller hangs up before receiving an answer from an agent. The call could be queued to a split/skill or in a vector/vector directory number (VDN) or ringing at an agent before it is abandoned.
Abandoned Call Search	An Automatic call Distribution (ACD) capability that enables the system to make sure on certain trunk types that the caller is on the line before passing the call to an agent.
Acceptable Service Level	<p>A target value set to define the acceptable amount of time for an agent to answer a call. Target values are normally set as objectives by management.</p> <p>A percentage of calls answered within a set amount of time (for example, 80 percent of calls answered within 20 seconds).</p>
Access Permissions	Permissions assigned to a Call Management System (CMS) user so that the user can access different subsystems in CMS or administer specific elements (splits/skills, trunks, vectors, and so on) of the ACD. Access permissions are specified as read or write permission. Read permission means the CMS user can access and view data (for example, run reports or view the Dictionary subsystem). Write permission means the CMS user can add, modify, or delete data and execute processes.
ACD	See <i>Automatic Call Distribution</i> .
ACD Call	A call that queued to a split/skill and was answered by an agent in that split/skill, or a call that queued as a direct agent call and was answered by the agent for whom it was queued.
Acknowledgment	A window that requires you to confirm an action or to acknowledge a system message (for example, system going down, warning, or fatal error for the user window). This window cannot be moved, sized, or scrolled and disappears only when you confirm the message.
Action List	A menu in the upper-right corner of most user windows. The menu lists the actions available for that particular user window (for example, add, modify, delete, and so on). You select an action after entering necessary data in the user window.
Activate Agent Trace	From this window you can start CMS tracing of agent activities. These activities include all agent state changes until the trace is turned off. You can

*Intel is a registered trademark of Intel Corp.

activate traces for a maximum of 100 agents for an *Intel* computer, or 250 agents for a *Sun* computer, at any one time. You must activate an agent trace to obtain an Agent Trace report.

Active VDN Calls Also known as counted-calls to a VDN. A Call Vectoring capability available with G3V4 or later switches. Counted-calls to a VDN is a parameter of the “go to step” and “go to vector” commands that provides conditional branching (to a different step in the same vector or to a different vector) based on the number of incoming trunk calls a VDN is currently processing in a vector or at an agent.

ACW See *After Call Work*.

Add A CMS Action that adds the data entered in the given window to the database.

Adjunct/Switch Applications Interface (ASAI) An open application interface through which processors and switches can jointly provide services that require applications to initiate, receive, and control calls or make use of switch features. (See *Open Application Interface*.)

After Call Work (ACW) An agent state generally representing work related to the preceding ACD call. Going on-hook after an ACD call during MANUAL-IN operation places the agent in ACW. With Generic 3, ACW is also accessible by a button on the agent's set and may not be related to an ACD call.

Agent A person or Voice Response Unit (VRU) port that answers calls to an ACD split/skill. The agent is known to CMS by a login identification keyed into a voice terminal.

Agent Login ID A 1- to 4-digit number (Generic 2) or a 1- to 9-digit number (Generic 3) entered by an ACD agent from a voice terminal to activate the agent's position. Agent logins are required for all CMS-measured ACD agents.

Agent Occupancy The percentage of time that you are expecting or targeting for each split/skill agent to spend, while logged in, on ACD calls and in ACW.

Agent Position (EAS) The combination of the agent login ID and the skills the agent is assigned. Data are collected for the agent by skill, so the total work for the agent must be summed over all skills in which the agent worked.

Agent Position (Non-EAS) The combination of agent login ID and split the agent logged into. Agents logged into multiple splits have multiple positions associated with them. Call data are collected separately for each agent/split combination.

Agent Role	<p>A description of the kind of service an agent in multiple skills gives to one of their skills. This is a combination of call handling preference and skill/reserve levels. The five roles are:</p> <ul style="list-style-type: none">• <i>Top</i>: top agents logged into their highest priority skill.• <i>Allocated</i>: agents with percent allocation call handling preference administered (see the CentreVu® Advocate User Guide, Lucent Technologies 585-215-855).• <i>Backup</i>: agent is assigned to a skill, but not as the top skill.• <i>Roving</i>: an agent answers the skill's calls when this skill has the greatest need.• <i>Reserve</i>: an agent who normally does not answer calls for this skill answers calls in the skill because the skill has surpassed its pre-set over-threshold conditions (see the CentreVu® Advocate User Guide, Lucent Technologies 585-215-855).
Agent Skill	<p>An attribute that is associated with an ACD agent. Agent Skills can be thought of as the ability for an agent with a particular set of skills to handle a call that requires one of a set of skills. An agent can be assigned up to 20 skills. The meaning of each Agent Skill is defined by the customer. Examples of what could be considered skills are: the ability to speak a particular language or the expertise to handle a certain product.</p> <p>See <i>Primary Skill</i>, <i>Secondary Skill</i>, and <i>Skill Level</i>.</p>
Agent State	<p>A feature of agent call handling. Agent states are the different call work modes and call states an agent can be in (ACD, ACW, AVAIL, AUX, UNSTAFF, DACD, DACW, OTHER, UNKNOWN, RING). Data about these states is displayed in real-time and historical reports.</p> <p>See the definition of each state for additional information.</p>
Agent Terminal	<p>The voice terminal used by a call center agent.</p>
Agent Trace	<p>You must start an agent trace before you can obtain an Agent Trace report. You can select the dates that the report will cover. This report lists each agent activity and the time it occurred. The Agent Trace report can be helpful when evaluating how well individual agents are using their time.</p>
AI	<p>See <i>Auto-In</i>.</p>
Algorithm	<p>A prescribed set of well-defined rules or instructions for the solution of a problem; for example, the performance of a calculation, in a finite number of steps. Expressing an algorithm in a formal notation is one of the main parts of a software program.</p>

ANI	See <i>Automatic Number Identification</i> .
Announcement	A recorded voice message that normally tells the caller what destination the call has reached. The announcement also often tries to persuade the caller to stay on the line. With Call Vectoring, announcements can be part of a vector's call processing. An announcement is assigned to a vector by entering an announcement number.
Answered Call	The agent's state changes to ACD or Direct ACD (DACD). The term answered is used only for split/skill and direct agent ACD calls. (See Connected for non-ACD calls.) For manual answer agents, the call is answered when the agent selects the ringing line appearance. For automatic answer agents, the call is answered directly after the zip tone is applied.
API	See <i>Application Programming Interface</i> .
Application Programming Interface (API)	A set of related functions that a computer programmer uses to obtain some kind of service from another piece of software. Programmers of <i>Windows</i> based applications use the <i>Windows</i> API to create windows, draw text on the screen, access files, and perform all other services provided by <i>Windows</i> . Despite the use of the word application in this term, applications might not be the only software using an API; lower-level software components such as network drivers also have APIs, but these components are not "applications" and are not used directly by applications.
ASA	See <i>Average Speed of Answer</i> .
ASAI	See <i>Adjunct/Switch Applications Interface</i> .
Auto-Available Split	An ACD capability that ensures that after a power failure or a system restart, Voice Response Units (for example, the <i>CONVERSANT</i> [®] Voice Information System) are brought on line again immediately, without time-consuming reprogramming.
Auto-In (AI)	An ACD work mode that makes the agent available to receive calls and allows the agent to receive a new ACD call immediately after disconnecting from the previous call.
Automatic Call Distribution (ACD)	A switch feature using software that channels high-volume incoming and outgoing call traffic to agent groups (splits or skills). Also an agent state where the extension is engaged on an ACD call. See <i>Redirect On No Answer</i> and <i>Auto-Available Split</i> .

Automatic Number Identification (ANI)	A general industry term referring to knowledge of the calling party number (CPN). When the calling party is behind a switch, the number provided can be either a billing number for the switch or the station identification (SID) number.
AUX	See <i>Auxiliary Work</i> .
AUX Reason Codes	AUX reason codes enable a call center to track an agent's time more precisely when the agent is in the AUX state. The agent can specify exactly why the AUX state is used, such as lunch or meetings.
Auxiliary Work (AUX)	An agent work mode. For example, the agent is engaged in non-ACD work, is on break, in a meeting, or at lunch. An agent can reach this state by pressing the AUX WORK button or dialing the proper access code from the voice terminal. The agent can also reach the state by going off-hook to make or answer an extension call while in AVAIL or with a call on hold while in AI/MI mode.
AVAIL	See <i>Available</i> .
Available (AVAIL)	An agent state in which the extension is able to accept an ACD call. The agent enters this state by selecting the AI or MI work mode.
Average Agent Service Time	The average time you are expecting or targeting each agent to spend on an ACD call, including talk time and after-call-work time.
Average Speed of Answer (ASA)	<p>The average amount of time a caller waits before connecting to an agent. ASA is usually an objective set by your call center's management.</p> <p>The actual ASA for a split/skill includes the time spent in queue and the time ringing an agent. ASA for a VDN includes the time spent in vector processing, in queue, and the time ringing an agent.</p> <p>Interval ASA is used for Basic Call Management System (BCMS) and CMS reporting where the ASA is calculated on reporting interval boundaries and the ASA is cleared to zero at the start of each reporting interval.</p> <p>See also <i>Rolling ASA</i>.</p>
B Channel	An Integrated Services Digital Network (ISDN) channel that carries voice and circuit-switched data at a bandwidth of 64,000 bits per second. It will carry packet data at 19,200 bits per second (dial-up) or 64,000 bits per second (dedicated). "B" is an abbreviation of Bearer Channel.

Backup	The process of protecting data by writing the contents of the disk to an archive (or tape) that can be removed from the computer environment and stored safely.
Basic Call Management System (BCMS)	Call center management information available on the G3 switches and the Enterprise Communications Server (ECS) that can provide statistics on a limited number of agents, splits/skills, trunk groups and VDNs.
Basic Rate Interface	An ISDN channel arrangement that provides multiple channels over the existing two-wire local loop. This service has up to three channels: one D and a maximum of two B channels.
Best Service Routing (BSR)	A method of automatic call distribution between switches based on expected wait time. BSR can be used as a single-site as well as a multi-site feature.
BCMS	See <i>Basic Call Management System</i> .
BRI	See <i>Basic Rate Interface</i> .
BSR	See <i>Best Service Routing</i> .
Calculation	The abbreviated name (calculation name) for the formula calculation that generates the data for a field in a report.
Call-Based Items	The category of database items in CMS that are committed to the database after the call completes. If a call starts and ends in different intrahour intervals, all of the call-based data is recorded in the interval in which the call completed. Most database items are call-based.
Call Handling Preference	A parameter of agent administration in the EAS environment that specifies how calls are selected for the agent.
Call-Handling Profile	A set of objectives describing how you want a split/skill to handle calls. Call-handling profiles are part of the Forecasting subsystem.
Call Management System (CMS)	A software product used by business customers that have Lucent Technologies telecommunications switches/ECS and receive a large volume of telephone calls that are processed through the Automatic Call Distribution (ACD) feature of the switch/ECS. The CMS collects call-traffic data, formats management reports, and provides an administrative interface to the ACD feature in the switch/ECS.

Call Management System Query Language (CMSQL)	A tool that allows direct queries of the historical database. This tool is the interactive interface typically used to view the <i>Informix</i> database. For CMS purposes, CMSQL is used instead of <i>Informix</i> SQL.
Call Prompting	A switch feature that routes incoming calls based on information entered by the calling party, such as an account number. The caller receives an announcement and is prompted to select an option from those listed in the announcement or to enter numeric information using the telephone dial pad.
Call Vectoring	<p>A switch feature that provides a highly flexible method for processing ACD calls.</p> <p>A call vector is a set of instructions that controls the routing of incoming and outgoing calls based on current conditions. Examples of call vector conditions include time of day and the number of calls in queue.</p>
Call Work Code (CWC)	An ACD capability that allows the agent to enter a string of digits during or after the call and send them to the <i>CentreVu</i> ® Call Management System for management reporting.
Caller Needs	The reasons a customer calls your call center.
Calls Carried	Trunk data. The number of inbound/outbound calls that occupied the trunk during a given time period.
Cancel	A CMS action that cancels the current Backup, Restore, or Migration that is in progress.
Caution Condition	A graph term. The bars in the graph report change color or intensity indicating that the data being displayed met your defined first threshold limit.
Central Office (CO)	The location housing telephone switching equipment that provides local telephone service and access to toll facilities for long-distance calling.
CentreVu Supervisor	The Call Management System application for the <i>Microsoft Windows</i> operating environment.
Change Agent Skills	An agent's skills may be changed from CMS by using the Change Agent Skills Window. Through this window a user can quickly see what skills are currently assigned to a single agent and the user can easily change a single agent's skill assignment.
Close	Users can close a window using the Close command on the Control menu. Closing an application's main window is the same as quitting it.

CMS	See <i>Call Management System</i> .
CMS Large	One of two hardware configurations for the CMS product line. This configuration has 128MB RAM, 4GB of disk storage space, and one processor.
CMS Options	The optional features available with CMS. CMS options include graphics, forecasting, and vectoring.
CMS X-Large	One of two hardware configurations for the CMS product line. This configuration has 128MB RAM, 4GB of disk storage space, and two processors.
CO	See <i>Central Office</i> .
Configuration	The way that the computer is set up to allow for particular uses or situations.
CONN	See <i>Connected</i> .
Connected (CONN)	A trunk state in which a caller and an agent are connected on an ACD call.
Connected Call	A non-ACD call (that is connected to an agent through a VDN) for which <i>CentreVu CMS</i> receives an indication that the call rang or was answered.
Continue	A CMS action that resumes the suspended Restore or Migration operation.
CONVERSANT	<i>CONVERSANT</i> [®] is a powerful voice response system which interacts with the caller. This system may include: automated call routing, announcement storage, message retrieval, and callback.
Copy Group or Profile	A CMS action that creates a new group or profile from an existing one.
Current	A CMS screen-labeled key (SLK) that allows the user to cycle through open windows.
Current Interval	Represents the current intrahour interval, which can be 15, 30, or 60 minutes. The current interval is part of the real-time database. CMS starts collecting ACD cumulative data at the beginning of the interval (on the hour, half-hour, or quarter hour) and continues collecting ACD cumulative data until the end of the interval. When the current interval changes, all cumulative data is cleared and CMS begins counting cumulative data again starting from zero. The length of the interval is set in the <i>System Setup: Storage Intervals</i> window and is called the <i>intrahour interval</i> .

Current Wait Time	The time a call has waited for service in a call queue adjusted for queue priority.
Current Window	The user window in which you are actively working.
CWC	See <i>Call Work Code</i> .
D Channel	An ISDN channel that transmits at 16,000 bits per second. The signaling information connects, monitors, and disconnects calls. It can also carry low-speed packet data at 9,600 bits per second. "D" is an abbreviation of Data Channel.
DABN	See <i>Dequeued and Abandoned</i> .
DACD	See <i>Direct Agent ACD</i> .
DACW	See <i>Direct Agent ACW</i> .
Daily Data	Interval data that has been converted to a 1-day summary.
Data Collection Off	CMS is not collecting ACD data. Data already collected will not be lost when turning data collection off.
Data Collection On	CMS is collecting ACD data.
Data Points	Dates of historical data used in Forecasting. A data point should include data for each interval of the working day.
Database	A group of files that store ACD data according to a specific time frame: current and previous intrahour real-time data and intrahour, daily, weekly, and monthly historical data.
Database Item	A name for a specific type of data stored in one of the CMS databases. A database item may store ACD identifiers (split numbers or names, login IDs, VDNs, and so on) or statistical data on ACD performance (number of ACD calls, wait time for calls in queue, current states of individual agents and so on).
Database Management System (DBMS)	The software that manages access to structured data. For example, the <i>Microsoft SQL Server</i> is a database management system. Database management system can also be used generally to include PC database products such as <i>Microsoft Access</i> , as well as any other software that can provide data access services.

Database Tables	CMS uses these tables to collect, store, and retrieve ACD data. Standard CMS items (database items) are names of columns in the CMS database tables.
Date Format	<p>The standard format for entering dates on CMS reports. Acceptable formats are:</p> <ul style="list-style-type: none">• Month/day/year (for example, 3/21/93)• A “-” offset based on today's date (for example, -1 for yesterday)• Separating individual data entry items using a semicolon (for example, 3/21/93;3/23/93;3/25/93)• Entering ranges by placing a hyphen between entries (for example, 3/21/93-3/25/93). <p>When you specify a date for a weekly report, that date must correspond to the week start day selected in the System Setup—Storage Intervals window. If the date and day do not match, the message <i>No records found</i> displays in the status line.</p> <p>The month start date must be the first day of the month.</p>
DBMS	See <i>Database Management System</i> .
DDC	See <i>Direct Department Calling</i> .
Default Skill (Generic 2.2 with EAS)	Every skill that ends with a “0” is called a default skill, since every staffed agent in the skill group is logged into this skill by default. The default skill is the first skill for each skill group.
Delete	A CMS action that removes the entry on the window from the database.
Dequeued and Abandoned (DABN)	A trunk state in which the trunk quickly goes to idle after the caller abandons the call.
Designer Reports	Customized reports that can be created using <i>CentreVu</i> Supervisor’s Report Designer feature, and which are run from <i>CentreVu</i> Supervisor.
Dialed Number Identification Service (DNIS)	A network capability that identifies, for each call, the number dialed or the area from which the call originated (for example, a specific 800 number set up for a promotion).
Dictionary	A CMS subsystem that can be used to assign names to various call center elements such as login IDs, splits/skills, trunk groups, VDNs and vectors. These names appear on reports, making them easier to interpret. Dictionary also allows customized calculations to be created for use in reports.

DID	See <i>Direct Inward Dialing</i> .
Dimmed	Indicates unavailable or disabled. A dimmed menu item, button, or command is displayed in a lighter shade than the standard display and cannot be selected. See <i>Grayed Out</i> .
Direct Agent ACD (DACD)	An agent state in which the agent is on a direct agent ACD call.
Direct Agent ACW (DACW)	An agent state in which the agent is in the after call work (ACW) state for a direct agent ACD call.
Direct Agent Calling	An EAS capability that makes it possible for a caller to reach the same agent every time and still include the call as an ACD call in the management tracking of the call center. This is ideal for claims processing where a client needs to speak with the agent handling the claim. This flexibility ensures a high level of customer service without reducing management control.
Direct Department Calling (DDC)	A process of selecting an agent when more than one agent is available. With DDC, the call will go to the agent closest to the top of an ordered list. (This is a non-EAS option only.)
Direct Inward Dialing (DID)	An incoming trunk used for dialing directly from the public network into a communications system without help from the attendant.
DNIS	See <i>Dialed Number Identification Service</i> .
Driver	A dynamic link library that processes ODBC function calls received from the driver manager, submits the resultant SQL requests to a specific data source, and returns the results to the requesting application. If necessary, a driver modifies an application's request so that the request conforms to the syntax supported by the associated DBMS. See <i>Structured Query Language</i> , <i>Database Management System</i> , and <i>Dynamic Link Library</i> for more information.
Driver Manager	A dynamic link library that loads drivers on behalf of an application.
Dynamic Link Library	A dynamic link library is another name for a driver or a driver manager. A dynamic link library is specific to the DBMS of the data being accessed. For example, an <i>Informix</i> specific dynamic link library will be used to access data in an <i>Informix</i> database, such as the CMS database.
EAD	See <i>Expert Agent Distribution</i> .

EAS	See <i>Expert Agent Selection</i> .
ECS	See <i>Enterprise Communications Server</i> .
Enterprise Communications Server (ECS)	An Enterprise Communications Server is another name for a switch release, for example, the G3 ECS, or switch.
Entity	A generic term that refers to one of the following: Agent, Split/Skill, Trunk, Trunk Group, VDN, or Vector.
Error Message	A response from a program indicating that a problem has arisen or something unexpected has happened, requiring your attention.
EWT	See <i>Expected Wait Time</i> .
Exception	A type of activity in the ACD which falls outside the limits you have defined. An exceptional condition is defined in the CMS Exceptions subsystem, and usually indicates abnormal or unacceptable performance of the ACD (by agents, splits/skills, VDNs, vectors, trunks, or trunk groups).
Exception Reports	Display occurrences of conditions that fall outside the limits you defined.
Expected Delay	See <i>Expected Wait Time</i> .
Expected Wait Time (EWT)	An estimate of how long a caller will have to wait in queue to be served by a call center considering the current and past traffic, handling time, and staffing conditions. (Also referred to as <i>expected delay</i> .) The time spent in vector processing before being queued and the time spent ringing an agent with manual answering operation are not included in the EWT prediction. EWT is a switch-based calculation that can be used in vector processing decisions and can be viewed from CMS (release R3V4 and later). The EWT feature is available on the G3V4 switch, and the <i>DEFINITY®</i> Enterprise Communications Server (ECS) Release 5.
Expert Agent Distribution (EAD)	An EAS process that selects an agent when more than one agent is available. With EAS, a call will go to the most idle agent with the skill as primary (skill level one). If none are available, the call goes to an agent who is idle and has the skill as secondary. If none are available, the selection process continues with progressively lower skill levels until an idle agent is found.
Expert Agent Selection (EAS)	Expert Agent Selection (EAS) is an optional switch feature that builds on the power of the Call Vectoring and ACD features of the switch to match the skills required to handle a particular call to an agent who has at least one of the

skills that a caller requires. Agents are assigned a single set of work mode buttons, rather than one set per skill. This simplifies the agent's interface to the work mode buttons. When the "MI" (Manual In) or "AI" (Auto In) button is lit, the agent is available to take a call in any assigned skills or, in the case of Multiple Call Handling (MCH), in any MCH skills. The ACD queuing and the vector commands *Queue-to-Main* and *Check-Backup* are used to route a call to an agent with the appropriate skill to handle that call.

CMS collects data on skills in the same manner as it collects data on splits. Real-time Agent reports generally indicate the skill in which agents are currently working. Skill reports show the performance of the skill overall, displaying such items as the ASA, the number of calls, and the percentage of calls answered within the target service level for the skill.

CMS also reports VDN data by VDN skill preference, so that customers can assess the call center performance relative to calls requiring particular skills. CMS reports how many calls were handled, how long these calls waited for service, and the average talk time for calls queued to a particular skill preference in a particular VDN.

External Call	Calls made to off-switch destinations. This includes calls to other switches in a DCS network.
Extension Call	Extension calls are any calls originated by agents and non-ACD calls received by agents. For the Generic 2.2 and Generic 3 switches and the ECS, these include calls an agent makes to set up a conference or transfer.
FBUSY	See <i>Forced Busy</i> .
FDISC	See <i>Forced Disconnect</i> .
Find One	A CMS action that searches the database for entries that match the input value and displays the matching entries one at a time.
First Threshold	A graph term for the lower limit for a particular condition in a graph report. The bar(s) change color/intensity when the defined limit is met, notifying you that a Caution condition could exist.
Flex Agents	Agents who have a role of Roving, Backup, or Allocated. Top and Reserve agents are not Flex agents. see the CentreVu® Advocate User Guide, Lucent Technologies 585-215-855, for more information.
Forced Busy (FBUSY)	A trunk state in which the switch sends a busy signal to a caller when the call center is too busy to handle the incoming call.

Forced Disconnect (FDISC)	A trunk state in which the switch disconnects the caller.
Forced Multiple Call Handling (FMCH)	A feature available with the G3V4 switch, and the ECS, that, when activated for a split/skill, allows calls to be automatically delivered to an idle line appearance if the agent is in the Auto-In/Manual-In work mode and an unrestricted line appearance is available on the voice terminal, even if the agent is talking on an ACD call.
Function Call	Another name for a request for data generated within an ODBC-enabled application. The embedded SQL function in the application submits the request to the ODBC driver manager as a function call, where it is translated and a DBMS-specific driver manager (dynamic link library) is selected based on the type of data requested.
Get Contents	A CMS action that provides access to the members of an agent group, or to tasks on a shortcut or timetable.
Graphics	A CMS reporting option that allows you to view some reports in bar graph format. <i>CentreVu</i> Supervisor supports other graphic formats.
Grayed Out	When you do not have access to a menu or action list item, it will be grayed out (that is, dimmed or displayed in a different color from the rest of the menu or action list). <i>See Dimmed.</i>
Historical Database	A database that contains intrahour records for up to 62 days in the past, daily records for up to 5 years in the past, and weekly/monthly records for up to 10 years in the past for each CMS-measured agent, split/skill, trunk, trunk group, vector, and VDN.
Historical Reports	Reports that display past ACD data for various agent, split/skill, trunk, trunk group, vector, or VDN activities.
HOLD	A trunk state in which the agent has put the call on hold.
ICM	<i>See Inbound Call Management.</i>
IDLE	A trunk state indicating that the trunk is not in use.
II	<i>See Information Indicator.</i>
Inbound Call Management (ICM)	A set of switch and adjunct features using ASAI to enable the adjunct to provide automatic screen delivery and call routing.

Information Indicator (II)	A 2-digit code that identifies the type of originating line (for example: hotel or pay phone) for incoming ISDN primary rate interface (PRI) calls.
Informix	A relational database management system used to organize CMS historical data.
Informix SQL	A query language tool that is used to extract data from an <i>Informix</i> database. For the CMS historical database, CMSQL is used in place of <i>Informix SQL</i> .
Input Field	An input field is an area on a user window into which a <i>CentreVu</i> CMS user enters one or more valid field values. For example, the valid values for the input field <code>split</code> are integers from 1 through 255 (Generic 3r Version 2 to Version 4) and split names that have been assigned in the Dictionary subsystem.
Integrated Services Digital Network (ISDN)	A digital standard for telephony that enables, among other things, telephone, television, and computer signals on the same lines.
Interval-Based Items	A category of database items. These items generally represent the amount of time during a collection interval spent doing a particular activity. Interval-based items are updated throughout the collection interval and timing is restarted at the end of the interval. Interval-based items should only be used to show the amount of time in an interval for an activity or to calculate percentages of time spent in an interval. Interval-based items should not be used to calculate averages (such as average hold time).
Intrahour Interval	A 15-, 30-, or 60-minute segment of time starting on the hour. An intrahour interval is the basic unit of CMS report time.
ISDN	See <i>Integrated Services Digital Network</i> .
LAI	See <i>Look Ahead Interflow</i> .
LAN	See <i>Local Area Network</i> .
List All	A CMS action that lists all the entries that matched the current field values.
List Devices	A CMS action that lists all the devices which have been specified in the Backup/Restore Devices window.
Local Area Network (LAN)	A private interactive communication network that allows computers to communicate over short distances, usually less than one mile, at high data transfer rates from 1 Mbps to as high as 100 Mbps.

Logical Agent	<p>An EAS feature that associates the agent's login ID with the physical extension when the agent logs in. Properties such as the assigned skills, class of restriction, and coverage path are associated with the login ID rather than the physical extension. This allows agents to log in at any available set.</p> <p>The Logical Agent capability allows agents to be called by dialing their login IDs. Calls to login IDs may be treated as direct agent ACD calls, given the proper class of restriction, or may be treated as extension (personal) calls. Treating the calls as direct agent calls can be used to help distinguish business-related from personal calls.</p>
LOGOFF	<p>An agent work mode indicating that an agent has logged out and is not available to take ACD calls.</p>
LOGON	<p>An agent work mode indicating that an agent logged in or is staffed.</p>
Logout Reason Codes	<p>Logout reason codes enable an agent to specify the reason for logging out, such as the end of a shift or training.</p>
Look Ahead Interflow (LAI)	<p>A switch feature that can be used to balance the call load among multiple call centers. The LAI feature works with Call Vectoring and ISDN PRI trunks to intelligently route calls between call centers. This allows multiple call centers to share work loads, expands hours of coverage, and allows calls to be transparently handled by call centers in different time zones.</p>
Maintenance	<p>A CMS subsystem that is used for routine maintenance of the <i>CentreVu</i> CMS, such as backing up data, checking on the status of the connection to the switch, and scanning the error log.</p>
Maintenance Busy (MBUSY)	<p>A trunk state in which the trunk is out of service for maintenance purposes.</p>
Manual In (MI)	<p>An ACD work mode. The Manual In (MI) mode makes the agent available to receive an ACD call and automatically places the agent into the ACW state upon release from the call.</p>
MBUSY	<p>See <i>Maintenance Busy</i>.</p>
MCH	<p>See <i>Multiple Call Handling</i>.</p>
Measured	<p>A term that means an ACD element (agent, split/skill, trunk, trunk group, vector, VDN) has been identified to the switch as of interest to the CMS. The switch sends messages to CMS only for ACD elements that are measured. If the ACD element is not measured on the switch, no data is collected.</p>

Messages	Temporary windows used only for displaying information like “field help” and syntactical field errors. Message windows cannot be moved, sized, or scrolled and do not count in the user window count. Message windows are automatically removed when you correct the error or move to the next field.
MI	See <i>Manual In</i> .
MIA	See <i>Most Idle Agent</i> .
Migration	The process of migrating CMS data when upgrading your CMS. (Execute the “R3 Migrate Data” or “R2 Migrate Data” menu item after selecting System Setup from the main menu.) Migration is sometimes necessary in order to move the customer’s data to a new release. Migration is usually, but not always, driven by a hardware platform change.
Modify	A CMS action that changes the database entry to reflect the new values entered in the given window.
Monthly Data	Daily data that has been converted to a monthly summary.
More Help	More help displays a user window containing expanded information about the current primary user window or menu selection in <i>CentreVu</i> CMS. More help is for primary windows only, not output windows (like list all windows).
Most Idle Agent (MIA)	This is an ACD distribution method that maintains a queue of idle agents and distributes a call to the split/skill to the agent closest to the head of the queue who is not marked as “busy.” (“Busy” is defined as being in ACW, on an AUXIN/OUT call, or on an ACD call for another split/skill.) Agents in multiple splits/skills are in multiple “eligible agent” lists. There is one list for each skill. If MIA across splits/skills is enabled, agents are put at the bottom of all agent lists after completing an ACD call for any split/skill. If MIA across skills is <i>not</i> enabled, an agent who is on an AUXIN/OUT extension call from the AVAIL wait or on an ACD call for another split/skill continues to move up the list of eligible agents for other splits/skills. Agents in ACW may or may not be in the eligible agent lists, depending upon the setting of an option (<i>DEFINITY</i> ECS R5 and later).
Multi-Agent Skill Change Window	This window allows the user to perform the following administration actions: add up to 32 agents to a skill; move up to 32 agents from one skill to another; and remove up to 32 agents from a skill.
Multiple Call Handling (MCH)	Allows an agent to receive an ACD call while other calls are active on the agent’s station. Unless forced MCH is in operation, the agent must put the current call on hold and press AI/MI in order to receive another ACD call.

Multiple Split/Skill Queuing	With Call Vectoring, a call can be queued to up to three splits/skills at the same time. The first agent who becomes free in any of the splits/skills gets the call.
Multiuser Mode	Any administered CMS user can log into CMS. Data continues to be collected if data collection is “on.” This is the normal CMS operating mode.
Name (Synonym) Fields	Fields in which you may enter a name (synonym) that has been entered in the Dictionary subsystem (for example, names of agents, splits/skills, agent groups, trunk groups, vectors, VDNs).
Next	A CMS action that displays the next match found following a Find One, Next, or Previous action.
Night Service	A switch capability that assists calls that come in after business hours or on weekends to be automatically rerouted to a split, an announcement, or an alternate destination specifically set up for after-hours coverage.
Nonprimary Split/Skill	When a call is queued to multiple splits/skills, the second and third splits/skills to which the call queues in a VDN are called nonprimary splits/skills. They are also referred to as secondary and tertiary splits/skills, respectively.
Nonzero (0) Skill (Generic 2.2 with EAS)	Any skill that does not end in “0” is called a nonzero skill.
Normal Condition	A graph term in which the bars in the graph report are within your defined acceptable limits.
OCM	See <i>Outbound Call Management</i> .
ODBC	See <i>Open Database Connectivity</i> .
Open Application Interface	An open applications interface allows the same application to be supported by a wide range of products and avoids the need to develop specialized interfaces for new applications.
Open Database Connectivity (ODBC)	Open Database Connectivity is a standard application programming interface (API) for accessing data in both relational and non-relational databases.
Open Window	A user window that remains open because you have not yet closed it with the  SLK. An open window becomes the current window when it initially

appears on the screen or when you make it the current window using the **Current** SLK.

OTHER	An agent state in which the agent is working on a call for another split or skill, or has put a call on hold and has not chosen another work mode. When the link to the switch first comes up or when the agent has just logged in, the agent state is OTHER until the switch notifies CMS of the agent's state.
Outbound Call Management (OCM)	A set of switch and adjunct features using ASAI, that distributes outbound calls initiated by an adjunct to internal extensions (usually ACD agents).
PEC	See <i>Price Element Code</i> .
Percent (%) Within Service Level	The percentage of calls that were answered by an agent within a specific number of seconds. This observed percentage is used in reports as a statistical value. See <i>Acceptable Service Level</i> .
Previous	A CMS action that displays the previous match found.
Previous Interval	Represents one intrahour interval and is part of the real-time database. At the end of each intrahour interval, the contents of the current intrahour interval are copied to the previous intrahour interval portion of the real-time database.
Price Element Code (PEC)	The set of numbers that Lucent Technologies has assigned to each part that may be ordered.
Primary Skill	The skills assigned to an agent. Primary skills are the areas in which the agent has the most expertise. (This is used in G3V2 through G3V4 with EAS.) See <i>Agent Skill, Skill Level</i> .
Primary Window	The first window opened in response to a menu selection. A primary window may also generate another user window (secondary window). A primary window can be moved, sized, or scrolled, and counts in the window count.
Private Report	A custom or designer report that only the creator can access.
Pseudo-ACD	An area you create on your CMS to place previously backed-up ACD data. A pseudo-ACD is not a <i>live</i> (real) ACD and does not communicate with any switch.

Queue	A holding area for calls waiting to be answered in the order in which they were received. Calls in a queue may have different priority levels, in which case, calls with a higher priority are answered first.
QUEUED	A trunk state in which an ACD call has seized the trunk and is queued to a split/skill waiting for an agent to answer.
R3V2	See <i>Release 3 Version 2</i> .
R3V4	See <i>Release 3 Version 4</i> .
R3V5	See <i>Release 3 Version 5</i> .
R3V6	See <i>Release 3 Version 6</i> .
Read Permission	The CMS user with read permission can access and view data (for example, run reports or view the Dictionary subsystem). Read permission is granted from the User Permissions subsystem.
Real-Time Database	Consists of the current and previous intrahour data on each CMS-measured agent, split/skill, trunk, trunk group, vector, and VDN.
Real-Time Reports	Reports that display current ACD call activity on agents, splits/skills, trunks, trunk groups, vectors, and VDNs for the current or previous intrahour interval. Current intrahour interval real-time reports are periodically updated as data changes during the interval. Previous intrahour interval real-time reports show data totals for activity that occurred in the previous intrahour interval.
Recorded Announcements	Prerecorded greetings and information played to callers as they wait for service.
Redirect On No Answer (RONA)	An ACD capability that removes an unanswered call from the voice terminal at which it is ringing, busies out the port or makes the agent unavailable, and requeues the call at top priority or sends it to a VDN.
Refresh Rate	The number of seconds CMS should wait for each update of the real-time report data. A user's fastest allowable refresh rate is defined in the User Permissions—User Data window as a minimum refresh rate. The default refresh rate when a user brings up the report input window is the administered minimum refresh rate plus 15 seconds.

Release 3 Version 2 (R3V2)	The CMS software release immediately preceding R3V4. When the term R3V2 is used in this document, it applies to the R3V2 software, regardless of the hardware platform on which it is being run.
Release 3 Version 4 (R3V4)	R3V4 can refer to a software version of <i>CentreVu</i> CMS.
Release 3 Version 5 (R3V5)	R3V5 can refer to a software version of <i>CentreVu</i> CMS.
Release 3 Version 6 (R3V6)	R3V6 can refer to a software version of <i>CentreVu</i> CMS.
Request Agent	The <i>OpenLink</i> Request Agent resides on the client and interfaces with the ODBC driver via proprietary protocol. The Request Agent links the client applications and the <i>OpenLink</i> Request Broker on the server to facilitate transparent database access.
Request Broker	The <i>OpenLink</i> Request Broker is an ODBC driver manager that resides on the server. ODBC uses data source names as the link between the ODBC Request Broker and the relevant ODBC driver for a particular database. The <i>OpenLink</i> Request Broker provides the link between the user's applications and the DBMS itself by selecting a dynamic link library specific to the database being queried.
RING	<p>An agent state in which a call rings at an agent's voice terminal after leaving the queue and before the agent answers the call. (<i>This agent state is available only with Generic 2.2, and with Generic 3 and ECS.</i>)</p> <p>A trunk state in which a call is ringing at the agent's voice terminal.</p>
Rolling ASA	<p>Rolling ASA is a running weighted average calculation without regard to any interval boundaries. A rolling ASA calculated by the switch or ECS can be used, beginning with R3V4 CMS, for vector routing. Rolling ASA is calculated on the G3V4 switch, and the ECS, and sent to R3V4 and later CMS releases.</p> <p>An additional Rolling ASA calculated by the switch is also available as a real time database item for G3V4 and later G3 switches.</p>
RONA	See <i>Redirect On No Answer</i> .
Run	A CMS action that starts the process for the given window.
Screen-Labeled Key (SLK)	The first eight function keys at the top of your keyboard that correspond to the screen labels at the bottom of your terminal screen in <i>CentreVu</i> CMS. The screen labels indicate the function each key performs.

Scrolling	Moving backward and forward within a window in CMS.
Second Threshold	A graph term for the upper limit you enter for a particular condition in a graph report. When this limit is met, the bar(s) change color/intensity, indicating that a possible Warning condition exists.
Secondary Split/Skill (G3 Vectoring, G2.2 EAS)	The second split/skill the call queues to in a VDN is called the secondary split/skill.
Skill Group (Generic 2.2 EAS)	A group of ten skills. Each consecutive ten skills ending with digits 0 through 9 constitute a skill tens group. For example, skills 10-19 form a skill tens group, as do skills 340-349.
Secondary Skill	Skills assigned to an agent. Secondary skills are the areas in which the agent does not have extensive expertise, or is not the agent's preference. (Used in G3V2 through G3V4 with EAS.) <i>See Agent Skill, Skill Level.</i>
Secondary Window	A user window that is generated from a primary window. Secondary windows can be moved, sized, or scrolled and do not count in the user window count.
SEIZED	A trunk state in which the trunk is being used for either an incoming or an outgoing call.
Select Tables	A CMS action that allows you to select specific tables to Back Up or Restore.
Service Observing—Remote	A feature that allows a user to dial into the switch and monitor a call.
Service Observing—VDNs	A feature available with the G3V4 switch and the ECS that gives a voice terminal user the ability to monitor the treatment a call receives as it is processed by a VDN, routes to another VDN or agent, or transfers to another VDN or agent.
Shortcut	A CMS capability that enables a series of tasks to run immediately on your screen. Shortcut is a fast, easy way to select windows that you might look at every day.
Single-User Mode	Only one person can log into CMS. Data continues to be collected if data collection is "on." This mode is required to change some CMS administration.

Skill	An attribute that is assigned to an ACD Agent when EAS is enabled. An agent skill is a particular expertise or speciality enabling an agent to handle a call which requires someone with that particular area of expertise. You define skills based on specific customer needs and call center requirements.
Skill Hunt Group	When EAS is enabled, calls route to specific skill hunt groups. These skill hunt groups are usually based on the needs of your customers. Agents are not assigned to a skill group (like split hunt groups), but agents are assigned specific skills that become active when they log in.
Skill Level	A priority level from 1 (highest) to 16 (lowest) indicating an agent's level of expertise or ability to handle calls to the given skill. (ECS Version 5 and later.)
Skill, Primary	See <i>Primary Skill</i> .
Skill, Secondary	See <i>Secondary Skill</i> .
SLK	See <i>Screen-Labeled Key</i> .
Solaris System	See <i>UNIX System/Solaris System</i> .
SPARCserver 5	See <i>Sun Computers</i> .
SPARCserver 10	See <i>Sun Computers</i> .
SPARCserver 20	See <i>Sun Computers</i> .
SPARCstation	A workstation client for the <i>SPARCserver</i> .
Split	A group of extensions (referred to as agents) that receives special-purpose calls in an efficient, cost-effective manner. Calls automatically go to a split and can queue if no agents are available.
Split/Skill ACD Call	A split/skill ACD call is a call that routed to a split/skill and was answered by an agent in that split/skill.
SQL	See <i>Informix SQL</i> .
Staffed Agent	An agent who is currently logged in to the switch.
Standard Reports	The set of reports that are delivered with the CMS or <i>CentreVu</i> Supervisor software.

Station	An unmeasured voice terminal extension. An extension that is not currently staffed by an agent or that is a member of an unmeasured split/skill or hunt group.
Stop	A CMS action that stops the Restore or Migration that is in progress. Stop is also a vector command.
String Values	The descriptive words that appear on reports dealing with agents, splits/skills, and trunks. A <i>word</i> is used to describe the value of the data (for example, HOLD, AVAIL, YES).
Structured Query Language (SQL)	A language used to interrogate and process data in a relational database (such as <i>Informix</i>). See <i>Informix SQL</i> .
Submenu	A menu that appears as a result of a menu selection. All menu selections followed by a ">" have submenus.
Subsystem	Each CMS main menu selection (for example, Reports, Dictionary, System Setup, Exceptions, and so on) along with Timetable and Shortcut is referred to as a subsystem of the Call Management System.
Sun Computers	A host computer that is attached to a network and provides services other than simply acting as a store-and-forward processor or communication switch. The <i>Sun SPARCserver 5</i> , 10, and 20 and <i>Sun Ultra 5</i> and <i>Ultra Enterprise 3000</i> computers are capable of hosting the CMS application software.
Sun Microsystems	Manufactures the <i>Sun Ultra 5</i> and <i>SPARCserver 10</i> , the host computers for the <i>CentreVu CMS R3V6</i> .
Sun Ultra 5	See <i>Sun Computers</i> .
Switch	A private switching system providing voice-only or voice and data communications services (including access to public and private networks) for a group of terminals within a customer's premises.
System	A general term for a computer and its software and data.
System Setup	A CMS subsystem that allows users to initialize their CMS (for example, selecting the size of the intrahour interval based on how much data storage will be used for CMS data).
Technical Service Center (TSC)	Provisioning, maintenance, and helpline support for Lucent Technologies call center customers.

Terminal	<p>A combination of monitor (video display) and keyboard used to communicate with a computer to enter and display information.</p> <p>See <i>Agent Terminal</i>.</p>
Tertiary Split/Skill	<p>Generic 3 and ECS with vectoring, Generic 2.2 with EAS only. When a call is queued to multiple splits/skills, the third split/skill the call queued to in a VDN is called the tertiary split/skill.</p>
Time Format	<p>The standard format for entering times on CMS reports. Acceptable formats are:</p> <ul style="list-style-type: none">• A 12-hour time format with AM/PM (for example, 7:30AM, 5:00PM).• A 24-hour time format (for example, 7:30, 17:00).• A “-” offset based on the current interval date (for example, -1 for the previous hour or -0:30 for the previous half hour interval).
Timetable	<p>A task or group of tasks (such as reports) scheduled for completion at a time that is convenient and nondisruptive for your call center's operation.</p>
Top Skill	<p>The agent's top skill is the agent's first-administered, highest-level skill. This concept is the most useful when you have a Generic 3 switch (with EAS) and with agents who are using skill level call handling preference. In this case, the agent's top skill represents the skill for which the agent is most likely to receive a call. Agents for whom a given skill is the top skill are the agents that a skill supervisor can count on to handle calls for the skill.</p> <p>⇒ NOTE:</p> <p>This concept is not useful for agents using the greatest need call handling preference or for agents who are not Generic 3 (with EAS) agents. In these cases, the top skill data is still populated. When using the Generic 2.2 (with EAS) switch, the agent's top skill is always the “zero skill.” For non-EAS agents, the top “skill” is the split the agent has been logged into the longest.</p>
Translations	<p>A CMS action that requests a full set of information about measured entities from the switch.</p>
Trunk	<p>A telephone line that carries calls between two switches, between a Central Office (CO) and a switch, or between a CO and a phone.</p>
Trunk Group	<p>A group of trunks that are assigned the same dialing digits—either a phone number or a Direct Inward Dialed (DID) prefix.</p>
TSC	<p>See <i>Technical Service Center</i>.</p>

UCD	See <i>Uniform Call Distribution</i> .
Ultra Enterprise 3000	A Sun Microsystems computer capable of hosting CMS. See <i>Sun Computers</i> .
Uniform Call Distribution (UCD)	<p>A process that selects an agent when more than one agent is available. With UCD, the most idle agent for the skill/split receives the call.</p> <p>See <i>Direct Department Calling and Expert Agent Distribution</i>.</p>
Universal Call Identifier (UCID)	A number that uniquely identifies a call in a network of nodes that support UCID. This number will be a part of the records in the Call History feature of CMS.
UNIX System/ Solaris System	A multi-user computer operating system that supports CMS. A user can access the <i>UNIX</i> system from the Commands SLK.
UNKNOWN	<p>An agent state in which CMS does not recognize the current state.</p> <p>A trunk state in which CMS does not recognize the state of the trunk.</p>
UNSTAF	See <i>Unstaffed</i> .
Unstaffed (UNSTAF)	An agent state in which the agent is not logged in and, therefore, is not being tracked by CMS.
Update	A process used to modify a customer's existing software release in order to give the customer additional functionality or to fix a problem. The update process involves downloading CMS update files from a cartridge tape to hard disk, and installing the new files.
Upgrade	A process used to move an existing CMS customer from one release/load to another, giving the customer the additional functionality provided in the new release. Depending on which release the customer is upgrading from, and to, the upgrade may involve upgrading the software only, or may involve upgrading software and hardware. When an upgrade involves a major release, then data migration may be a necessary part of the upgrade procedure.
User Application	A user application is the software on the user's PC that the data is being accessed for. For example, a user may want to access data in the CMS database for use in <i>Microsoft Excel</i> . The user generates a query from the embedded data querying function in the application. The data returned from the CMS database can then be used to generate a spreadsheet.
User ID	The login ID for a CMS user.

User Permissions	A CMS subsystem that allows the CMS administrator to define user access permissions.
User Window	A window you can move, size, or scroll. It may contain input fields, reports, or help information.
VDN	See <i>Vector Directory Number</i> .
VDN Counted-Calls	Also known as counted-calls to VDN and active VDN calls. A Call Vectoring capability available with the G3V4 switch and the ECS. Counted-calls to VDN is a parameter of the “go to step” and “go to vector” commands that provides conditional branching (to a different step in the same vector or to a different vector) based on the number of incoming trunk calls currently in a VDN (in vector processing or at an agent).
VDN of Origin Announcement (VOA)	A short announcement that is assigned to a VDN through switch administration. The VOA identifies the origin or purpose of a call for the call center agent who answers the call.
VDN Skill Preference	A prioritized list of agent skills assigned to a VDN. Up to three skills can be assigned. VDN skill preferences are referred to in the vector as “1st,” “2nd,” and “3rd.” Vectors use VDN skills to queue calls based on your preference. CMS tracks calls by VDN skill preference.
Vector	A list of steps that process calls in a user-defined manner. The steps in a vector can send calls to splits/skills, play announcements and music, disconnect calls, give calls a busy signal, or route calls to other destinations. Calls enter vector processing via VDNs, which may have received calls from assigned trunk groups, from other vectors, or from extensions connected to the switch.
Vector Command	A vector step that describes the action to be executed for a call (for example, “Queue to main”, “check backup”, “disconnect”).
Vector Directory Number (VDN)	An extension number that enables calls to connect to a vector for processing. A VDN is not assigned an equipment location. It is assigned to a vector. A VDN can connect calls to a vector when the calls arrive over an assigned automatic-in trunk group, dial-repeating (DID) trunk group, or ISDN trunk group. The VDN by itself may be dialed to access the vector from any extension connected to the switch.
Vector Step	One processing step in a vector. A vector step consists of a command and one or more conditions or parameters.

Vector Step Condition	A condition accompanying a vector command that defines the circumstances in which the command will be applied to a call.
VOA	See <i>VDN of Origin Announcement</i> .
Voice Response Unit (VRU)	A switch that routes calls to a VRU adjunct computer that provides interactive voice related services to inbound callers.
Voice Terminal	A telephone set, usually with buttons, that gives an agent some control over the way calls are handled.
VRU	See <i>Voice Response Unit</i> .
Warning Condition	A graph term. The bars in the graph report change color or intensity indicating that the data being displayed met your defined second threshold limit.
Weekly Data	Daily data that has been converted to a weekly summary.
Window	Any rectangle on your CMS screen that encloses a menu, data entry fields, reports, or messages.
Window Count	The number of primary windows that can be open at any one time.
Write Permission	The CMS user can add, modify, or delete data and execute processes. Write permission is granted from the User Permissions subsystem.
Zero (0) Skill (Generic 2.2 with EAS)	Every skill that ends with a "0" is called a zero skill. The zero skill is the first skill for each skill group. This is the same as the default skill.

A

ABNCALLS 3-9, 3-39, 3-84
ABNCALLS1-10 3-10, 3-95
ABNQUECALLS 3-64, 3-84, 3-95
ABNRINGCALLS 3-10, 3-64, 3-84, 3-95
ABNTIME 3-10, 3-40, 3-84, 3-96
ABNVECCALLS 3-65
ACCEPTABLE 3-11, 3-96
ACD (index) 3-11, 3-40, 3-65, 3-77, 3-84, 3-96, 3-113,
3-116, 3-120, 3-125, 3-127, 3-130, 3-141, 3-142,
3-145, 3-146, 3-150, 3-154, 3-155, 3-157, 3-158,
3-159, 3-161
ACD Shifts Database Items 3-148
ACD_ID 3-144
ACD_RELEASE 3-40
acd_shifts table 3-8
ACDAUXOUTCALLS 3-11, 3-40
ACDCALLS 3-11, 3-40, 3-65, 3-77, 3-84, 3-96, 3-113
ACDCALLS1-10 3-12
ACDTIME 3-12, 3-40, 3-96, 3-114
ACWINCALLS 3-12, 3-41
ACWINTIME 3-12, 3-41
ACWOUTADJCALLS 3-12, 3-41
ACWOUTCALLS 3-12, 3-41
ACWOUTOFFCALLS 3-13, 3-41
ACWOUTOFFTIME 3-13, 3-42
ACWOUTTIME 3-13, 3-42
ACWTIME 3-13, 3-42, 3-96, 3-114, 3-130
Add Data Source screen 2-17
Add Login ID-Based Data Source 2-17
Add System Data Sources 2-16
Add System Data Sources screen. 2-16
ADJATTEMPTS 3-85, 3-96
ADJROUTED 3-85, 3-97
Administer ODBC Options. 2-22
Administration Interface
Add Data Source screen 2-17
Add System Data Sources screen 2-16
ODBC Data Source fields 2-15
ODBC Drivers screen. 2-16
OpenLink Setup input fields 2-21
System Data Sources Screen 2-17
Administrative data, definition 3-3
ag_actv table 3-8
Agent Database Items 3-39
Agent Exceptions Database Items 3-150
Agent Group Database Items 3-145
Agent Login/Logout Database Items 3-116
Agent Login/Logout ECS Cross-Reference 3-118
Agent Trace Database Items 3-120
Agent Trace ECS Cross-Reference 3-123
agex table 3-8
AGOCC 3-127
agroups table 3-8
AGT_RELEASED 3-120, 3-130
ALLINUSETIME 3-65

ANI_SID 3-159
ANSCONNCALLS1-10. 3-97
ANSHOLDTIME 3-130
ANSLOGIN 3-130
ANSREASON 3-131
ANSRINGTIME. 3-43
ANSTIME. 3-13, 3-85, 3-97
Application Programming Interface (API) 1-5
arch_stat table 3-8
ARCH_TYPE 3-142
Archiver Execution Historical Database Items 3-142
ASSIST 3-131
ASSIST_ACTV 3-120
ASSISTS 3-14, 3-43
AUDIO 3-65, 3-77, 3-131
AUXINCALLS 3-14, 3-43
AUXINTIME 3-14, 3-43
AUXOUTADJCALLS 3-14, 3-43
AUXOUTCALLS 3-14, 3-44
AUXOUTOFFCALLS 3-15, 3-44
AUXOUTOFFTIME. 3-15, 3-44
AUXOUTTIME 3-15, 3-44
AUXREASON 3-120
AVGAGSERV 3-127
AVGSPEEDANS 3-127

B

BACKUPCALLS 3-15, 3-65, 3-85, 3-98
BH_ABNCALLS 3-66, 3-98
BH_ACDCALLS 3-66, 3-98
BH_ACDTIME 3-98
BH_ALLINUSETIME 3-66
BH_BUSYCALLS 3-66, 3-98
BH_DISCCALLS 3-66, 3-99
BH_INCALLS. 3-66
BH_INTERVAL 3-66, 3-99
BH_INTIME 3-66
BH_OABNCALLS 3-66
BH_OACDCALLS 3-67
BH_OOTHERCALLS. 3-67
BH_OTHERCALLS. 3-67, 3-99
BH_OUTCALLS 3-67
BH_OUTTIME 3-67
BH_STARTTIME 3-68, 3-99
BH_VDNCALLS 3-99
BLOCKAGE 3-68
BSRPLAN 3-99
Building an ODBC Application 2-10
Busy Hour data. 3-3
BUSYCALLS 3-16, 3-68, 3-86, 3-100
BUSYTIME 3-16, 3-86, 3-100

C

Call Record Database Items 3-130
Call Record ECS Cross-Reference 3-139

Call Work Codes Database Items	3-113
Call Work Codes ECS Cross-Reference	3-115
call_rec table	3-8
Call-based data, definition	3-3
CALLER_HOLD	3-120
CALLID	3-131
CALLING_II	3-120
CALLING_PTY	3-121, 3-131
CALLSOFFERED	3-16
CentreVu Call Management System (CMS)	2-2
CHANGE	3-125
CHPROF	3-125
Client Trace	2-27
CMS Documents	P-3
CMS Feature Interactions	1-3
Permissions	1-4, 3-6
COMPLETED	3-68
Composing efficient database queries	1-3
CONFERENCE	3-16, 3-44, 3-121, 3-132
Configuration	
Desktop Computer Client	2-13
Add Login ID-Based Data Source	2-17
Add System Data Sources	2-16
Administer ODBC Options	2-22
New ODBC Data Source	2-18
View Installed ODBC Drivers	2-16
Server	2-5
Solaris client	2-8
Configure a New ODBC Data Source	2-18
Connect to a Data Source	2-25
CONNECTCALLS	3-68, 3-100
CONNECTTIME	3-100
CONNTALKTIME	3-101
CONSULTTIME	3-132
Conventions Used	P-2
COUNTS	3-144
Cumulative data, definition	3-3
Current Day Configuration ECS Cross-Reference	3-126
Current Day Configuration Forecast Database Items	3-125
Current Day Forecast Report Database Items	3-127
Current Day Report ECS Cross-Reference	3-129
Customer Log Database Items	3-144
customer_log table	3-8
Customer-created tables	1-3
CWC (index)	3-114

D

d_secs table	3-8
DA_ABNCALLS	3-45
DA_ABNTIME	3-45
DA_ACDCALLS	3-45
DA_ACDTIME	3-45
DA_ACWINCALLS	3-16, 3-45
DA_ACWINTIME	3-17, 3-46

DA_ACWOADJCALLS	3-46
DA_ACWOCALLS	3-17, 3-46
DA_ACWOFFCALLS	3-46
DA_ACWOFFFTIME	3-46
DA_ACWOTIME	3-17, 3-47
DA_ACWTIME	3-47
DA_ANSTIME	3-47
DA_OTHERCALLS	3-47
DA_OTHERTIME	3-47
DA_QUEUED	3-132
DA_RELEASE	3-48
dagent table	3-7
Data Collection Exceptions Database Items	3-161
Data Collection Period Historical Database Items	3-141
Database Items	
ACD Shifts	3-148
Agent	3-39
Agent Login/Logout	3-116
Agent Trace	3-120
Archiver Execution	3-142
Call Record	3-130
Call Work Codes	3-113
Current Day Configuration Forecast	3-125
Current Day Forecast Report	3-127
Customer Log	3-144
Data Collection Period	3-141
Database Items Table	3-149
Exceptions	
Agent	3-150
Data Collection	3-161
Disk Full	3-153
Malicious Call Trace	3-159
Split/Skill	3-154
Trunk Group	3-155
VDN	3-157
Vector	3-158
Split/Skill	3-9
Synonyms	3-146
Trunk	3-76
Trunk Group	3-63
VDN	3-93
Vector	3-83
Database Items Presentation	3-2
Database Items table	3-149
Database Logic Structure	3-5
Circular Structure	3-6
historical tables	3-5
Database Management System (DBMS)	1-5
Database queries	1-2
Database Table Names	3-7
Database table permissions	1-3
Database Tables	
acd_shifts	3-8
ag_actv	3-8
agex	3-8

agroups	3-8
arch_stat	3-8
call_rec	3-8
customer_log table	3-8
d_secs	3-8
dagent	3-7
dbitems	3-8
dcwc	3-8
dsplit	3-7
dtkgrp	3-7
dtrunk	3-7
dvdn	3-7
dvector	3-7
f_cday	3-8
f_cdayrep	3-8
fullex	3-8
hagalog	3-8
hagent	3-7
hcwc	3-8
hsplit	3-7
htkgrp	3-7
htrunk	3-7
hvdn	3-7
hvector	3-7
linkex	3-8
m_secs	3-8
magent	3-7
mctex	3-8
mcwc	3-8
msplit	3-7
mtkgrp	3-7
mtrunk	3-7
mvdn	3-8
mvector	3-7
spex	3-8
synonyms	3-8
tgex	3-8
vdnex	3-8
vecex	3-8
w_secs	3-8
wagent	3-7
wcwc	3-8
wsplit	3-7
wtkgrp	3-7
wtrunk	3-7
wvdn	3-8
wvector	3-7
DATE_OCCURRED	3-144
dbitems table	3-8
dcwc table	3-8
DEFLECTCALLS	3-86, 3-101
DEQUECALLS	3-17
DEQUETIME	3-17
DESCR	3-146, 3-149
DESCRIPTION	3-144
Desktop Computer Client Configuration	2-13

Add Login ID-Based Data Source	2-17
Add System Data Sources	2-16
Administer ODBC Options	2-22
Configure a New ODBC Data Source	2-18
ODBC Administration Interface	2-13
View Installed ODBC drivers	2-16
Installation	2-11
for Windows 95 and NT 4.0	2-11
for Windows NT 3.51	2-12
System Requirements	2-11
Test Connectivity	2-25
Connect to a Data Source	2-25
Disconnect From a Data Source	2-26
Desktop Computer Client Installation	2-11
DIALED_NUM	3-132
DIGITS_DIALED	3-121
DIRECTION	3-121
DISCCALLS	3-18, 3-69, 3-87, 3-101
Disconnect From a Data Source	2-26
DISCTIME	3-18, 3-101
Disk Full Exceptions Historical Database Items	3-153
DISPIVECTOR	3-132
DISPOSITION	3-132
DISPPRIORITY	3-134
DISPSPLIT	3-134
DISPTIME	3-134
DISPVDN	3-134
Document Numbers	P-3
dsplit table	3-7
dtkgrp table	3-7
dtrunk table	3-7
DURATION	3-121, 3-135, 3-161
dvdn table	3-7
dvector table	3-7

E

ECS Cross-Reference	
Agent Login/Logout	3-118
Agent Trace	3-123
Call Record	3-139
Call Work Codes	3-115
Current Day Configuration	3-126
Current Day Report	3-129
Split/Skill	3-35
Trunk	3-82
Trunk Group	3-74
VDN	3-109
Vector	3-92
EQLOC	3-135, 3-155, 3-159
EQLOC (index)	3-77
ERROR_CODE	3-144
EVENT_TIME	3-121
EVENT1-9	3-18, 3-48, 3-135
EXT_CALL_ORIG	3-121
EXTENSION	3-48

EXTN 3-116
 EXTTYPE 3-150, 3-154, 3-155, 3-157, 3-158

F

f_cday table 3-8
 f_cdayrep table 3-8
 FAILURES 3-69, 3-78
 FCALLS 3-127
 FIRSTVDN 3-135
 FIRSTVECTOR 3-135
 FMETHOD 3-125
 FORMULA 3-149
 fullex table 3-8

G

GOTOCALLS 3-88
 GOTOTIME 3-88

H

hagent table 3-7
 haglog table 3-8
 hcwc table 3-8
 HDATE1 3-125
 HELD 3-135
 HIGHCALLS 3-19
 HOLDABN 3-135
 HOLDABNCALLS 3-19, 3-48, 3-102
 HOLDACDCALLS 3-102
 HOLDACDTIME 3-48, 3-102
 HOLDACCALLS 3-19, 3-48, 3-102
 HOLDTIME 3-19, 3-49, 3-102
 hsplit table 3-7
 htkgrp table 3-7
 htrunk table 3-7
 hvdn table 3-7
 hvector table 3-7

I

I_ACDAUX_OUTTIME 3-20, 3-49
 I_ACDAUXINTIME 3-19, 3-49
 I_ACDOThERTIME 3-20, 3-49
 I_ACDDTIME 3-20, 3-49
 I_ACWINTIME 3-20, 3-50
 I_ACWOUTTIME 3-21, 3-50
 I_ACWTIME 3-21, 3-50
 I_ARRIVED 3-102
 I_AUXINTIME 3-21, 3-50
 I_AUXOUTTIME 3-21, 3-51
 I_AUXTIME 3-22, 3-51
 I_AUXTIME0 3-22
 I_AUXTIME1-9 3-22
 I_AVAILTME 3-22, 3-51
 I_DA_ACDDTIME 3-22, 3-51

I_DA_ACWTIME 3-23, 3-52
 I_INOCC 3-69, 3-78
 I_OTHERTIME 3-52
 I_OUTOCC 3-70, 3-78
 I_RINGTIME 3-24, 3-53
 I_STAFFTIME 3-24, 3-53
 I_TAUXTIME 3-24
 I_TAVAILTIME 3-24
 II_DIGITS 3-159
 INCALLS 3-70, 3-78, 3-88, 3-102
 INCOMPLETE 3-25, 3-53, 3-70,
 3-79, 3-88, 3-103, 3-114
 INFLAG 3-116
 INFLOWCALLS 3-25, 3-88, 3-103

Installation

Desktop Computer Client 2-11
 for Windows 95 and NT 4.0 2-11
 for Windows NT 3.51 2-12
 System Requirements 2-11
 Server 2-4
 Procedures 2-4
 System Requirements 2-4
 Solaris client 2-7
 Procedure 2-7
 System Requirements 2-7

Installation and Configuration Overview 2-3
 INTERFLOWCALLS 3-26, 3-89, 3-103
 Interval-based data, definition 3-3
 INTIME 3-70, 3-79, 3-89, 3-104
 INTRVL 3-26, 3-54, 3-71, 3-79,
 3-89, 3-104, 3-114, 3-127
 ITEM_NAME 3-145, 3-147, 3-149
 ITEM_TYPE 3-147
 ITEM_TYPE (index) 3-145, 3-149

K

KEYBD_DIALED 3-121

L

LAST_TIME 3-142, 3-144
 LASTCWC 3-136
 LASTDIGITS 3-136
 LASTOBSERVER 3-136
 linkex table 3-8
 LOGID 3-116, 3-152, 3-156, 3-159
 LOGID (index) 3-54, 3-121
 LOGIN 3-116
 LOGONSKILL2-5 3-116
 LOGOUT 3-117
 LOGOUT_DATE 3-117
 LOGOUT_REASON 3-117
 LOGOUTREASON 3-122
 LOOKATTEMPTS 3-89, 3-104
 LOOKFLOWCALLS 3-89, 3-104

LOWCALLS 3-27

M

m_secs table 3-8
magent table 3-7
MALICIOUS 3-136
Malicious Call Trace Exceptions Database Items 3-159
Maximum Interval Value data 3-3
MAXINQUEUE 3-27
MAXOCWTIME 3-27, 3-104
MAXSTAFFED 3-27
MAXTOP 3-27
MAXWAITING 3-105
MBUSYTIME 3-71, 3-79
MCT 3-122
mctex table 3-8
mcwc table 3-8
MEDCALLS 3-27
msplit table 3-7
mtkgrp table 3-7
mtrunk table 3-7
mvdn table 3-8
mvector table 3-7

N

NETDICCALLS 3-90
NETDISCCALLS 3-105
NETINCALLS 3-105
NETINTIME 3-105
NETPOLLS 3-90, 3-105
NOANSREDIR 3-28, 3-54, 3-105
NUMAGREQ 3-127
NUMTGS 3-105

O

O_ABNCALLS 3-29, 3-71, 3-80
O_ACDCALLS 3-29, 3-54, 3-71, 3-80
O_ACDTIME 3-29, 3-54
O_ACWTIME 3-29, 3-55
O_OTHERCALLS 3-29, 3-71, 3-80
OBSERVINGCALL 3-136
ODBC Administration Interface 2-13
ODBC Data Source dialog box 2-13
ODBC Data Source Fields 2-15
ODBC Driver 2-2
 Function 2-2
 Vendor 2-2
ODBC Driver Functionality 1-7
ODBC Drivers screen 2-16
ODBC feature, description 1-2
ODBC Functionality 1-5
ODBC Installation for Solaris 2-4
ODBC Login dialog box 2-25
ODBC performance impact 1-3

ODBC SDK 2.0 C++ Demo SQL Query dialog box 2-25
ODBC SDK C++ Demo screen 2-25
ODBC security 1-3
ODBC terminology 1-2
Open Database Connectivity (ODBC), definition 1-5
OpenLink Error Messages 2-27
OpenLink Multi-Tier ODBC driver 2-2
OpenLink Multi-Tier ODBC driver, description 1-2, 2-2
OpenLink ODBC Driver (MT) User's Guide and Reference Manual 2-3
OpenLink Request Agent 2-2
OpenLink Request Broker 2-2
OpenLink Setup Input Fields 2-21
ORIGHOLDTIME 3-136
ORIGLOGIN 3-136, 3-137
OTHERCALLS 3-30, 3-72, 3-80, 3-90, 3-105
OTHERTIME 3-30, 3-90, 3-106
OUTCALLS 3-72, 3-80
OUTFLAG 3-117
OUTFLOWCALLS 3-30, 3-32, 3-90, 3-106
OUTFLOWTIME 3-32, 3-90, 3-106
OUTTIME 3-72, 3-81

P

PERIOD 1-9 3-33
PERIOD1-9 3-106
PERIODCHG 3-33, 3-106
PHANTOMABNS 3-33, 3-55, 3-91, 3-107
Presentation
 Administrative data 3-3
 Call-based data 3-3
 Cumulative data 3-3
 Database Items 3-2
 ECS Reference Tables 3-4
 Interval-based data 3-3
PROC_NAME 3-153

R

RAGOCC 3-128
RAVGSPEEDANS 3-128
REASON 3-161
REASON_CODE 3-152
RECONNECT 3-122
Related Documents P-3
Relational database model 1-6
Remove a Data Source 2-18
Requesting Data in ODBC 1-7
RETURNCALLS 3-107
RINGCALLS 3-33, 3-55, 3-91, 3-107
RINGTIME 3-33, 3-55, 3-91, 3-107
ROW 3-158
Row Identifier data 3-3

ROW_DATE (index) . . .3-33, 3-55, 3-72, 3-81, 3-91,
 3-107, 3-114, 3-117, 3-122, 3-125, 3-128, 3-137,
 3-141, 3-152, 3-153, 3-154, 3-156, 3-157, 3-158,
 3-160, 3-161
 ROW_TIME 3-152, 3-153, 3-154, 3-156, 3-157, 3-158,
 3-160, 3-161
 ROW_TIME (index)3-137
 RSERVLEVELP3-128

S

SECPERWK3-141
 SECSPERDAY3-141
 SECSPERMN3-141
 SEGMENT3-137
 SEGSTART3-137
 SEGSTOP3-137
 SEQNO.3-152, 3-153, 3-154,
3-156, 3-157, 3-158, 3-160, 3-161
 Server
 Configuration2-5
 Installation
 Procedures2-4
 System Requirements2-4
 Server - Run ODBC2-6
 Server Configuration
 Debug Levels2-6
 Log Levels.2-6
 Run ODBC2-6
 Server Debug Levels.2-6
 Server Inicheck Utility2-27
 Server Log File2-27
 Server Log Levels2-6
 SERVICELEVEL3-34, 3-107
 SERLEVELP3-128
 SERVLEVELT3-128
 SETUPTIME3-72
 SEVERITY3-144
 SHORTCALLS3-73, 3-81
 SKILL1-33-108
 SKILLACWTIME1-33-107
 SKILLCALLS1-33-107
 SKILLTIME1-33-108
 SKILLTYPE3-117
 SKILLTYPE2-43-117
 SKLEVEL.3-117
 SKLEVEL2-203-118
 SLVLABNS3-34, 3-108
 SLVLOUTFLOWS3-34, 3-108
 Solaris Client
 Building an ODBC Application2-10
 Configuration2-8
 Installation.2-7
 Procedure2-7
 System Requirements2-7
 Test Connectivity2-9
 Possible Failure Causes2-9

Special Table data3-3
 spex table3-8
 SPLIT3-73, 3-122, 3-152, 3-154, 3-160
 SPLIT (index)3-34, 3-55, 3-118, 3-125, 3-128
 Split/Skill Database Items.3-9
 Split/Skill ECS Cross-Reference3-35
 Split/Skill Exceptions Database Items3-154
 SPLIT13-137
 SPLIT23-137
 SPLIT33-138
 SQL Data Sources dialog box2-25
 SQL queries.1-2, 2-2
 STANDARD.3-147, 3-149
 STARTTIME3-34, 3-56, 3-114, 3-122, 3-128
 STATUS3-143
 Structured Query Language1-6
 Structured Query Language (SQL)1-2, 2-2
 Supported Logins2-2
 SVCLEVELCHG3-34, 3-108
 Synonyms Database Items3-146
 synonyms table3-8
 System Data Sources screen.2-17

T

Table Names3-7
 Table-Level Permissions2-2
 Table-level permissions.1-2
 TALKTIME3-138
 TASK_GRP3-153
 Test Connectivity
 Desktop Computer Client2-25
 Connect to a Data Source.2-25
 Disconnect Form a Data Source2-26
 ODBC Login dialog box2-25
 ODBC SDK 2.0 C++ Demo SQL Query dialog box
2-25
 ODBC SDK C++ Demo screen2-25
 Solaris Client.2-9
 Possible Failure Causes2-9
 SQL Data Sources dialog box2-25
 tgex table3-8
 THRESHOLD3-152, 3-154,
3-156, 3-157, 3-159, 3-161
 TI_AUXTIME3-56
 TI_AUXTIME03-56
 TI_AUXTIME1-93-56
 TI_AVAILTIME3-57
 TI_OTHERTIME3-57
 TI_STAFFTIME3-58
 TIME3-152, 3-155, 3-156, 3-157, 3-159
 TKGRP3-138, 3-160
 TKGRP (index)3-73, 3-81
 TOPCALLS3-34
 TRANSFERRED3-34, 3-58, 3-73, 3-108, 3-122, 3-138
 TRENDBASE3-126
 TRKGRP3-156

Troubleshooting	2-27
Client Trace	2-27
OpenLink Error Messages	2-27
Server Inicheck Utility	2-27
Server Log File	2-27
Trunk Database Items	3-76
Trunk ECS Cross-Reference	3-82
Trunk Group Database Items	3-63
Trunk Group ECS Cross-Reference	3-74
Trunk Group Exceptions Database Items	3-155
TRUNKS	3-73

U

UCID	3-122, 3-138
Uses for ODBC data	1-7

V

VALUE	3-145, 3-147
VDISCCALLS	3-91, 3-108
VDN	3-73, 3-158, 3-160
VDN (index)	3-109
VDN Database Items	3-93
VDN ECS Cross-Reference	3-109
VDN Exceptions Database Items	3-157
vdnex table	3-8
vecex table	3-8
VECTOR	3-73, 3-158, 3-159
VECTOR (index)	3-91, 3-109
Vector Database Items	3-83
Vector ECS Cross-Reference	3-92
Vector Exceptions Database Items	3-158
View Installed ODBC Drivers	2-16

W

w_secs table	3-8
wagent table	3-7
wcwc table	3-8
Windows 95, NT 4.0 ODBC Installation	2-11
Windows NT 3.51 ODBC Installation	2-12
WMODE_SEQ	3-122
WORKCODE	3-123
WORKMODE	3-123
wsplit table	3-7
WT1	3-126
wtkgrp table	3-7
wtrunk table	3-7
wvdm table	3-8
wvector table	3-7

