

Advantage™ CA-Datcom® Database Server Option

User Guide

5.0



Computer Associates®

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

© 2003 Computer Associates International, Inc.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.



Contents

Chapter 1: Introduction

Audience	1-2
ODBC Architecture	1-3
JDBC Architecture	1-5
JDBC Architecture on z/OS Machines	1-7
Navigational Architecture	1-9
Advantage CA-Realia II Workbench SQL Architecture	1-11
Preprocessor C Architecture	1-13
Conventions	1-15
Related Publications	1-16

Chapter 2: Starting Advantage CA-Datacom Server

z/OS or OS/390 Startup JCL	2-2
VSE Startup JCL	2-3
Input Parameters	2-4

Chapter 3: Monitoring Advantage CA-Datacom Server

Administrator Responsibilities	3-1
Restarting a Server	3-1
Using Server Groups	3-2
Diagnosing Problems	3-2
Monitoring Workstation Connections	3-3
Monitoring User Requirements Tables	3-3

Chapter 4: Using the Communication Utility

MVS or OS/390 SVCOMPR JCL	4-2
VSE SVCOMPR JCL	4-3
SVCOMPR and Console Input Parameters	4-4
Sample Output from STATUS Command	4-8

Chapter 5: ODBC Driver Configuration and Administration

Administrator Tasks	5-1
ODBC Driver	5-1
ODBC Driver Manager	5-2
ODBC Administrator	5-2
Add Data Source	5-3
ODBC Setup	5-4
Limiting Lists of Tables and Views	5-10
ODBC Connection Options	5-11
Connecting to Advantage CA-Datcom/DB Using a Connection String	5-11
SQLBrowseConnect: Selective Server Specification	5-13
SQLDriverConnect: Selective Option Specification	5-14
SQLConnect: Automatic Option Specification	5-15
Default Data Source Name	5-15
Connection Window (Advantage CA-Datcom/DB Logon)	5-16

Chapter 6: JDBC Driver Configuration and Administration

Chapter 7: Using the Java Server

Installing the Java Server	7-1
Java Server Config	7-3

Chapter 8: Using the Translation Table Editor

Building a Translation Table	8-1
Editing a Translation Table	8-2
Selecting Code Pages	8-2
Editing Character Mapping	8-3

Chapter 9: Using the Advantage CA-Datcom Server Preprocessors

COBOL Preprocessor	9-1
Using SQL Statements	9-2
COBOL SQL Preprocessor Setup	9-2
Using the COBOL SQL Preprocessor	9-6
Linking Your DLL	9-10
Connecting and Disconnecting	9-10
Locking, Transactions, and Abends	9-10
Mixing SQL with Navigational Commands in COBOL Programs	9-11
C Preprocessor	9-11
Using SQL Statements	9-12

Chapter 10: Using Advantage CA-Datcom Navigational Commands

Using Navigational Commands with C	10-2
Sample Program	10-2
Using Navigational Commands with Advantage CA-Realia II Workbench	10-6
DBNTRY Programming	10-6
Coding Your COBOL Programs	10-6
Linking Your COBOL Programs	10-7
Executing Your COBOL Programs	10-7

Appendix A: Compatibility Levels

ODBC API Conformance Levels	A-1
JDBC API Conformance Levels	A-1
ODBC SQL Conformance Levels	A-1
ODBC Data Type Support	A-2
Advantage CA-Datcom/DB SQL Preprocessor Options	A-3
Advantage CA-Datcom/DB Navigational Command Support	A-3

Appendix B: Workstation Templates

InstallShield Wizard	B-2
Advantage CA-Datcom/DB ODBC Setup	B-3
Connection Window Template	B-4
CAICCI Window Template - Test	B-5
CAICCI Window Template - TCP/IP	B-6
CAICCI Window Template - LU2	B-7

Appendix C: Messages

Workstation Messages	C-1
Messages with Positive Message Numbers	C-2
Messages with Negative Message Numbers	C-10
Mainframe Messages	C-17
Advantage CA-Datcom/DB Return Codes	C-36

Index

Introduction

Advantage™ CA-Datcom® Database Server Option (Advantage CA-Datcom Server) allows workstation-hosted applications to use the Advantage™ CA-Datcom/DB® (Advantage CA-Datcom/DB) Database management system (DBMS). Advantage CA-Datcom Server implements industry standard Application Programming Interfaces (APIs) and the Advantage CA-Datcom/DB proprietary interface. The Open Database Connectivity (ODBC) and Java Database Connectivity (JDBC) industry standard interfaces allow Advantage CA-Datcom/DB access without any special programming by using generic tool set applications. These interfaces can also be utilized by coding specialized programs to the API standards.

Advantage CA-Datcom Server is an ODBC 3.5 compliant driver. This driver implements the ODBC API and requires that your programs conform to the ODBC specification. This driver also requires that the Microsoft ODBC Driver Manager software is installed and operational. ODBC data sources must be defined to provide information about the Advantage CA-Datcom Server.

Advantage CA-Datcom Server JDBC Driver is a JDBC 1.2 compliant driver. The JDBC driver implements the JDBC API and requires that your Java programs conform to the JDBC specification. This driver requires a Java Virtual Machine (JVM) and a JDBC Driver Manager. The Advantage CA-Datcom Server can operate as a “Type 1,” “Type 2,” or “Type 4” JDBC driver.

Advantage CA-Datcom Server Workstation Driver provides the Advantage CA-Datcom/DB proprietary interface. This interface implements the navigational record-at-a-time and set-at-a-time (RAAT/SAAT) programming specifications and requires programs to be coded in a language that can make calls to a Dynamic Link Library (DLL).

Advantage CA-Datcom Server Workstation Driver also provides the Advantage™ CA-Realia® II Workbench (Advantage CA-Realia II) COBOL interface. This provides the ability to build COBOL programs that utilize a static mainframe SQL plan. The Advantage CA-Datcom Server Preprocessor and the Advantage CA-Realia II Workbench product can be used together to preprocess COBOL programs that build static mainframe SQL plans, to compile and link the COBOL programs for execution, and to execute the program to obtain results.

Advantage CA-Datcom Server Workstation Driver also provides the C interface. This interface provides the ability to build C programs that utilize static mainframe SQL PLAN. The Advantage CA-Datcom Server Preprocessor and the C compilers can be used together to preprocess C programs that build SQL plans, to compile and link C programs for execution, and to execute the program to obtain results.

Advantage CA-Datcom Server Java Server is a TCP/IP interface to the Advantage CA-Datcom Server JDBC Driver. The Java Server allows Advantage CA-Datcom JDBC applets to be included in web pages and executed by web browsers.

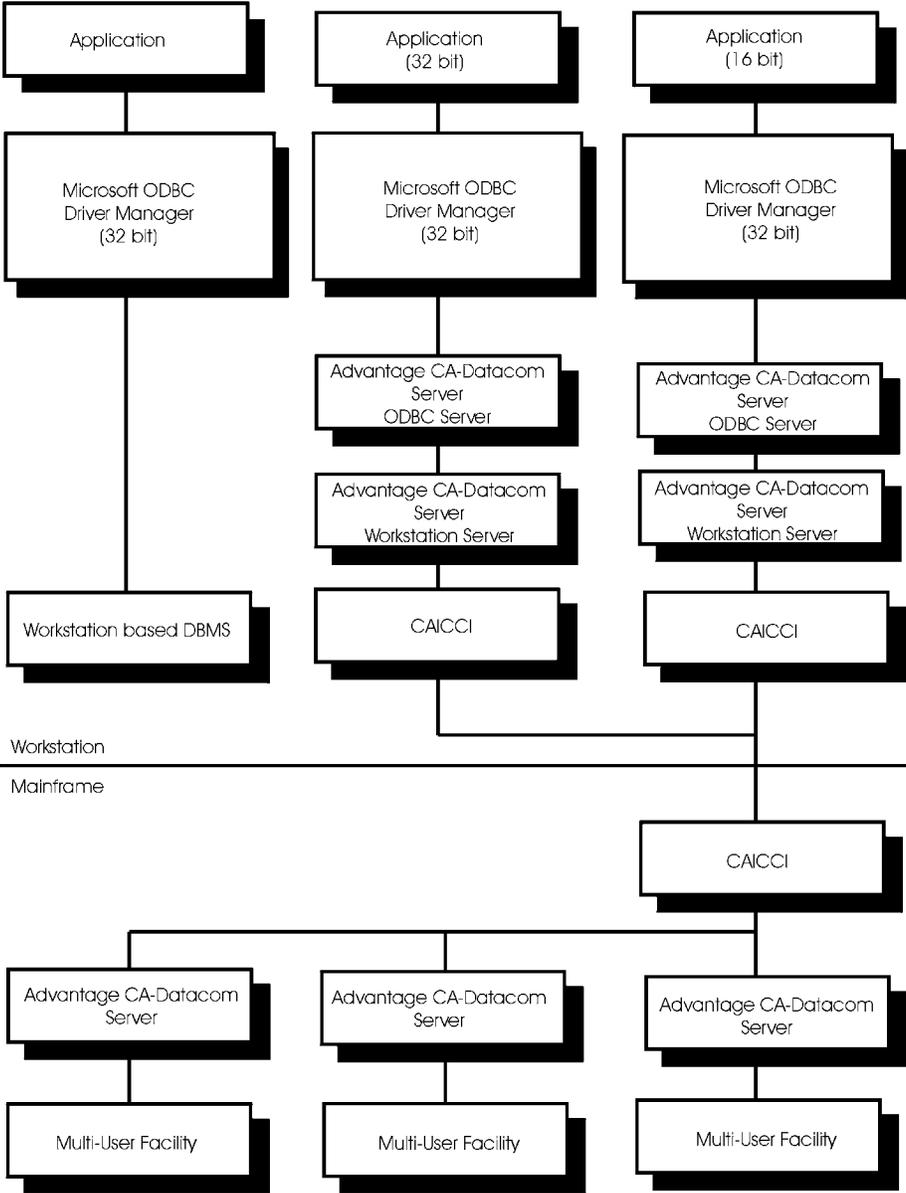
Audience

This guide assumes that you are already familiar with the concepts and features of Advantage CA-Datcom/DB. If you are unfamiliar with Advantage CA-Datcom/DB, refer to its documentation for more information.

This guide also assumes that you are familiar with Windows terminology and navigational techniques. This includes how to work with standard Windows items like menus, dialogs, the clipboard, and the Control Panel.

ODBC Architecture

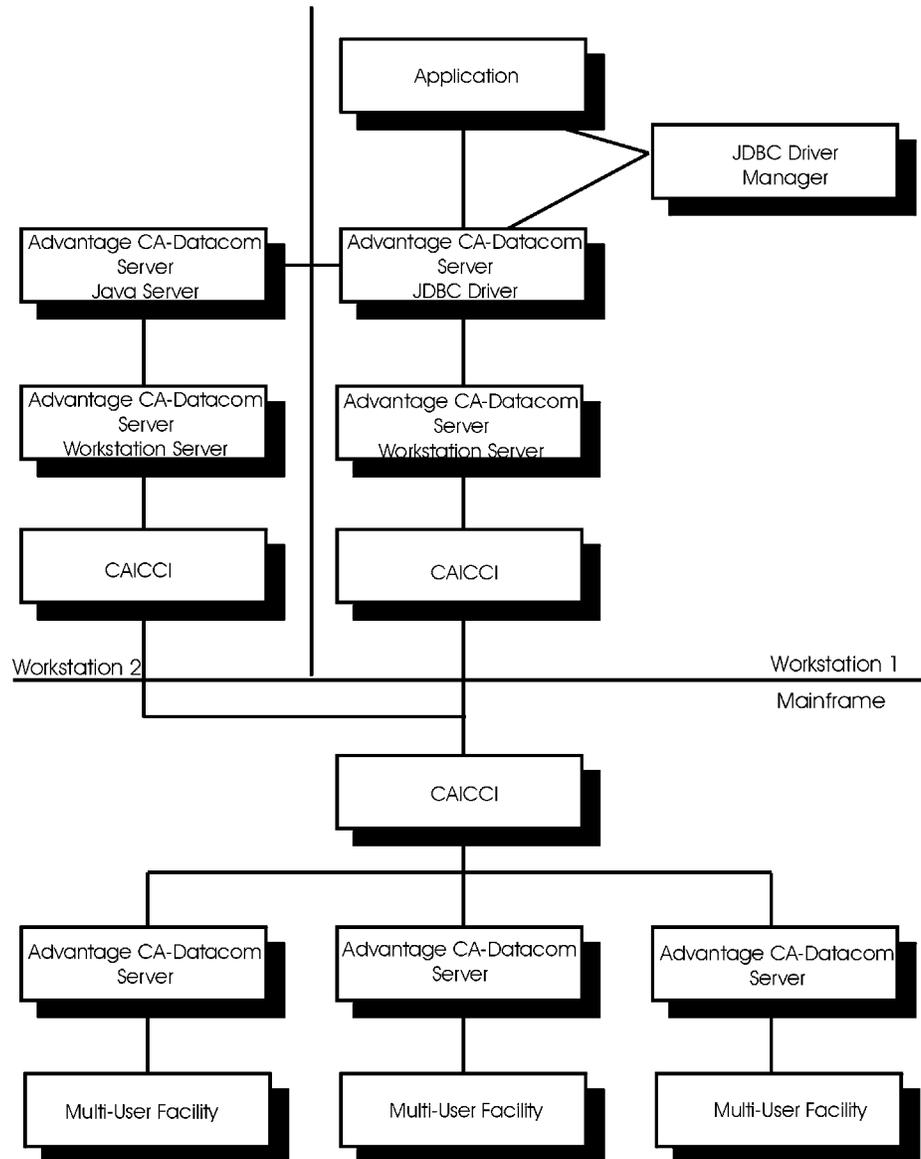
The following illustration shows the relationship of Advantage CA-Datcom Server to workstation applications that use ODBC to access Advantage CA-Datcom/DB mainframe data.



Application (Workstation)	<p>The components shown in this illustration are described as follows:</p> <p>SQL-based workstation applications using the Microsoft ODBC interface running under Microsoft Windows 98 or higher can access data in a CA-Datcom/DB database. An application issues a “connect” request to the Microsoft ODBC Driver Manager for a specific driver, such as Advantage CA-Datcom Server.</p>
Microsoft ODBC Driver Manager (Workstation)	<p>The Microsoft ODBC Driver Manager resides on the workstation or LAN. Any number of workstation applications can connect to the ODBC Manager and the ODBC Manager can connect to any number of drivers. The ODBC Manager forwards an application's “connect” request to the appropriate driver, based on the identifiers in the “connect” call. Microsoft provides the Microsoft ODBC Driver Manager.</p>
Advantage CA-Datcom Server ODBC Driver (Workstation)	<p>ODBC Driver implementation for Advantage CA-Datcom/DB. Drivers, connected to ODBC, interpret calls from the Microsoft ODBC Driver Manager and make calls to the requested DBMS. The Advantage CA-Datcom Server driver is a 32-bit thread-safe ODBC 3.5 compliant driver.</p>
Advantage CA-Datcom Server Workstation Server (Workstation)	<p>The Workstation Server provides the proprietary Advantage CA-Datcom/DB interface in addition to providing data mapping, data translation, and data communication services. This server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datcom Server on the mainframe.</p>
CAICCI (Workstation and Mainframe)	<p>CAICCI isolates Advantage CA-Datcom Server from the communication protocol used between the workstation and mainframe. This means that Advantage CA-Datcom Server can use any, and only, protocols supported by CAICCI.</p>
Advantage CA-Datcom Server (Mainframe)	<p>Advantage CA-Datcom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a “connect” request for a specific server other than itself, it forwards the “connect” request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that Server can communicate directly.</p>
Multi-User Facility (Mainframe)	<p>A Multi-User Facility provides the ability to access any Advantage CA-Datcom/DB database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.</p>

JDBC Architecture

The following illustration shows the relationship of Advantage CA-Datacom Server to workstation applications that use JDBC to access Advantage CA-Datacom/DB mainframe data.



The components shown in this illustration are described as follows:

**Application
Browser
(Workstation)**

Java SQL-based workstation applications using the JDBC interface and running under a Java Virtual Machine supporting JDBC 1.2 can access data in an Advantage CA-Datcom/DB database through any web-enabled browser. An application issues a `DriverManager.getConnection` connection request to the JDBC Driver manager for a specific driver, such as Advantage CA-Datcom Server.

**Advantage CA-
Datcom Server
JDBC Driver**

The JDBC Driver implementation for Advantage CA-Datcom/DB. Drivers, once connected by the JDBC Driver Manager, interpret calls from the application and the JDBC Driver Manager and make calls to the requested DBMS. The Advantage CA-Datcom Server driver is a JDBC 1.2 compliant driver.

**Advantage CA-
Datcom Server**

The server provides the proprietary Advantage CA-Datcom/DB interface in addition to providing data mapping, data translation, and data communication services. This server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datcom Server on the mainframe.

**Advantage CA-
Datcom Server
Java Server**

The Advantage CA-Datcom Server Java Server is a Windows NT service that provides a TCP/IP interface to Java based applications. This server is invoked if specified by the connection URL and provides the ability for Advantage CA-Datcom Server to function as a "Type 4" JDBC driver.

**CAICCI
(Mainframe)**

CAICCI isolates Advantage CA-Datcom Server from the communication protocol used between the workstation and mainframe. This means that Advantage CA-Datcom Server can use any, and only, protocols supported by CAICCI.

**Advantage CA-
Datcom Server
(Mainframe)**

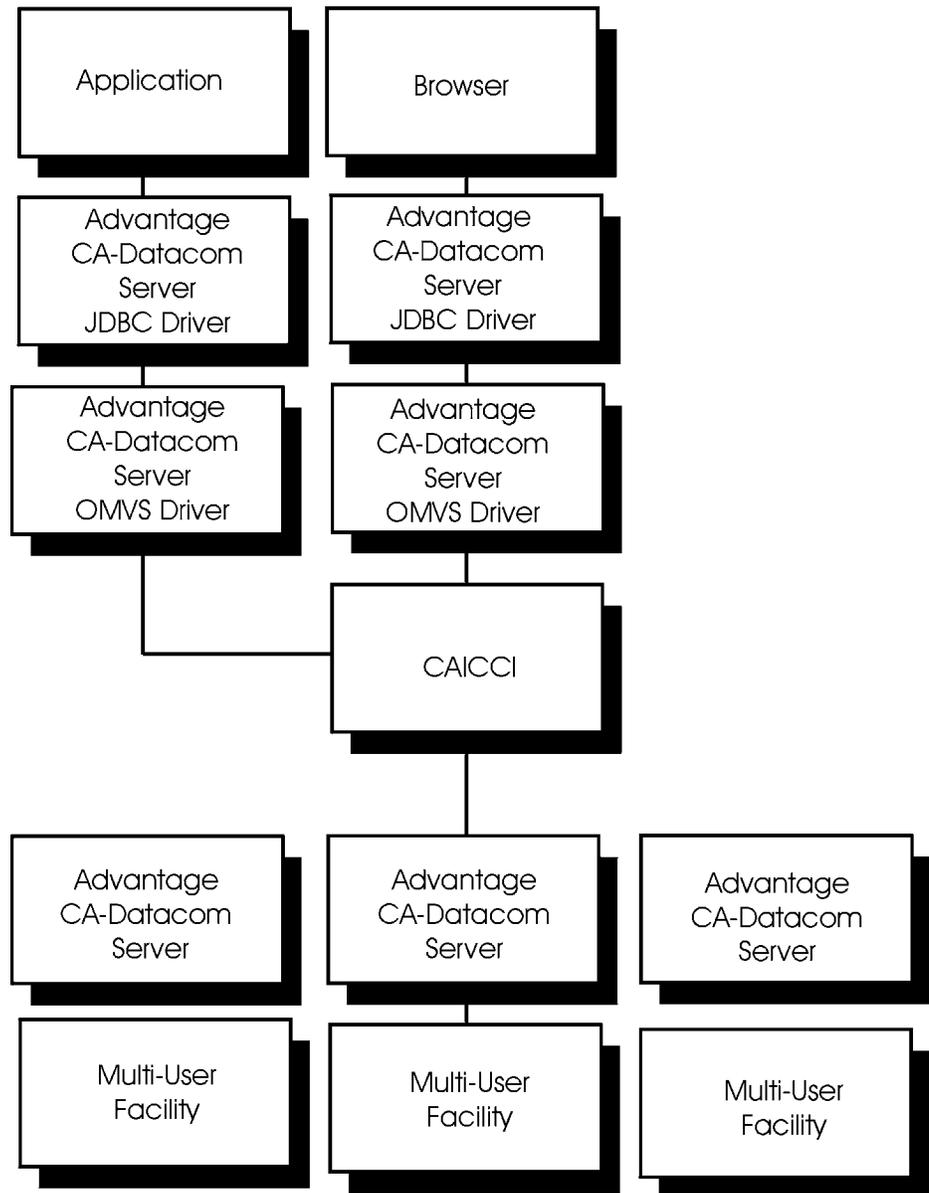
Advantage CA-Datcom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a connect request for a specific server other than itself, it forwards the connect request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that Server can communicate directly.

**Multi-User Facility
(Mainframe)**

A Multi-User Facility provides the ability to access any Advantage CA-Datcom/DB database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.

JDBC Architecture on z/OS Machines

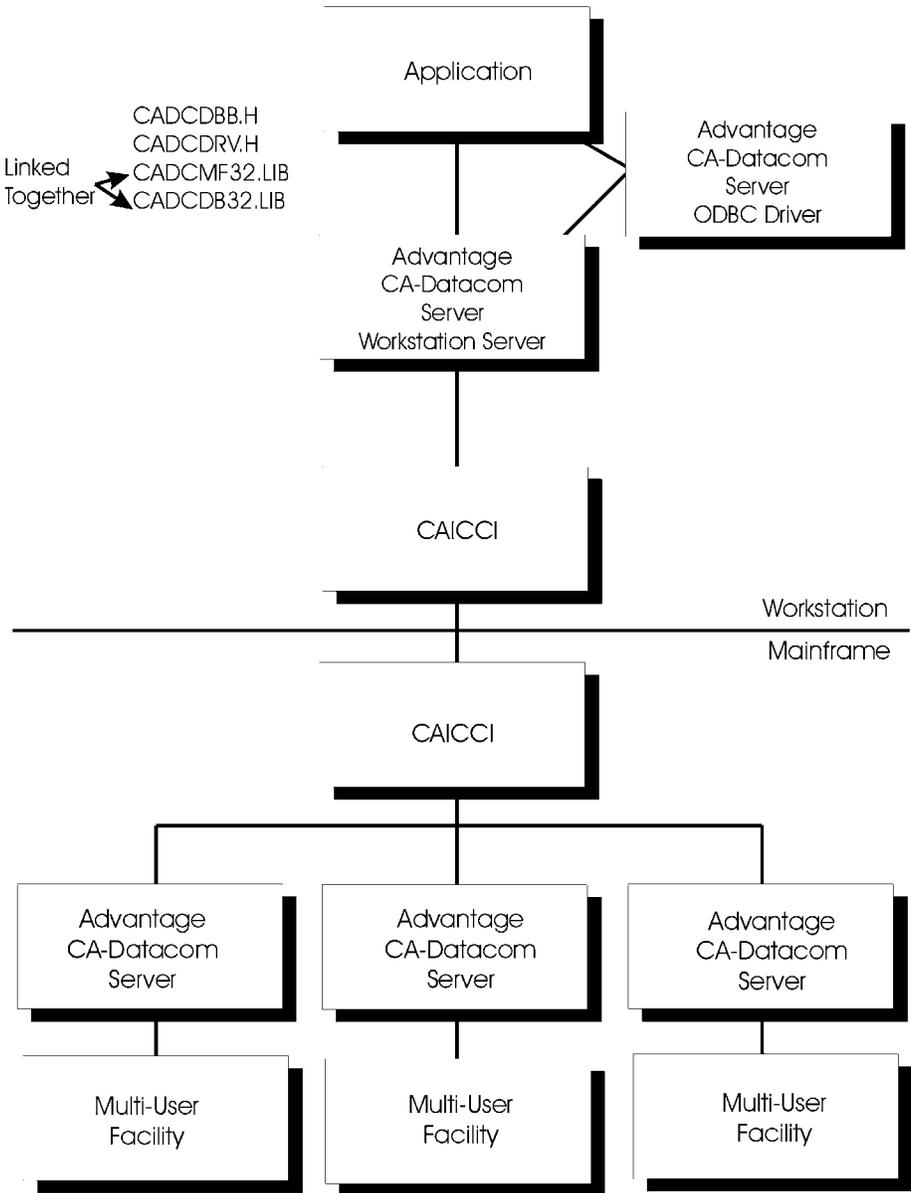
The following illustration shows the relationship of Advantage CA-Datacom Server to OMVS applications that use JDBC to access Advantage CA-Datacom/DB mainframe data on z/OS machines.



Application (OMVS)	<p>The components shown in this illustration are described as follows:</p> <p>Java SQL-based OMVS applications using the JDBC interface and running under a Java Virtual Machine supporting JDBC 1.2 can access data in an Advantage CA-Datcom/DB database. An application issues a <code>DriverManager.getConnection</code> connection request to the JDBC Driver manager for a specific driver, such as Advantage CA-Datcom Server.</p>
JDBC Driver Manager (OMVS)	<p>The JDBC Driver Manager is part of the OMVS Java virtual machine. The OMVS applications can connect to the JDBC Driver Manager and the JDBC Driver Manager can connect to any number of drivers. The JDBC Driver Manager forwards an application's connection request to the appropriate driver based on the Uniform Resource Locator (URL) specified in the connection request.</p>
Advantage CA-Datcom Server JDBC Driver (OMVS)	<p>The JDBC Driver implementation for Advantage CA-Datcom/DB. Drivers, once connected by the JDBC Driver Manager, interpret calls from the application and the JDBC Driver Manager and make calls to the requested DBMS. The Advantage CA-Datcom Server driver is a JDBC 1.2 compliant driver.</p>
Advantage CA-Datcom Server Workstation Server (OMVS)	<p>The OMVS Server provides the proprietary Advantage CA-Datcom/DB interface in addition to providing data mapping, data translation, and data communication services. This server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datcom Server on the mainframe.</p>
CAICCI (OMVS and Mainframe)	<p>CAICCI isolates Advantage CA-Datcom Server from the communication protocol used between OMVS and the Advantage CA-Datcom Server address space. This means that Advantage CA-Datcom Server can use any, and only, protocols supported by CAICCI.</p>
Advantage CA-Datcom Server (Mainframe)	<p>Advantage CA-Datcom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a connect request for a specific server other than itself, it forwards the connect request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that server can communicate directly.</p>
Multi-User Facility (Mainframe)	<p>A Multi-User Facility provides the ability to access any Advantage CA-Datcom/DB database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.</p>

Navigational Architecture

The following illustration shows the relationship of Advantage CA-Datcom Server to COBOL applications that use the Advantage CA-Datcom ODBC Driver for navigational access to Advantage CA-Datcom/DB mainframe data.

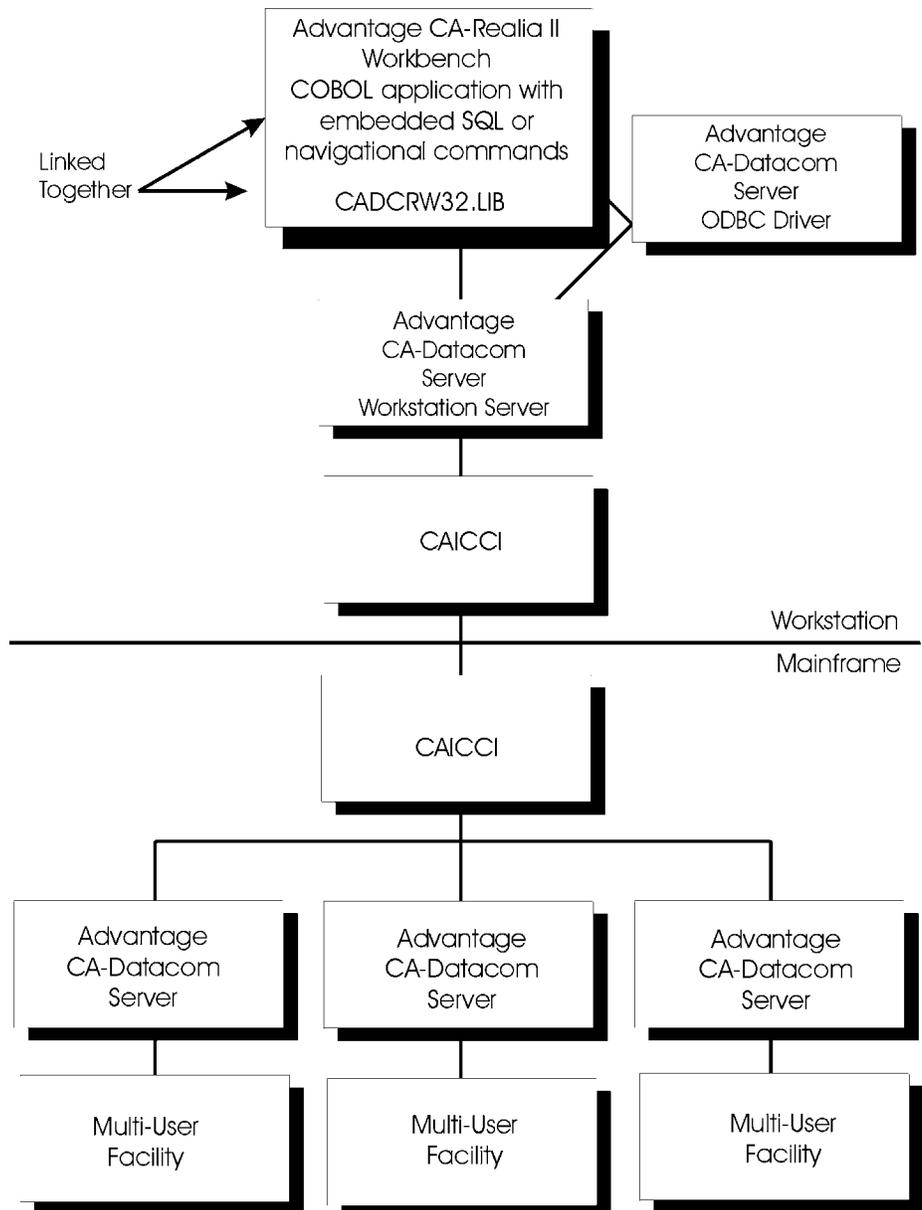


The components shown in this illustration are described as follows:

Application (Workstation)	Applications running on the workstation can use navigational Advantage CA-Datcom/DB commands to access mainframe Advantage CA-Datcom/DB data through Advantage CA-Datcom Server. The applications must be able to make DLL calls.
Advantage CA-Datcom Server ODBC Driver (Workstation)	ODBC Driver implementation for Advantage CA-Datcom/DB. Drivers, connected to ODBC, interpret calls from the Microsoft ODBC Driver Manager and make calls to the requested DBMS. The navigational programs access the Advantage CA-Datcom/DB ODBC Driver directly to perform connection and disconnection requests. The Advantage CA-Datcom Server driver is a 32-bit thread-safe ODBC 3.5 compliant Driver.
Advantage CA-Datcom Server Workstation Server (Workstation)	The Workstation Server provides the proprietary Advantage CA-Datcom/DB interface in addition to providing data mapping, data translation, and data communication services. This Server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datcom Server on the mainframe.
CAICCI (Workstation and Mainframe)	CAICCI isolates Advantage CA-Datcom Server from the communication protocol used between the workstation and mainframe. This means that Advantage CA-Datcom Server can use any, and only, protocols supported by CAICCI.
Advantage CA-Datcom Server (Mainframe)	Advantage CA-Datcom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a connect request for a specific server other than itself, it forwards the connect request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that server can communicate directly.
Multi-User Facility (Mainframe)	A Multi-User Facility provides the ability to access any database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.

Advantage CA-Realia II Workbench SQL Architecture

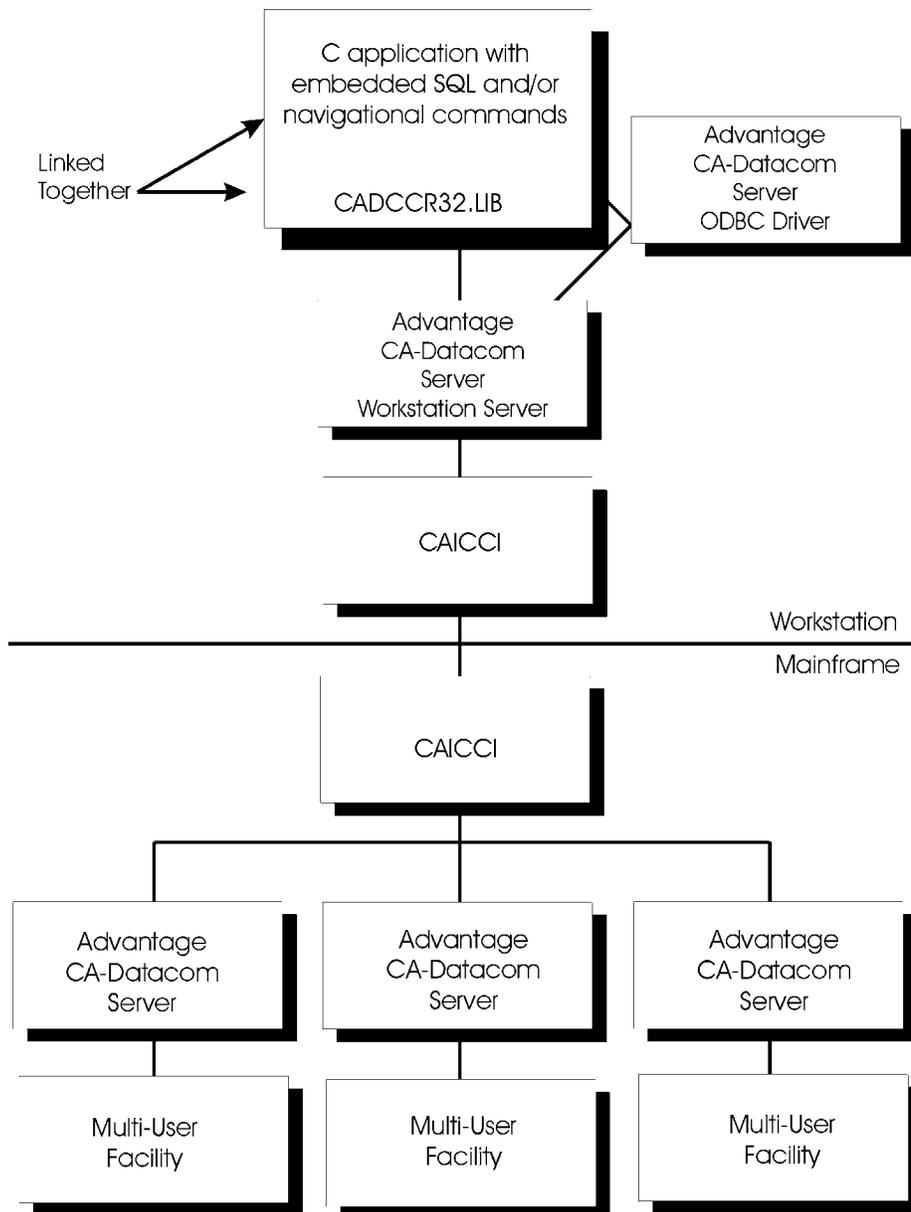
The following illustration shows the relationship of Advantage CA-Datacom Server to Advantage CA-Realia II Workbench COBOL applications that use the Advantage CA-Datacom Driver to access Advantage CA-Datacom/DB mainframe data.



Advantage CA-Realia II Workbench COBOL Applications (Workstation)	<p>The components shown in this illustration are described as follows:</p> <p>Advantage CA-Realia II Workbench lets you compile COBOL programs that access mainframe Advantage CA-Datacom/DB data through both SQL and navigational Advantage CA-Datacom/DB commands on the workstation. Such programs must be pre-compiled using the SQL Preprocessor installed with Advantage CA-Datacom Server.</p>
Advantage CA-Datacom Server ODBC Driver (Workstation)	<p>ODBC Driver implementation for Advantage CA-Datacom/DB. Drivers, connected to ODBC, interpret calls from the Microsoft ODBC Driver Manager and make calls to the requested DBMS. The navigational programs access the Advantage CA-Datacom Server ODBC Driver directly to perform connection and disconnection requests. The Advantage CA-Datacom Server driver is a 32-bit thread-safe ODBC 3.5 compliant driver.</p>
Advantage CA-Datacom Server Workstation Server (Workstation)	<p>The Workstation Server provides the proprietary Advantage CA-Datacom/DB interface in addition to providing data mapping, data translation, and data communication services. This server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datacom Server on the mainframe.</p>
CAICCI (Workstation and Mainframe)	<p>CAICCI isolates Advantage CA-Datacom Server from the communication protocol used between the workstation and mainframe. This means that Advantage CA-Datacom Server can use any, and only, protocols supported by CAICCI.</p>
Advantage CA-Datacom Server (Mainframe)	<p>Advantage CA-Datacom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a connect request for a specific server other than itself, it forwards the connect request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that server can communicate directly.</p>
Multi-User Facility (Mainframe)	<p>A Multi-User Facility provides the ability to access any database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.</p>

Preprocessor C Architecture

The following illustration shows the relationship of Advantage CA-Datcom Server to Advantage CA-Realia II Workbench COBOL applications that use the Advantage CA-Datcom Driver to access Advantage CA-Datcom/DB mainframe data.



The components shown in this illustration are described as follows:

**C Applications
(Workstation)**

Various C development environments lets you compile C programs that access mainframe Advantage CA-Datcom/DB data through both SQL and navigational Advantage CA-Datcom/DB commands on the workstation. Such programs must be precompiled using the SQL Preprocessor installed with Advantage CA-Datcom Server.

**Advantage CA-
Datcom Server
ODBC Driver
(Workstation)**

ODBC Driver implementation for Advantage CA-Datcom/DB. Drivers, connected to ODBC, interpret calls from the Microsoft ODBC Driver Manager and make calls to the requested DBMS. The navigational programs access the Advantage CA-Datcom Server ODBC Driver directly to perform connection and disconnection requests. The Advantage CA-Datcom Server driver is a 32-bit thread-safe ODBC 3.5 compliant driver.

**Advantage CA-
Datcom Server
Workstation
Server
(Workstation)**

The Workstation Server provides the proprietary Advantage CA-Datcom/DB interface in addition to providing data mapping, data translation, and data communication services. This server provides the CAICCI interface that allows the request to be sent through CAICCI to the Advantage CA-Datcom Server on the mainframe.

**CAICCI
(Workstation and
Mainframe)**

CAICCI isolates Advantage CA-Datcom Server from the communication protocol used between the workstation and mainframe. This means that Advantage CA-Datcom Server can use any, and only, protocols supported by CAICCI.

**Advantage CA-
Datcom Server
(Mainframe)**

Advantage CA-Datcom Server is a task that executes in its own address space and connects to a particular Multi-User Facility. This task will also attempt to connect to other servers in its group. A server group is defined as having names that begin with the same first four characters. When a server receives a connect request for a specific server other than itself, it forwards the connect request to the appropriate server in its group. It also passes the address of the appropriate server back to the driver so that future calls for that server can communicate directly.

**Multi-User Facility
(Mainframe)**

A Multi-User Facility provides the ability to access any database concurrently from multiple regions. Each server communicates with only one Multi-User Facility, but a system can have multiple Multi-User Facilities accessed by multiple servers or multiple servers accessing the same Multi-User Facility.

Conventions

Key Names	The names of keys, such as Enter, Ctrl, and Delete appear in the document as they do on your keyboard, where possible.
Key Combinations	Whenever two keys are joined together with a plus (+) sign (for example, Ctrl+R), you should hold down the first key while pressing the second key to complete the command. Release the second key first.
User Input Examples	<p>The following conventions are used for user input:</p> <p>Literal information (text that the user must enter exactly as shown) is shown in bold, for example:</p> <p>In the Data Source edit control, enter new_york.</p>
Cross-References	<p>The following documentation conventions are used:</p> <ul style="list-style-type: none">■ Guide names are in italics, for example: See the <i>Advantage CA-Datacom Server Getting Started</i> guide.■ Chapter names are in double quotes, for example: See "Setting System-Wide Options."■ Section names are as they appear in the document, for example, Also see Opening a Saved Workspace.

Related Publications

The following documentation should be available if you plan to develop applications using Open Database Connectivity (ODBC). ODBC documentation should be available from Microsoft. These publications are not provided by CA.

Document Title	Description
<i>Programmer's Reference</i> Microsoft Open Database Connectivity Software Development Kit	Explains the ODBC interface, describes its features and explains how it is used by applications.
<i>SDK Guide</i> Microsoft Open Database Connectivity Software Development Kit	Acts as a guide to the Software Development Kit. Describes ODBC, the contents of the kit, hardware and software requirements, and installation of the kit.

The following documentation should be available if you plan to develop applications using Java Database Connectivity (JDBC). JDBC documentation should be available from Sun Microsystems. These publications are not provided by Computer Associates.

Document Title	Description
<i>Specification</i> JDBC 1.20 Specification	Specifies the JDBC API, which is a Java application-programming interface to SQL databases.
JDBC Technology Guide: Getting Started	Acts as a guide to JDBC.
<i>JDBC Documentation</i> JDBC API 1.20 Documentation	Explains the JDBC interface, describes its features and explains how it is used by applications.

Starting Advantage CA-Datacom Server

The Multi-User Facility requires a SYSIN statement in its startup job to identify each of the Advantage CA-Datacom products and options it is operating with. If a product or option is omitted from the list in the startup SYSIN, the Multi-User Facility does not permit the omitted product or option access to CA-Datacom/DB. Therefore, if you are installing Advantage CA-Datacom Server for the first time, add the following statement to the SYSIN in your Multi-User Facility startup job:

```
          DATACOM          DTCMSRVR
```

To demonstrate the successful installation of this product:

1. Create the Advantage CA-Datacom Server startup JCL (see the following pages).

Note: When it starts, Advantage CA-Datacom Server expects the Multi-User Facility (with which it communicates) and CAICCI to already be active. If they are not, the Advantage CA-Datacom Server startup job will not complete successfully.

2. Define a data source driver on the workstation. (See the ODBC Driver Configuration and Administration chapter for more information.)
3. Open a mainframe connection using any workstation-hosted application which supports the ODBC industry standard APIs, such as, CleverPath™ CA-Visual Express® Reporter (CleverPath CA-Visual Express), or Microsoft Access.

If the connection fails, verify that the workstation entries were entered correctly. If they were, reinstall the product. If further problems occur, follow the procedure on contacting CA Technical Support in the Troubleshooting section of the installation guide.

Note: The module SVDBSPR must be in an authorized library if you allow LOGON=YES for external security in the mainframe startup parameters. See the LOGON= parameter in the topic Input Parameters.

z/OS or OS/390 Startup JCL

An individual Advantage CA-Datcom Database Server must be brought up by submitting a JCL member like the following example to execute the startup program. See the description that follows for possible input parameters, which you can insert where indicated in the example. The order of the input parameters does not matter.

```
//DBSERVER EXEC PGM=SVDBSPR,REGION=6000K
//STEPLIB DD DSN=datacom.server.loadlib,DISP=SHR server loadlib
//          DD DSN=datacom.loadlib,DISP=SHR database loadlib
//          DD DSN=cautng.loadlib,DISP=SHR loadlib for Unicenter TNG
//SYSPRINT DD SYSOUT=* Framework for OS/390
//SYSUDUMP DD SYSOUT=*
//SNAPER DD SYSOUT=*
//SYSIN DD *
SERVERNAME=datacom_server_name
APPLID=datacom_server_CCI_applid
PLANNAME=sql_plan_prefix
DBUSERS=n
AUTHID=xxxxxx
 insert optional input parameters here
/*
```

If your site has multiple Advantage CA-Datcom Server servers, you must submit the startup JCL for each server as a separate job. Use the following control statements to identify the server, the libraries it needs, its Multi-User Facility, its server group (if any), the SQL plan options to use, and the number of Advantage CA-Datcom/DB threads to get.

Read further for a list of valid SQL plan options. For complete details about the SQL plan options, see the *Advantage CA-Datcom/DB SQL User Guide*.

Statements

The following describes the startup JCL job control statement requirements for this product:

JCL Statement	Description
EXEC	(Required.) Provides the name of the program to execute. SVDBSPR is the program name.
JOB	(Required.) Specifies a valid job card for the site. Parameters are site-specific.
SNAPER	(Required.) Advantage CA-Datcom/DB uses the SNAPER data set.

STEPLIB	<p>(Required.) The DSN statements represent the Advantage CA-Datcom Server, Advantage CA-Datcom/DB, and the CA Common Services for z/OS and OS/390. The Advantage CA-Datcom/DB load library includes the specification for the Multi-User Facility to which the server is to connect. If an operating group of servers is running at the same time, each can only connect with one Multi-User Facility, but each Multi-User Facility can be accessed by more than one server.</p> <p>The library containing the URTs to be loaded must be concatenated into the server STEPLIB. The server startup parameter LOADURT attempts to locate the URT from this DD.</p>
SYSPRINT	(Required.) Advantage CA-Datcom Server writes its messages to SYSPRINT.
SYSUDUMP	Recommended.

VSE Startup JCL

An individual Advantage CA-Datcom Server must be initiated by submitting a JCL member like the following example to execute the startup program. See the description that follows for possible input parameters, which you can insert where indicated in the example. The order of the input parameters does not matter.

```
* $$ JOB JNM=DBSERVER,...
* $$ LST CLASS=A
// JOB DBSERVER
// DLBL DBLIB,'ca.database.library',99/365,SD
// EXTENT      ,volser
// DLBL CACIS,'cacis.library',99/365,SD
// EXTENT      ,volser
// DLBL DBSLIB,'ca.dbserver.library',99/365,SD
// EXTENT      ,volser
// LIBDEF PHASE,SEARCH=(DBLIB.sublib,cacis.sublib,DBSLIB.sublib)
// EXEC SVDBSPR,SIZE=786K
SERVERNAME=datacom_server_name
APPLID=datacom_server_CCI_applid
PLANNAME=sql_plan_prefix
DBUSERS=n
AUTHID=xxxxxxx
LOGON=NO
ISOLEVEL=C
DATE=ISO
TIME=JIS
  insert optional input parameters here
/*
/&
* $$ EOJ
```

If your site has multiple Advantage CA-Datcom Servers, you must submit the startup JCL for each server as a separate job. Use the following control statements to identify the server, the libraries it needs, its Multi-User Facility, its server group (if any), the SQL plan options to use, and the number of Advantage CA-Datcom/DB threads to get.

Read further for a list of valid SQL plan options. For complete details about the SQL plan options, see the *Advantage CA-Datcom/DB SQL User Guide*.

Statements

The following describes the startup JCL job control statement requirements for this product:

JCL Statement	Description
EXEC	<i>(Required.)</i> Names the program to execute. SVDBSPR is the program name.
JOB	<i>(Required.)</i> Specifies a valid job card for the site. Parameters are site-specific.
LIBDEF PHASE	<i>(Required.)</i> The DSN statements represent the Advantage CA-Datcom Server, Advantage CA-Datcom/DB, and the CA-CIS load libraries. The Advantage CA-Datcom/DB load library includes the specification for the Multi-User Facility to which the server is to connect. If an operating group of servers is running at the same time, each can only connect with one Multi-User Facility, but each Multi-User Facility can be accessed by more than one server.

Note: In VSE, Advantage CA-Datcom Server writes its messages to SYSLST.

Input Parameters

The following describes the Advantage CA-Datcom Server startup JCL input parameters. Code all input parameters in the form:

option-name=option-value

Parameter	Description
ACCESS=	<i>(Optional.)</i> Specifies whether the server has read-only or read/write access and only applies to ODBC access. Navigational access is not affected by this parameter. Enter R for read-only access or W for read/write access. Valid Entries: R or W Default Value: W

APPLID=	<p>(Required.) Identifies the Unicenter TNG Framework for OS/390 or CA-CIS CCI application ID. When an operation group of servers exists, the first four characters of this option must be the same for every member of the group. Enter 1 to 20 alphabetic characters, following the rules for SQL names. Each Server must have a unique APPLID.</p> <p>Valid Entries: 1 to 20 alphabetic characters, following the rules for SQL names</p> <p>Default Value: No default</p>
AUTHID=	<p>(Optional.) Enter a valid SQL AUTHID to be used by all clients of this server. This AUTHID must already exist on the attached Multi-User Facility. For details on creating the AUTHID, see the <i>Advantage CA-Datcom/DB SQL User Guide</i>.</p> <p>Valid Entries: Any valid SQL AUTHID</p> <p>Default Value: SYSUSR</p>
CONEXIT=	<p>(Optional.) Specifies a user coded connection exit to replace the security provided by Advantage CA-Datcom Server. A sample connection exit, SVCXTPR, is provided.</p> <p>Valid Entries: A valid connection exit program</p> <p>Default Value: No default (spaces)</p>
DBUSERS=	<p>(Required.) The number of Advantage CA-Datcom/DB threads to be dedicated to this server. The server uses this number to reserve threads to service requests from the server. Since any number of non-update requests can share threads, the number to request is determined mainly by the number of concurrent update requests. When a client workstation issues any request causing an update, the thread being used is held for this task until the task issues a COMMIT or ROLBK or requests a disconnect. In estimating the number of DBUSERS, consider the number of update requests operating concurrently a large percentage of the time and allocate a thread for each of these. In addition, allocate one thread for each five concurrent read-only type requests. See the topic Monitoring Advantage CA-Datcom Server for details on determining how to monitor this number.</p> <p>Valid Entries: A number from 1 to 999</p> <p>Default Value: No default</p>

DDBID= (Optional.) Enter the BASEID of the Datadictionary database (DATA-DICT) for the Datadictionary on the Multi-User Facility that is accessed through this server.

Valid Entries: A valid Advantage CA-Datacom database ID.

Default Value: 2

DDDID= (Optional.) Enter the BASEID of the Data Definition Directory database (DDD-DATABASE) for the Datadictionary on the Multi-User Facility that is accessed through this server.

Valid Entries: A valid Advantage CA-Datacom database ID

Default Value: 15

DYNURT= (Optional.) Specifies whether Advantage CA-Datacom Server is to dynamically build User Requirements Tables (URTs) for non-SQL access to Advantage CA-Datacom tables. If you specify DYNURT=NO (or allow this parameter to default), non-SQL access is restricted to those tables named in the URTs loaded with the LOADURT= parameter. If you specify DYNURT=YES, URTPRTY= will default to 7 unless you specify a value for the URTPRTY= parameter.

For details on how Advantage CA-Datacom Server handles URTs, see the section on managing URTs in the chapter "Monitoring Advantage CA-Datacom Server".

Valid Entries: NO or YES

Default Value: NO

LOADURT=	<p>(Optional.) Specifies the name(s) of one or more User Requirements Tables (URTs) for CA-Datacom Server to load. If you specify DYNURT=NO (or allow that parameter to default), non-SQL access is restricted to those tables named in the URTs loaded with the LOADURT= parameter. Although CA-Datacom Server loads these URTs at startup, it opens them only as they are required to fulfill a request from a workstation.</p> <p>To specify more URTs than will fit on a single input line, specify multiple LOADURT= parameters.</p> <p>For details on how CA-Datacom Server handles URTs, see the section on managing URTs in the chapter “Monitoring Advantage CA-Datacom Server.”</p> <p>Valid Entries: A URT name or a list of URT names</p> <p>Note: When specifying as a list, the list must be comma-separated and enclosed in single quotes with no embedded blanks.</p> <p>Default Value: No default</p>
LOGON=	<p>(Optional.) Specifies whether the server will attempt to do a logon for each user to external security. If LOGON=YES, each workstation user will have to be authorized on external security to log on to the Server.</p> <p>Note: If LOGON=YES, the module SVDBSPR must be in an authorized library.</p> <p>Valid Entries: NO or YES</p> <p>Default Value: YES</p>
MAXURTS=	<p>(Optional.) Specifies the maximum number of URTs that can be loaded by Advantage CA-Datacom Server.</p> <p>Valid Entries: A number from 1 to 999</p> <p>Default Value: 100</p>
PLANNAME=	<p>(Required.) Specifies a four-character prefix to be used as the first four characters of all plan names created by the server. These must be four alphanumeric or @, \$, or # characters. Each server attached to the same Multi-User Facility should provide a unique PLANNAME prefix. This ensures that the servers do not generate conflicting PLANNAMEs on the Multi-User Facility.</p> <p>Valid Entries: 4 alphanumeric characters, @, \$, or #.</p> <p>Default Value: No default</p>

SECEXIT= (Optional.) Specifies a user-coded security exit to enhance the security provided by Advantage CA-Datacom Server. A sample security exit, SVCXTPR, is provided.

Valid Entries: A valid security exit program

Default Value: No default (spaces)

SERVERNAME= (Required.) Identifies the name of the individual server being started. Each server must have a unique name. Enter 1 to 32 alphabetic characters, following the rules for SQL names. See the Advantage CA-Datacom/DB SQL User Guide.

Valid Entries: 1 to 32 alphabetic characters, following the rules for SQL names

Default Value: No default

SQLMEM= (Optional.) Specifies the SQL memory usage. SQL statement storage is kept in memory for reuse. The KEEP parameter will allow this memory to be kept. The FREE parameter will cause this memory to be freed from memory and will not allow reuse. The Java code for ParameterizedQueries needs this value to be specified as KEEP. Specifying FREE can reduce the resources consumed in SQL.

Valid Entries: KEEP or FREE

Default Value: KEEP

SQL Plan Options (Optional.) Input parameters for SQL preprocessor options that you can set at startup. In most cases, you do not need to change these parameters. Therefore, we recommend that you accept the defaults.

DATE=, TIME=, and DECPOINT= are probably the only parameters you may want to change. (The default values for these parameters are appropriate for most PC applications.) Their options are listed following. (For complete details on the purpose of all plan options other than DATE=, TIME=, and DECPOINT=, see the Advantage CA-Datacom/DB SQL User Guide.)

Function	SQL Plan Option Parameters with Defaults
SQL mode selection	SQLMODE= DATACOM

Performance or Resource adjustment	CBSIO= 0 PRTY= 7 WORKSPACE= 16 TIMEMIN= 0	PLNCLOSE= T ISOLEVEL= C OPT= P TIMESEC= 0
Debug	MSG= NN	
Delimiter character strings	STRDELIM= A	
Display format change for date, time, or decimal point.	DATE=ISO <ul style="list-style-type: none"> ■ EUR dd.mm.yyyy (IBM European Standard)(default) ■ ISO yyyy-mm-dd (International Standards Organization) ■ JIS yyyy-mm-dd (Japanese Industrial Standard) ■ USA mm/dd/yyyy (USA Standard) TIME=JIS <ul style="list-style-type: none"> ■ EUR: hh.mm.ss ■ ISO hh.mm.ss ■ JIS hh:mm:ss (default) ■ USA hh:mm AM or PM DECPOINT=P <ul style="list-style-type: none"> ■ P Specifies that a period be used as a decimal point. (default) ■ C Specifies that a comma be used as a decimal point 	
SUPMSG=	(Optional.) Specifies if the connect and disconnect messages are written out to the server output logs.	
	Valid Entries: NO or YES	
	Default Value: NO	

TIMEOUT=	<p>(Optional.) Time in minutes a connection from a client may remain idle before an automatic rollback and disconnect occur. TIMEOUT=1440 (24 hours) specifies that no connection is ever automatically disconnected.</p> <p>Valid Entries: A number from 1 to 1440</p> <p>Default Value: 120</p>
TIMEOUTWAIT=	<p>(Optional.) Time in minutes that server will wake up and time out tasks using the TIMEOUT value. This time out-task will wake up every interval and time-out tasks that have had no activity for the interval specified in the TIMEOUT parameter. This parameter will allow for server to timeout task when there is no activity against server. A value of 0 will cause no timeouts to occur.</p> <p>Valid Entries: A number from 0 to 720</p> <p>Default Value: 15 minutes</p>
TRACEON=	<p>(Optional.) Turn on the Trace Facility.</p> <p>Valid Entries: NO or YES</p> <p>Default Value: NO</p>
TRUSER1= TRUSER2= TRUSER3= TRUSER4= TRUSER5=	<p>(Optional.) The user ID of the session a trace is desired. This will cause the trace specified in the 'TRACE' options to be in effect for this user. Setting this value to spaces will remove the user ID from the trace list.</p> <p>Valid Entries: A valid user ID</p> <p>Default Value: None or Spaces</p>
URTPRTY=	<p>(Required if DYNURT=YES.) Specifies the priority that Advantage CA-Datacom Server is to use for dynamically built User Requirements Tables (URTs).</p> <p>For details on how Advantage CA-Datacom Server handles URTs, see the section on managing URTs in the chapter "Monitoring Advantage CA-Datacom Server."</p> <p>Valid Entries: A number from 1 to 15</p> <p>Default Value: 7</p>

Monitoring Advantage CA-Datcom Server

Advantage CA-Datcom Server provides a communications utility (SVCOMPR) that you can use to issue instructions to an Advantage CA-Datcom Server. See the chapter “Using the Communication Utility” for information about this utility.

Administrator Responsibilities

As the Advantage CA-Datcom Server administrator, you are responsible for the following tasks:

- Restarting a mainframe Advantage CA-Datcom Server when it has been terminated after a Multi-User Facility failure
- Establishing and maintaining Advantage CA-Datcom Server groups
- Diagnosing problems
- Monitoring workstation connections
- Monitoring Advantage CA-Datcom Server User Requirements Tables

Restarting a Server

Multi-User Facility Failure

During Advantage CA-Datcom Server operation, a Multi-User Facility failure may occur, causing the server to automatically terminate. Any request from the workstation receives a communications error after the server is terminated. When the server is terminated by the communication utility, all currently active connections are rolled back and the connections are terminated.

If there are no current database requests at the time of a Multi-User Facility termination, the server may detect a Multi-User Facility abend. In that event, the workstation receives a database error at the next Advantage CA-Datcom/DB request. When Multi-User Facility is restarted after an abend, Advantage CA-Datcom/DB rolls back all requests that have not been committed.

You can successfully restart the server only after the Multi-User Facility has been restarted. Any time the Multi-User Facility is terminated or abends and is restarted, you must terminate the server (if it does not terminate automatically) and restart it.

Using Server Groups

To facilitate access to multiple servers, servers can be assigned to a group. To designate a group of servers, simply assign all servers in the group APPLIDs that have the same first four characters as the APPLID for each server in the group.

At connect time, when server groups are used, the workstation application may present the workstation user with a list of servers currently in the group and the user can select a server to connect to from that list. This allows the mainframe administrator to add or delete servers or change their names without requiring any administrative changes at the workstation.

Each workstation is initially connected to the server group of the server identified in the data source that the workstation user chooses. If a server receives a request that requires a Multi-User Facility other than the one to which it is connected, the server forwards the request to the appropriate server for that Multi-User Facility. It also returns the address of the correct server to the Advantage CA-Datcom driver originating the request so that it can communicate future requests directly with the correct server on the mainframe.

Diagnosing Problems

The Advantage CA-Datcom Server has a built-in diagnostic capability. The server communications utility (SVCOMPR) lets you turn on a trace facility and produce a dump. (See the chapter "Using the Communication Utility" for details on controlling this trace facility.) This dump is usable only by CA Technical Support. It is printed on SYSPRINT.

Monitoring Workstation Connections

The Advantage CA-Datcom Server is expected to remain active for long periods of time. The server communications utility (SVCOMPR) STATUS and CANCEL statements discussed in the chapter “Using the Communication Utility” provide a way to submit batch jobs to determine who is connected to the server and to cancel any connection.

Periodically, issue the STATUS command through the communication utility (SVCOMPR) to see if any client workstation requests are waiting for a free Advantage CA-Datcom/DB thread. If so, you probably need to increase the DBUSERS= number. You may also need to increase the value of the TASKS Multi-User Facility startup option.

Monitoring User Requirements Tables

All non-SQL access to Advantage CA-Datcom/DB data requires a User Requirements Table. The communications utility (SVCOMPR) provides an option to report the status of all server URTs (see the chapter, “Using the Communication Utility”). In addition to reporting the status, this option reports the URT number, that you must use to close or open a URT.

When you need to perform maintenance on a database, you can prevent non-SQL access to that database by using the communications utility (SVCOMPR) to close any URTs containing tables in that database. The server will not build a dynamic URT to access a table, which is in a closed URT.

If the server receives a non-SQL Advantage CA-Datcom request for a table in a closed URT, it returns an Advantage CA-Datcom/DB return code 05 (34) to the application.

Once you have closed a URT, you can use the communication utility (SVCOMPR) to make that URT available again.

Using the Communication Utility

A server should terminate automatically if its Multi-User Facility fails while there are active database requests. If the Multi-User Facility fails and the server does not terminate automatically, or if you need to restart a server with different input parameters, you must terminate the server yourself.

The following SVCOMPR batch job can be used to do the following:

- Terminate a server
- Turn on the trace facility
- Turn off the trace facility
- Display the status of User Requirements Tables
- Load User Requirements Tables for use by the server
- Close User Requirements Tables
- Open previously closed User Requirements Tables
- Get a list of connections from workstations to the server
- Cancel a connection that is waiting on a request from the workstation

This chapter contains complete descriptions of all SVCOMPR input parameters.

You must run the SVCOMPR on the same system as the server with which you want to communicate.

MVS or OS/390 SVCOMPR JCL

```
//JOBNAME JOB . . .,
//DBSRVC02 EXEC PGM=SVCOMPR,REGION=6000K
//STEPLIB DD DSN=datacom.Server.loadlib,DISP=SHR Server loadlib
//          DD DSN=datacom.loadlib,DISP=SHR database loadlib
//          DD DSN=cautng.loadlib,DISP=SHR loadlib for Unicenter TNG
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SNAPER DD SYSOUT=*
//SYSIN DD *
APPLID=datacom_server_cci_applid
insert optional input parameters here
//
```

Statements

The following describes the JCL control statement SVCOMPR requirements for Advantage CA-Datacom Server.

JCL Statement	Description
EXEC	<i>(Required.)</i> Names the program to execute. SVCOMPR is the program name for the server communications utility.
JOB	<i>(Required.)</i> This statement specifies a valid job card for the site. Parameters are site-specific.
SNAPER	<i>(Required.)</i> Advantage CA-Datacom/DB uses the SNAPER data set.
STEPLIB	<i>(Required.)</i> The DSN statements represent the Advantage CA-Datacom Server, Advantage CA-Datacom/DB, and the CA Common Services for z/OS and OS/390 (formerly CA90s Services) load libraries. The Advantage CA-Datacom/DB load library includes the specification for the Multi-User Facility to which the Server is to connect. If an operating group of servers is running at the same time, each can only connect with one Multi-User Facility, but each Multi-User Facility can be accessed by more than one server (see the explanation of server groups in the chapter “Monitoring Advantage CA-Datacom Server”). The library containing the URTs to be loaded must be concatenated into the server STEPLIB. The server startup parameter LOADURT attempts to locate the URT from this DD.
SYSPRINT	<i>(Required.)</i> Advantage CA-Datacom Server writes its messages to SYSPRINT.
SYSUDUMP	Recommended.

VSE SVCOMPR JCL

```

* $$ JOB JNM=DBSVCOMM,...
* $$ LST CLASS=A
// JOB DBSVCOMM
// DLBL DBLIB,'ca.database.library',99/365,SD
// EXTENT      ,volser
// DLBL CACIS,'cacis.library',99/365,SD
// EXTENT      ,volser
// DLBL DBSLIB,'ca.dbserver.library',99/365,SD
// EXTENT      ,volser
// LIBDEF PHASE,SEARCH=(DBLIB.sublib,CA-CIS.sublib,DBSLIB.sublib)
// EXEC SVCOMPR,SIZE=768K
APPLID=datacom_server_cci_applid
insert optional input parameters here
/*
/&
* $$ EOJ

```

Statements

The following describes the JCL job control statement SVCOMPR requirements for Advantage CA-Datacom Server.

JCL Statement	Description
EXEC	<i>(Required.)</i> Names the program to execute. SVCOMPR is the program name for the server communications utility.
JOB	<i>(Required.)</i> Specifies a valid job card for the site. Parameters are site-specific.

LIBDEF PHASE *(Required.)* The DSN statements represent the Advantage CA-Datcom Server, Advantage CA-Datcom/DB, and the CA-CIS (formerly CA90s Services) load libraries. The Advantage CA-Datcom/DB load library includes the specification for the Multi-User Facility to which the server is to connect. If an operating group of servers is running at the same time, each can only connect with one Multi-User Facility, but each Multi-User Facility can be accessed by more than one server (see the explanation of Server groups in the chapter “Monitoring Advantage CA-Datcom Server”).

Note: In VSE, Advantage CA-Datcom Server writes its messages to SYSLST.

SVCOMPR Input Parameters

The following describes the SVCOMPR input parameters for this product. These options are commands you can send to the server. Entries must begin in column 1.

APPLID= *(Required.)* Identifies the CA Common Services for z/OS and OS/390 or CA-CIS application ID. This is the APPLID of the specific Server to receive the communication.

Column	Entry
1-7	APPLID=
8-15	1 to 20 characters to identify the CA Common Services for z/OS and OS/390 or CA-CIS application ID to receive the communication Valid Entries: 1 to 20 alphabetic characters, following the rules for SQL names Default Value: No default

CANCEL= (Optional.) To cancel a connection, first use the STATUS command to get a list of connections. Then create a statement as follows:

Column	Entry
1-7	CANCEL=
8-15	1 to 15-digit number specifying the number of the connection obtained from the STATUS list.

CLOSEURT= (Optional.) Specifies a User Requirements Table (or list of User Requirements Tables) to be closed. Once a User Requirements Table is closed, Advantage CA-Datcom Server will not reopen the URT until instructed to do so with the SVCOMPR OPENURT= option. Also, the server will not dynamically build a User Requirements Table for access to a table in a database involved in a closed User Requirements Table.

Column	Entry
1-9	CLOSEURT=
10-80	User Requirements Tables to be closed, separated by commas, with no embedded blanks. If you specify more than one User Requirement Table, enclose the list in single quotes. Specify the User Requirements Tables by number. Use the SVCOMPR URTSTATUS option to determine each User Requirements Table's number.

EOJ (Optional.) Tells the Server to terminate immediately.

Column	Entry
1-3	EOJ

LOADURT= (Optional.) Specifies a User Requirements Table name (or list of URT names) to be loaded. Once it has loaded a URT, the server does not open it until it needs it to satisfy a request from a workstation or it is instructed to do so with the SVCOMPR OPENURT= option.

Column	Entry
1-8	LOADURT=
9-80	The names of URTs to be loaded, separated by commas, with no embedded blanks. If you specify more than one URT, enclose the list in single quotes. A LOADURT statement may not be continued, but there may be multiple LOADURT statements in the input to the utility.

MODIFY= *(Optional.)* Allows modification of startup server options while the server is up. Any option modified by this statement will be in effect for all connections to the server after modification and will be in effect until another MODIFY is issued or until the server is recycled. All of the input options may be modified except the following:

APPLID
 CONEXIT
 DDBID
 DBUSERS
 DDDID
 LOADURT
 MAXURTS
 SERVERNAME
 TRACEON

Column	Entry
1-7	MODIFY=
8-80	Options to be modified, separated by commas, with no embedded blanks. A MODIFY statement may not be continued, but there may be multiple MODIFY statements in the input to the utility. See the chapter, "Starting Advantage CA-Datacom Server" for input options for the Server.

OPENURT= (Optional.) Specifies a loaded User Requirements Table (or list of URTs) to be opened. Once a URT has been closed with a SVCOMPR CLOSEURT= option, the server will not reopen it until instructed to do so with the OPENURT= option. Also, the server will not dynamically build a URT for access to a table in a database involved in a closed URT.

Note: When the server receives an OPENURT= request, it flags the named URT as available, but it does not actually open it until it needs that it to satisfy a request from a workstation.

Column	Entry
1-8	OPENURT=
9-80	URT's to be opened, separated by commas, with no embedded blanks. If you specify more than one URT, enclose the list in single quotes. Specify the URTs by number. Use the SVCOMPR URTSTATUS option to determine each URT's number.

OPTIONS (Optional.) Requests the communications utility to report a list of all the server options and their current settings.

Column	Entry
1-7	OPTIONS

STATUS (Optional.) Produces a status report of activity currently being processed by Advantage CA-Datacom Server.

Column	Entry
1-6	STATUS

TRACEON (Optional.) Turn on Trace facility.

Column	Entry
1-7	TRACEON

TRACEOFF (Optional.) Turn off Trace facility.

Column	Entry
1-8	TRACEOFF

URTSTATUS (Optional.) Get a list of User Requirements Tables loaded by the server (including dynamically built URTs), their identification numbers (for use with the OPENURT= and CLOSEURT= options), and status. Only the native access URTs are displayed.

Column	Entry
1-9	URTSTATUS

Sample Output from STATUS Command

In the report, the displayed Status values can be as follows:

```

CA-Datacom Server 5.0 0000 COMMUNICATIONS UTILITY MESSAGES.
Current Date: 2/12/2003                Current time: 08:55:30

DSV00060I-Communications utility input command received:
APPLID=LV2_TRAN_SV50
DSV00060I-Communications utility input command received:
STATUS
DSV00062I-Communications utility CCI System ID:A44SENF
DSV00065I-Communications utility sending request to APPLID:LV2_TRAN_SV50

DSV00067I-FOLLOWING REPORT RETURNED FROM SERVER 5.0 0000:
SERVER GROUP:
SERVERNAME                SYSTEMID APPLID
-----
LV2_TRAN_SV50             A44SENF LV2_TRAN_SV50
LV2_LCE_SV40_MUFY        A44SENF LV2_LCE_SV40_MUFY
CONNECTIONS TO SERVER: LV2_TRAN_SV50
TASK NO USERID  SYSTEMID APPLID                STATUS
-----
0000008 SVDBSPR A44SENF DBSRVCOM                PROCESSING REQUEST
0000007 DOEJO01F WDBSQLE135459 515** WAITING REQUEST-NO THREAD HELD
0000006 DOEJA01C WDBSQLE135446 529** WAITING REQUEST-HOLDING THREAD
0000004 SMIJO01A WDBSQLE135326 75**  WAITING REQUEST-HOLDING THREAD
DSV00068I-END OF REPORT
    
```

ODBC Driver Configuration and Administration

Workstation users must configure at least one data source for the Advantage CA-Datcom Server driver so that the driver can be chosen to connect to the mainframe. The Advantage CA-Datcom Server ODBC Driver must know the CAICCI system and application IDs of a mainframe server. Whether or not the workstation user fills in the CAICCI IDs and other data source options when configuring a data source depends on how the user's ODBC-enabled applications handle ODBC connections.

Administrator Tasks

Given the particular workstation ODBC-enabled applications executed, the database administrator should decide on naming standards and supply data source configuration instructions and connection parameters to workstation users. See Workstation Templates for templates to be used when supplying workstation users with entries for the Advantage CA-Datcom Server screens.

ODBC Driver

The Advantage CA-Datcom/DB ODBC Driver is dependent upon the Microsoft ODBC Driver Manager and ODBC Administrator. This software provided by Microsoft establishes the interface between the ODBC Driver and application and the administration utilities needed to configure ODBC Drivers. The Advantage CA-Datcom Server ODBC Driver also provides a configuration dialog. This dialog is used to configure the specific Advantage CA-Datcom access parameters. The following are requirements and recommendations for you to set up Advantage CA-Datcom/DB ODBC access.

ODBC Driver Manager

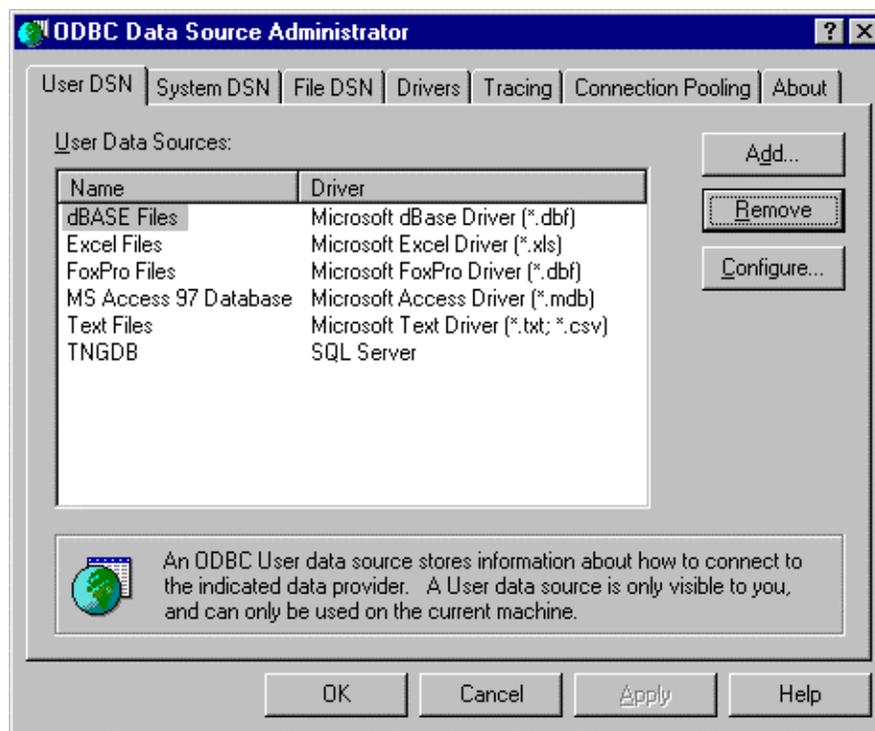
The Driver Manager is software responsible for managing communication between applications and ODBC drivers. This software resolves common problems: determining which ODBC driver to load based upon common information contained in a data source, loading and unloading drivers, and evoking ODBC driver functions. The Driver Manager is made up of DS16GT, DS32GT.DLL, ODBC16GT.DLL, ODBC32.DLL, ODBC32GT.DLL, ODBCCP32.CPL, ODBCCP32.DLL, ODBCCR32.DLL, ODBCINT.DLL, and ODBCTRAC.DLL as defined in the ODBC 3.0 specification.

ODBC Administrator

The Microsoft ODBC Administrator is used to configure access to Advantage CA-Datcom/DB.

When you execute the ODBC Administrator from Windows, the following dialog appears. This dialog may vary according to the version of ODBC that is installed on the workstation.

Use the Help provided by the ODBC Administrator.



Click Add. The Create New Data Source window appears.

Add Data Source

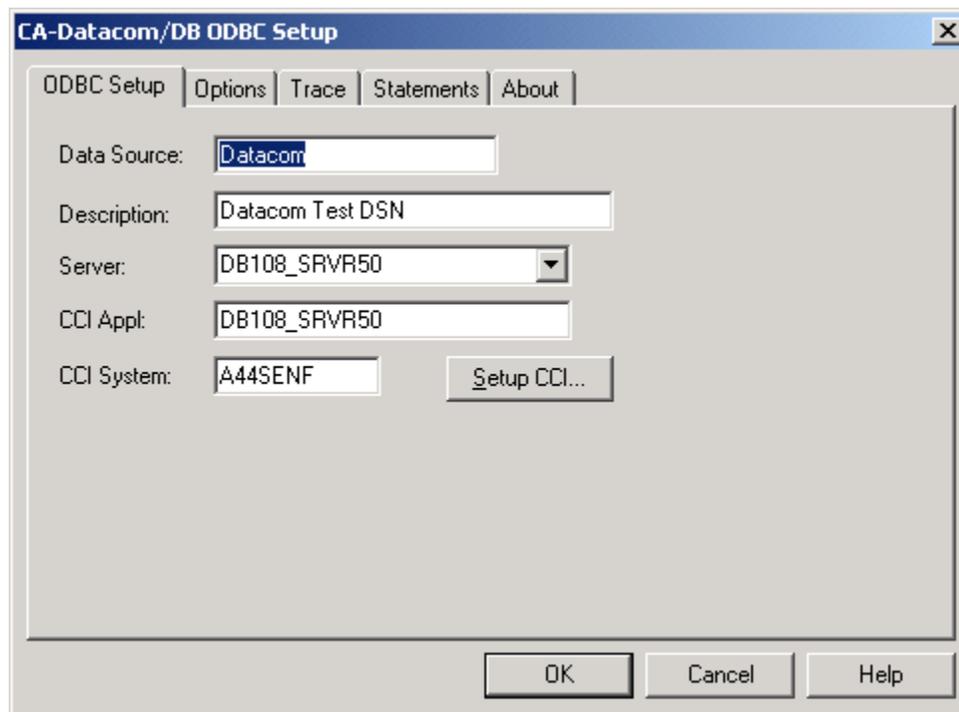
Select CA-Datcom/DB in the list of installed drivers.



Click Finish or press Enter.

ODBC Setup

Use the ODBC Setup window to define a data source or to configure the CAICCI connection, as directed by your administrator.



ODBC Setup Tab

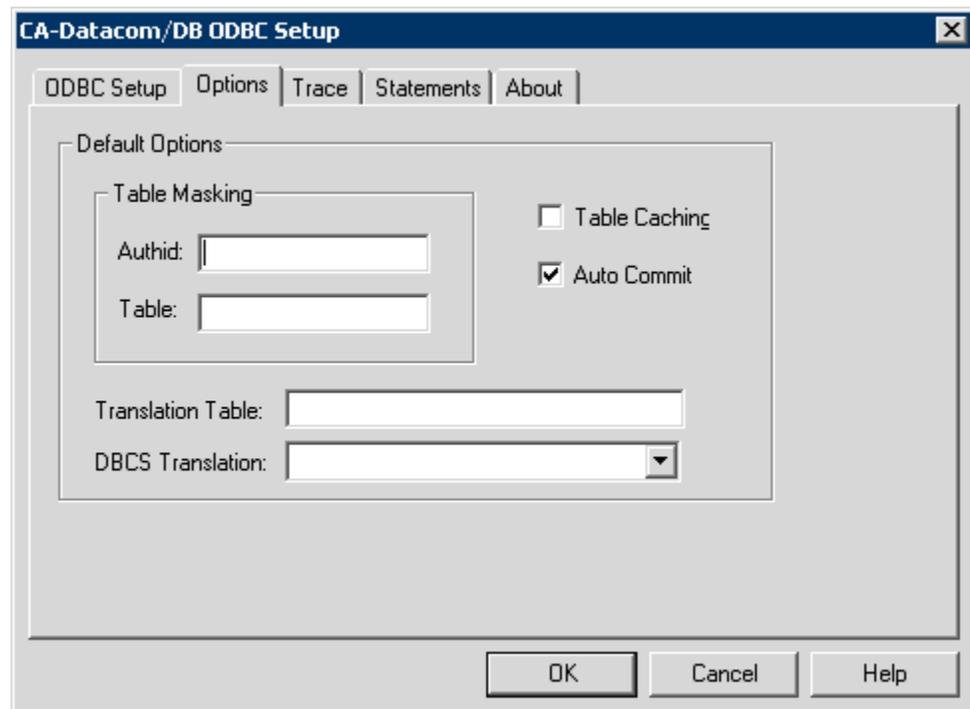
This tab contains the following fields:

- | | |
|-------------|---|
| Data Source | <i>(Required.)</i> Provide a standardized data source name or permit user to specify a name. |
| Description | <i>(Optional.)</i> Provide a standardized description or permit user to describe the new data source. |
| Server | <i>(Required.)</i> Enter the same server name as it is written in the startup JCL. This is the SERVERNAME= parameter in the Server startup JCL. |
| CCI Appl | <i>(Required.)</i> Enter the same Applid as it is written in the startup JCL. This is the APPLID= parameter in the server startup JCL. |
| CCI System | <i>(Required.)</i> Enter the CAICCI System ID of the mainframe environment to which the workstation is to connect when the data source is selected. This is the SYSID name specified in the CAICCI startup JCL. |

Setup CCI

The CA Common Services for z/OS and OS/390 or CA-CIS CAICCI is a communications interface used by several CA products, including Advantage CA-Datacom Server. When you are adding or modifying a data source, you can access the CAICCI Configurator windows by selecting Setup CCI on the Advantage CA-Datacom/DB ODBC Setup window. See the CAICCI documentation for details on the CAICCI Configurator.

Access to the CAICCI Configurator is also provided through an icon in the Windows Control Panel.

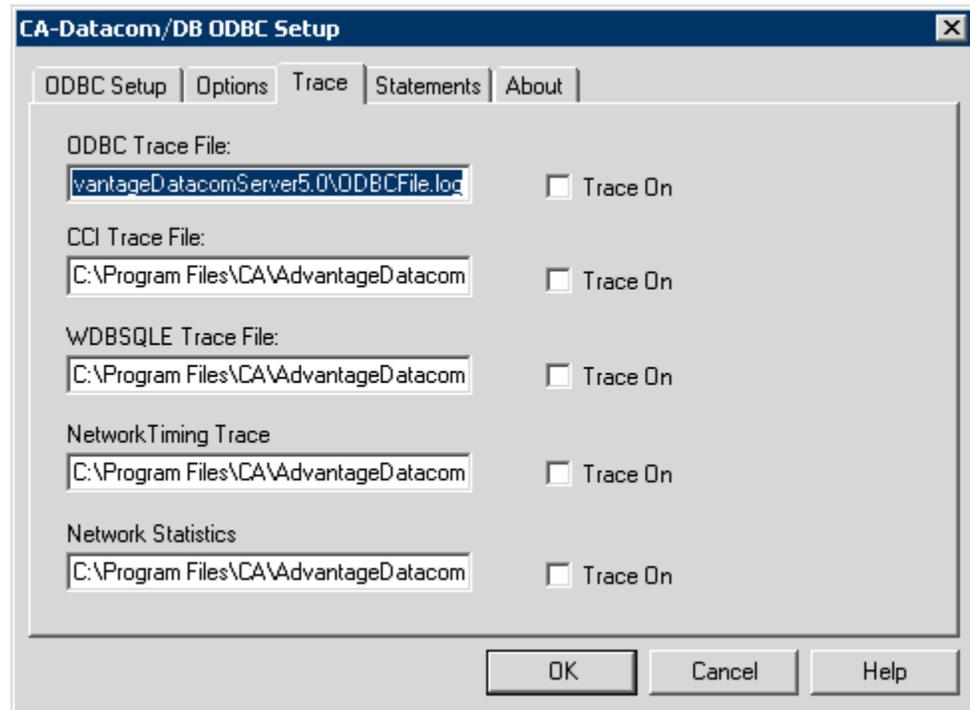


Options Tab

This tab contains the following fields:

Table Masking	(Optional.) Use these fields to specify masking for the lists of tables and views presented for user selection. (See the topic Limiting Lists of Tables and Views later in this chapter for details on how masking works and for an explanation of valid entries for these fields.)
---------------	---

Table Caching	<p>(Optional.) Check this box to specify whether the server is to perform table/view list caching. With caching enabled, when Advantage CA-Datacom Server retrieves a table/view selection list from the mainframe database, it stores the list locally (on the PC) for the duration of the connection. This allows the server to redisplay the table/view selection list quickly since it does not have to retrieve the list from the mainframe.</p> <p>Leave the box clear to specify that Advantage CA-Datacom Server retrieve a table/view list from the mainframe every time it needs to display it.</p>
Auto Commit	<p>(Optional.) Check this box to specify that Advantage CA-Datacom Server issue a COMMIT WORK after every SQL statement that causes an update to the database.</p> <p>Leave the box clear to specify that updates be committed only when the application explicitly issues a COMMIT WORK statement.</p>
Translation Table	<p>(Optional.) In this field, you can specify the file name of an ASCII to EBCDIC code page translation table to use for ACSII to EBCDIC data conversions. If you leave this field blank, Advantage CA-Datacom Server performs data conversions with an internal default ASCII to EBCDIC translation table. You can use the Translation Table Editor (see the topic Using the Translation Table Editor) to create a custom translation table to meet your requirements.</p>
DBCS Translation	<p>(Optional.) In this field, you can specify the type of double-byte character translation to perform. If you leave this field blank, Advantage CA-Datacom Server will not translate double-byte characters.</p> <p>Click the drop-down box to display a menu of the types of double-byte character translations available on your system.</p>

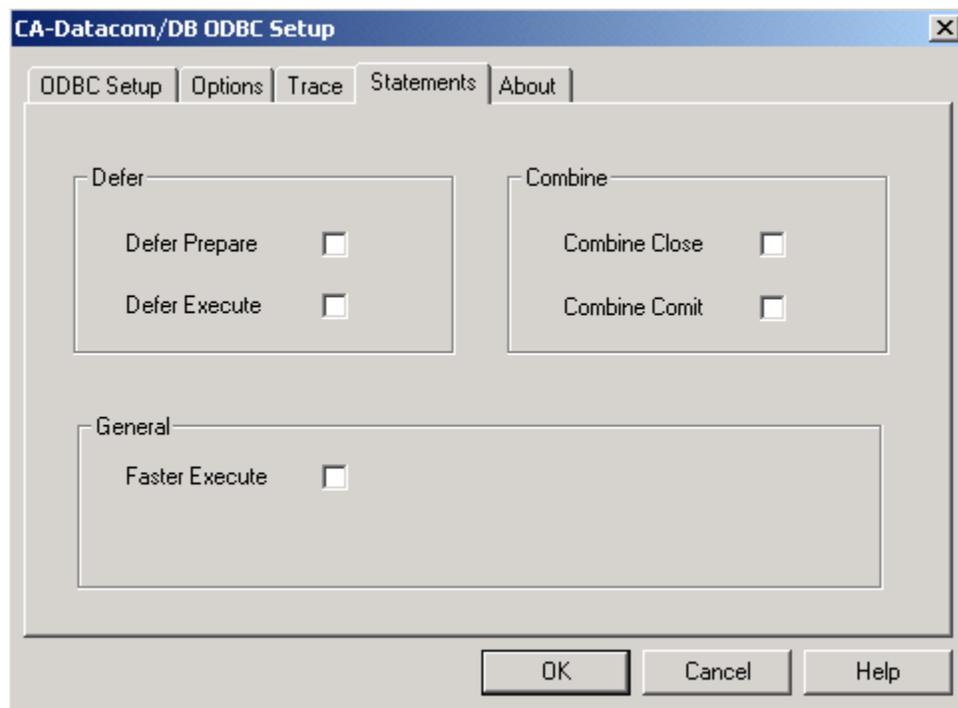


Trace Tab

You can specify that Advantage CA-Datcom Server trace ODBC calls and write that information to a trace file.

(Optional.) In each Trace File field, accept the default or specify the file name (including path) of a PC file in which Advantage CA-Datcom Server should write trace information.

To activate the desired trace, click the Trace On checkbox.



Statements Tab

You can modify the statement processing options of the server from this tab. These options were implemented in order to gain performance in web-based applications that typically return a small amount of data by reducing the number of database calls needed to complete processing in certain circumstances.

Note: The statement processing options may cause certain features to fail and should be turned on and tested with applications on a case-by-case basis. The failures can occur when statements are related in a non-serial method. The Combine Comit flag can result in invalid cursor status -135 return codes.

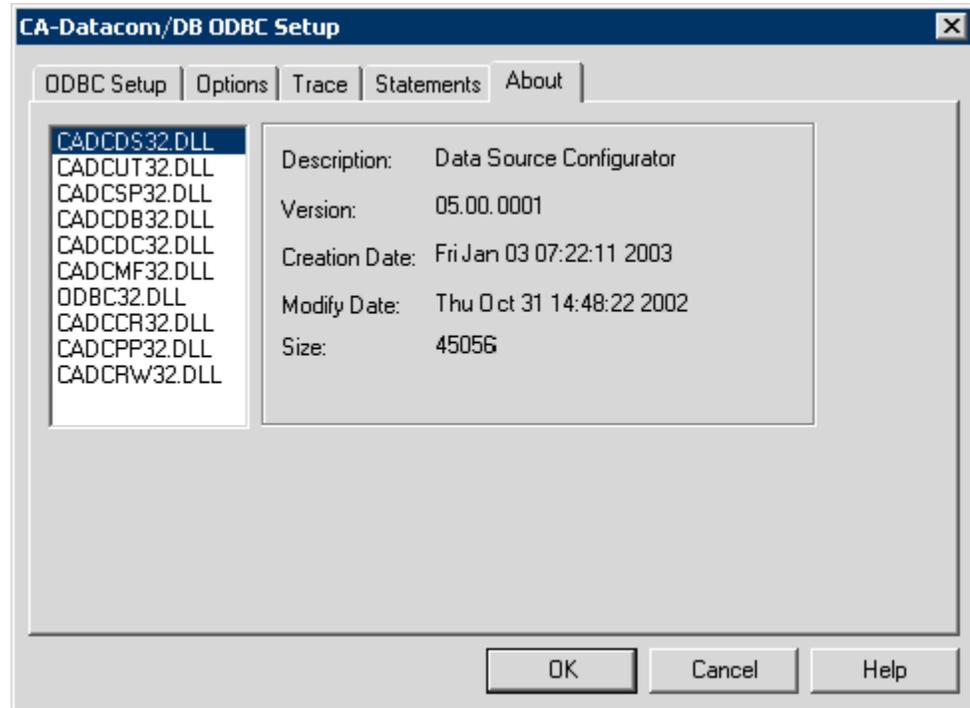
This tab contains the following fields:

- Defer Prepare Defers the prepare until the execute is issued.

 - Defer Execute Defers the prepare until the execute is issued and combines the database calls.

 - Combine Close Combines the close processing with the fetch processing.
 - Combine Comit Combines the close and comit processing with the fetch processing. This option is dependent upon the auto comit processing.
- Note:** This option is known to cause MS Access to fail and should not be used with MS ACCESS.

Faster Execute Turns Defer Prepare, Defer Execute, Combine Close, and Combine Commit on.



About Tab

Click this tab to display information about the version level of the Advantage CA-Datcom Server driver installed on this workstation.

Limiting Lists of Tables and Views

Many workstation applications show lists of available tables and views. These lists are built using the ODBC SQLTables function. When accessing mainframe Advantage CA-Datcom/DB, these lists can be long. Advantage CA-Datcom Server allows you to limit the tables and views the server returns in response to the SQLTables function.

To limit the list of tables and views for a specific data source, enter the AUTHID and Table in the fields in the “Table Masking” area.

Note: When an application issues the SQLTables function to obtain a Table list, the application can specify a table mask. This application mask will override the mask provided in this dialog except when it is a single wildcard character.

Authid (Optional.) To limit the listed tables and views to those associated with a specific SQL AUTHID, specify that AUTHID. To limit the listed tables and views to those associated with a group of AUTHIDs, use the wildcard characters described following.

Table (Optional.) To further limit the listed tables and views, you can specify a specific table or view name. To limit the listed tables and views to a subset of the tables and views associated with the specified AUTHID, use the wildcard characters described following.

WildCard Characters

When specifying the AUTHID and table or view name in the Table Masking fields, you can use the following wildcard characters:

Character	Meaning
%	Use the percent (%) character to represent a sequence of any number (including zero) of characters.
_	Use the underscore (_) character to represent a single character.

Note: To specify an AUTHID or table name that includes a percent or underscore character, precede the percent or underscore with a backslash (\) character. For example: to specify the table ACCT.DEPT_01, code ACCT in the Authid: field and DEPT_01 in the Table: field.

ODBC Connection Options

ODBC provides an application with three different ways to accomplish connection to a data source. The workstation applications determine what method of connection is used. The method of connection is not determined by the server or the user, but can be manipulated depending on how the application is designed. The strategies you use to provide options for each connection method determine the degree of control you exert over the workstation users.

The ODBC connection methods provide for a variety of connection approaches, and therefore require some planning on the part of the Database Administrator. The process of connection depends on the following information:

1. Which ODBC function the application uses to connect.
2. How the application presents that function's features to the workstation users.
3. How the data source configuration options are filled in.

Each of the three connection methods uses the data source's configured options and interacts with the user differently. The three connection methods are listed following:

- SQLBrowseConnect
- SQLDriverConnect
- SQLConnect

The following sections describe each connection method.

Connecting to Advantage CA-Datacom/DB Using a Connection String

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the driver which ODBC.INI section to use for the default connection information. Optionally, you can specify attribute=value pairs in the connection string to override the default values stored in ODBC.INI. These values are not written to ODBC.INI.

You can specify either long or short names in the connection string. The connection string has the form:

```
DSN=data_source_name; attribute=value; attribute=value; ...
```

An example of a connection string for Advantage CA-Datcom/DB is:

```
DSN=Datcom Server; SERVER=MUF1_SERVER;
SYSID=CCISYSID; APPLID=MUF1_SERVER;UID=USER1;
PWD=PASSWORD; TABLECACHE=N;
TABLLISTMASK=; TRANSTABLE=''
```

The following list gives the long and short names for each attribute, as well as a description:

DSN (Data Source Name)	<i>(Required.)</i> A string that identifies an Advantage CA-Datcom/DB data source configuration in ODBC.INI. Examples include "Accounting" and "Datcom-Serv1."
Server	<i>(Optional.)</i> Specifies the SERVERNAME= parameter in the server startup JCL.
SYSID	<i>(Required.)</i> Specifies the SYSID name specified in the CAICCI startup JCL. It is the CAICCI system ID of the mainframe environment that the workstation connects to when a data source is selected.
APPLID	<i>(Required.)</i> Specifies the APPLID= parameter in the server startup JCL.
UID	<i>(Optional.)</i> Specifies your user ID.
PWD (Password)	<i>(Optional.)</i> Specifies your password.
TABLECACHE	<i>(Optional.)</i> Set this value to Y to cache a table list retrieved from the mainframe on the PC. On subsequent calls for a table list, the list can be retrieved from the workstation.
TABLELISTMASK	<i>(Optional.)</i> Specifies a mask for the list of tables and views presented for user selection.
TRANSTABLE	<i>(Optional.)</i> Specifies the name of an ASCII to EBCDIC translation table.

SQLBrowseConnect: Selective Server Specification

The driver does not consult the data source configuration options in the SQLBrowseConnect form of connection. The user fills in only the data source name when configuring the data source, and only one data source must be configured for Advantage CA-Datcom/DB access.

The advantage of this form of connection is that the list of available servers is provided at connection time, which allows the server administrator to change the names of servers without having to inform workstation users of the change. They simply get a different list of servers from which to choose. The user is configured with a single data source identifying any single member of the server group.

SQLBrowseConnect is the only form of connection that allows the workstation user to select from a list of currently available servers.

The disadvantage of this form of connection is that the application must include additional programming to collect the CAICCI IDs, the server name, user ID, and password from the workstation user instead of relying on the ODBC driver to use any values installed for that data source as defaults. However, the SQLBrowseConnect form of connection allows the application to store the results of a connection process and automatically reconnect to that same server without having to collect the information again, by using SQLDriverConnect.

When an application uses this connection method, the workstation user must supply a data source name, unless the application already knows one. If the connection string does not include a data source name, the user is asked to choose one from those installed on the workstation. If the connection string omits an option, the ODBC Driver prompts the user for the options with the Connection window, which is similar to the ODBC administrator window. Values specified in the ODBC administrator window when the data source was installed are displayed as default values in the Connection window.

SQLDriverConnect: Selective Option Specification

The advantage of this method of connection is that the data source configuration options are used. Once the data source is configured, the workstation user does not reenter the data source options again. Options left blank when configuring the data source are collected from the user at each connection attempt, which can be appropriate for user ID, password, and server name. It is possible to connect to any server in the server group (identified by CAICCI system and application IDs). It is also possible to connect to any server by overriding the configured CAICCI IDs and the server name.

The disadvantage of this method of connection is that the user cannot get a list of available servers. Only the server name configured for the data source is available unless the user overrides it with a server name published by the server administrator.

An application using this form of connection must build a connection string that specifies values for the data source name and for the five data source options required by Advantage CA-Datcom/DB. The options required are:

- SERVER (Server Name)
- UID (User ID)
- PWD (Password)
- SYSID (System ID)
- APPLID (Application ID)

This string can be constructed by the application to match an earlier connection established with the SQLBrowseConnect.

If the connection string does not include a data source name, the user is asked to choose one from those installed on the workstation. If the connection string omits an option, the ODBC driver asks the user for the options with a window called the Connection window, which is similar to the ODBC Setup window. Values specified in the Setup window when the data source was installed are displayed as default values in the Connection window.

The Connection window is bypassed only if all options are specified during data source configuration, or are provided by the application in the connection string. However, the application may specifically request the ODBC driver to display the window.

SQLConnect: Automatic Option Specification

With this form of connection, the application relies on all driver-specific data source options being or having been specified when the data source is installed. The application asks only for user ID and password. The server name, CAICCI IDs, and so forth, must have been configured for that data source for the connection to succeed.

The advantage of this approach is that the workstation user does not have to deal with the Advantage CA-Datcom/DB options, such as server name, except when following the server administrator's instructions while configuring a data source. One data source equals one server.

A disadvantage of this approach is that a different data source must be configured for each server to which the user connects. Since the user only selects a data source name, the names chosen for the data source and server must be similar. In contrast, the other connection approaches lets the user use a single data source to access one or more servers. In this case the data source equates to all Advantage CA-Datcom/DB servers, instead of a single server.

Another disadvantage is the requirement that configuration changes on the mainframe must be reflected in configuration changes to the configured data sources at the workstation.

Default Data Source Name

Whenever the workstation user or an application must choose a data source name, and the name is left blank, the Microsoft ODBC Driver Manager uses a data source named *Default*. To use the default, tell the workstation user to configure a data source named *Default* which specifies the data source options necessary for that user and tell the user to always leave the data source name blank when asked for a data source.

Connection Window (Advantage CA-Datcom/DB Logon)

In addition to the Advantage CA-Datcom specific configuration dialog, Advantage CA-Datcom Server provides an “Advantage CA-Datcom/DB LOGON” dialog to address Advantage CA-Datcom specific information. When you are running an application that uses Advantage CA-Datcom Server to access mainframe Advantage CA-Datcom/DB, a Connection dialog like the following example may appear, requesting signon information. If you are unsure of the information needed, contact the ODBC administrator for values to specify in this window.

The type of connection call that is made to the Advantage CA-Datcom/DB ODBC Driver (SQLDriverConnect) or the omission of required information to the connection call determines the appearance of this dialog. The ODBC Driver tries to connect using the values provided. The ODBC Driver asks the user for the options with the Advantage CA-Datcom/DB Logon window:

The screenshot shows a dialog box titled "CA-Datcom/DB Logon". It has a "Logon" tab. The fields and their values are:

- User ID: [Empty]
- Password: [Empty]
- New Password: [Empty]
- Verify Password: [Empty]
- Server Name: PROD_SERVER
- System ID: CCIPROD
- Appl ID: PROD_SERVER

Buttons: OK, Cancel

User ID	Enter the ID of the workstation user who is attempting to connect.
---------	--

Password	Enter the password, if any, of the workstation user who is attempting to connect.
New Password	Enter a new password for the workstation user who is attempting to connect.
Verify Password	Enter the new password a second time to confirm it.
Server Name	Enter the name of the Advantage CA-Datcom Server to which the user wants to connect (as it is specified in the SERVERNAME= parameter of the Server startup JCL). See the mainframe <i>Advantage CA-Datcom Database Server Option Installation and Maintenance Guide</i> for your operating system for details on the SERVERNAME= parameter.
System ID	Enter the Unicenter TNG Framework for OS/390 or CA-CIS System ID of the mainframe environment to which the workstation is to connect when the data source is selected.
Appl ID	Enter the APPLID as it is shown in the startup JCL. This is the APPLID= parameter of the server startup JCL. See the mainframe <i>Advantage CA-Datcom Database Server Option Installation and Maintenance Guide</i> for your operating system for details on the APPLID= parameter.

JDBC Driver Configuration and Administration

The Advantage CA-Datcom/DB JDBC Driver is dependent upon the JDBC Driver Manager. The driver manager is a static class that provides services to connect to JDBC drivers. This class is provided by JavaSoft and does not require the JDBC Driver to perform any implementation. The primary function of the driver Manager is to load the JDBC driver. Once the driver is loaded, the driver has very little interface with the driver manager.

Note: See /SAMPLES/JCF/ in the installation directory for the JDBC programs you can use to test your installation.

To use a JDBC driver, the driver manager must first load it. The loading of the JDBC driver does not connect to Advantage CA-Datcom Server; it only creates the environment. For the JDBC Driver Manager to load the proper JDBC driver, the request to the driver manager includes a connection URL. The Advantage CA-Datcom JDBC driver URL allows the driver to be configured much like the ODBC driver. Each JDBC driver has a unique URL.

The format of a JDBC URL is:

```
jdbc:subprotocol:subname
```

Where *jdbc* specifies JDBC, *subprotocol* indicates the access method, and *subname* is a name that has significance for the *subprotocol* being used.

The Advantage CA-Datcom Server JDBC Driver URL is:

```
Subprotocol = datacom
```

```
Subname =//host:port/ServerName=,SystemID=, ApplicationID=, UserID=, Password=,
NewPassword=, CCIHost=, CCIPort=, TransTable=, Trace=, LogTimeout=, TraceJdbc=,
TraceLife=, TraceNative=, SnapJdbc=, SnapNative=, SnapObject=, SnapBytes=,
Include=, Exclude=
```

host:port

(Optional.) Enter the hostname and port of the Advantage CA-Datcom Server Java Server with which this connection should be established.

ServerName	(Required.) Enter the same server name as it is written in the startup JCL. This is the SERVERNAME= parameter in the server startup JCL.
SystemID	(Required.) Enter the CAICCI System ID of the mainframe environment to which the workstation is to connect when the data source is selected. This is the SYSID name specified the CAICCI startup JCL.
ApplicationID	(Required.) Enter the same Applid as it is written in the Advantage CA-Datcom Server startup JCL. This is the APPLID= parameter in the Server startup JCL.
UserID	(Optional.) Enter a valid User ID.
Password	(Optional.) Enter a valid Password.
NewPassword	(Optional.) Enter a valid new Password. It is the applications responsibility to force the user to validate, and reenter the password.
CCIDHost	(Reserved.)
CCIPort	(Reserved.)
TransTable	(Optional.) The translation table to be used for the EBCDOC to ASC01 translation.
Trace	(Optional.) Trace level. Valid entries are 1, 2, and 3.
LogTimeout	(Optional.) Time in seconds to elapse before a time out occurs.
TraceJdbc	(Optional.) Trace detail flag. Valid entries are Y or N.
TraceLife	(Optional.) Trace detail flag. Valid entries are Y or N.
TraceNative	(Optional.) Trace detail flag. Valid entries are Y or N.
SnapJdbc	(Optional.) Trace detail flag. Valid entries are Y or N.
SnapNative	(Optional.) Trace detail flag. Valid entries are Y or N.
SnapObject	(Optional.) Trace detail flag. Valid entries are Y or N.
SnapBytes	(Optional.) Trace detail flag. Valid entries are Y or N.
Include	(Optional.) Trace detail flag. Valid entries are ca.datcom package class names.
Exclude	(Optional.) Trace detail flag. Valid entries are ca.datcom package class names.

Using the Java Server

Advantage CA-Datacom Server installation optionally installs the Java Server. The Java Server is used to allow the JDBC driver to connect to the Workstation Server through TCP/IP, the native Java protocol. This allows the JDBC Server to operate as a “Type 4” JDBC driver.

Note: /SAMPLES/JCF/ cadcjc.html in the installation directory can be used to test your installation.

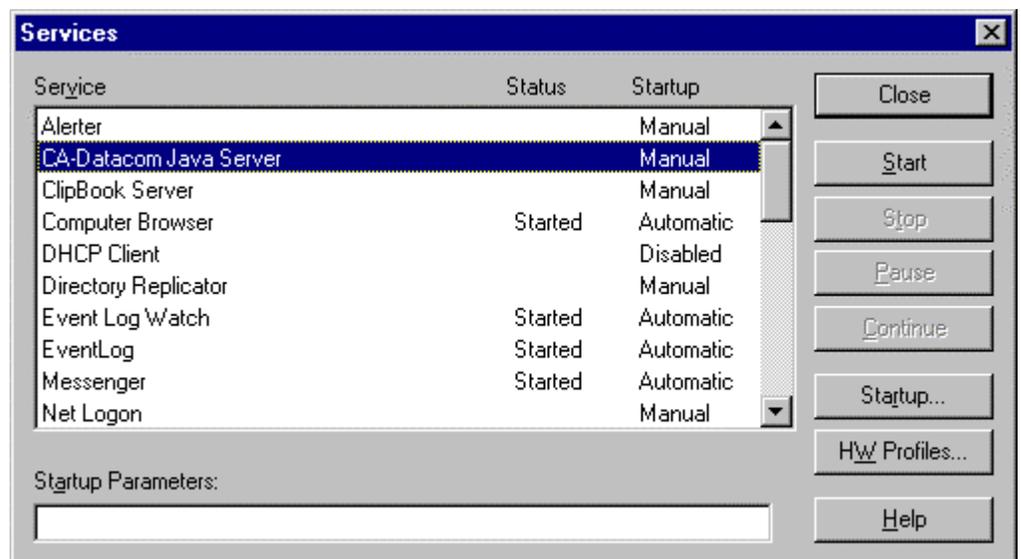
Installing the Java Server



Services

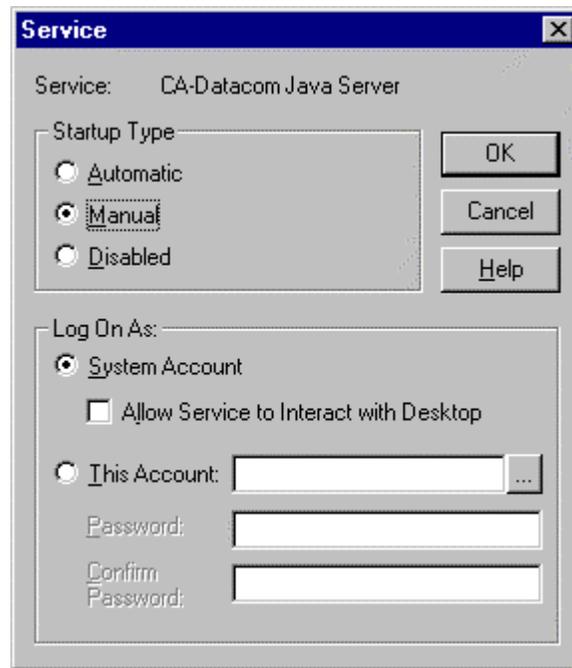
The Java Server is installed as an NT Service. It is accessible from the Control Panel services icon.

Select the Advantage CA-Datacom Java Server in the Services control.



Click Startup.

The Service configuration dialog appears, which is a generic dialog used for setting common services parameters. The Java Server specific parameters are set using the Java Server Config.



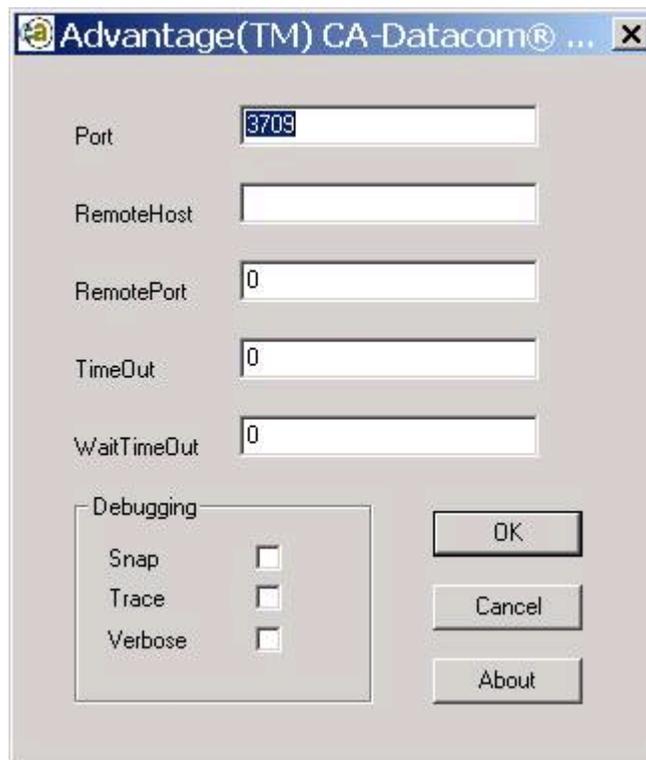
Select Help for more information.

Java Server Config

To open the Java Server Config, follow these steps:

1. From the Start menu select Programs.
2. Select Advantage CA-Datacom Database Server Option
3. Select Server Option.
4. Select Java Server Config.

The Java Server Config screen appears.



The Java Server Config must be used to configure the Java Server. An example of the entries are provided following:

Port	<i>(Optional.)</i> The TCP/IP Port to use for communication. This must match the Port entered in the connection URL. The default is 3709.
RemoteHost	<i>(Optional.)</i> The TCP/IP name of the remote Advantage CA-Datacom Java Server.
RemotePort	<i>(Optional.)</i> Enter the same Applid as it is written in the Advantage CA-Datacom Server startup JCL. This is the APPLID= parameter in the server startup JCL.
Snap	<i>(Optional.)</i> Trace detail flag. Check box to activate.
Timeout	<i>(Optional.)</i> Time in seconds the Java Server will wait for a response to complete before timing out.
Trace	<i>(Optional.)</i> Trace detail flag. Check box to activate.
Verbose	<i>(Optional.)</i> Trace detail flag. Check box to activate.
WaitTimeout	<i>(Optional.)</i> Time in seconds the Java Server passes to CCI to use for its time out value.

Using the Translation Table Editor

The Advantage CA-Datcom Translation Table Editor lets you create and modify tables that control the translation of character data between the EBCDIC character set used on the host and the ASCII character set used on a PC. You can select Country Extended Code Pages for the host and PC. You can also edit the ASCII to EBCDIC and EBCDIC to ASCII translation tables to specify character conversions not provided by the standard code pages.

To start the Translation Table Editor, double-click its icon in the Advantage CA-Datcom Server group.



Building a Translation Table

To build a new translation table:

1. From the File menu, select New.
2. Edit the translation table (see the following sections).
3. From the File menu, select Save As... to assign a name and save the file. Translation table files have the extension .TAB.

To modify an existing translation table:

1. From the File menu, select Open...
2. Select the translation table file that you want to modify.
3. Edit the translation table (see the following sections).
4. From the File menu, select Save to save the modified file.

Editing a Translation Table

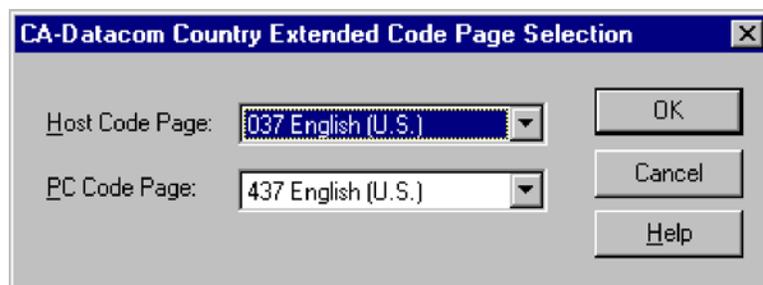
The Translation Table Editor provides facilities for performing two different types of edits to a translation table. Select from these choices on the Edit drop down list:

- Editing the actual ASCII to EBCDIC or EBCDIC to ASCII character mapping.
- Selecting the code pages to be used on each platform.

Selecting Code Pages

Country Extended Code Pages provide additional characters beyond the standard ASCII and EBCDIC character sets. These additional characters are specific to individual countries and languages.

Selecting Code Pages from the Edit menu displays the following window where you can specify different code pages for the host computer and PC.



From the Host Code Page pull-down menu, select the code page you want Advantage CA-Datcom Server to use for the EBCDIC character set. From the PC Code Page pull-down menu, select the code page you want Advantage CA-Datcom Server to use for the ASCII character set.

When you click OK, the Translation Table Editor combines the selected PC and host code pages to create the ASCII to EBCDIC and EBCDIC to ASCII translation tables.

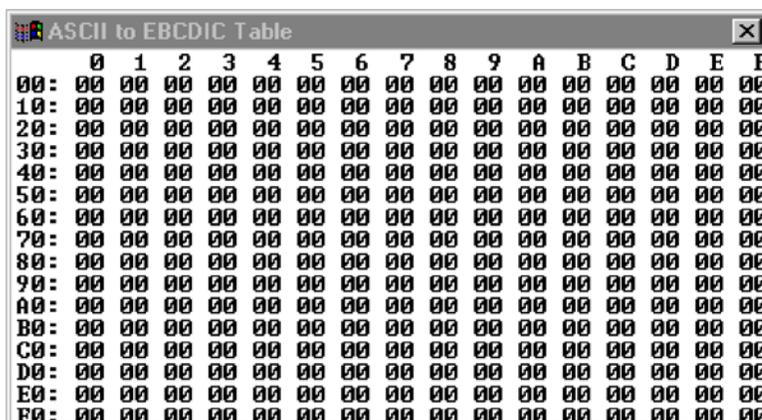
To review or edit the resulting character mapping, select either the EBCDIC to ASCII or the ASCII to EBCDIC option from the Edit menu. (See the following section.)

To close the dialog without creating a new translation table, click Cancel.

Editing Character Mapping

Although CA provides code pages for many languages, there may be conversions you require that are not supported by these standard code pages. You can use the edit windows to customize the translation tables created from the PC and host code pages you have specified.

There are two edit windows, one for the ASCII to EBCDIC translation table and the other for the EBCDIC to ASCII translation table. To display these windows, select the ASCII to EBCDIC or the EBCDIC to ASCII option from the Edit menu. Each window displays an array of the 256 hexadecimal values. Each entry in the array represents the output character set code value indexed by the input character set code value.



The screenshot shows a window titled "ASCII to EBCDIC Table" with a close button in the top right corner. The window contains a grid of 256 hexadecimal values, organized by input character set code value (rows) and output character set code value (columns). The columns are labeled 0 through F, and the rows are labeled 00 through FF. All values in the grid are currently set to 00.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
10:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
60:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
90:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
FF:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

To edit the displayed translation table, locate the cursor over the value to be changed and left-click. Then, type the new hexadecimal value. The editor ignores any characters entered other than 0-9, a-f, and A-F.

You can also use the following keys to move the cursor around the translation table edit window:

Key	Resulting Movement
Cursor Left	Left one digit
Cursor Right	Right one digit
Cursor Up	Up one row
Cursor Down	Down one row
Home	Beginning of row
End	End of row
PageUp	Top row
PageDown	Bottom row
Enter	Beginning of next row
Ctrl+CursorLeft	Left one entry
Ctrl+CursorRight	Right one entry
Ctrl+Home	Beginning of table
Ctrl+End	End of table

Using the Advantage CA-Datacom Server Preprocessors

The following preprocessors of Advantage CA-Datacom Server support access to mainframe Advantage CA-Datacom/DB:

- COBOL preprocessor
- C preprocessor

COBOL Preprocessor

The Advantage CA-Datacom Server supports access to mainframe Advantage CA-Datacom/DB data through CA-Realia II Workbench COBOL programs containing embedded SQL statements.

Note: The Advantage CA-Datacom Server COBOL preprocessor currently:

- Supports uppercase COBOL source code only. Ensure that all of your code that may be referenced by the Advantage CA-Datacom preprocessor does not contain mixed case. This includes all DECLARED host variables as well as entire EXEC SQL commands.
- Supports the ASCII code set and COMP-5 integers only. This prevents using it successfully with Advantage™ CA-Realia® Runtime for CICS (Advantage CA-Realia CICS Runtime).

Using SQL Statements

The Advantage CA-Datcom Server supports embedding Advantage CA-Datcom/DB SQL statements in CA-Realia II Workbench COBOL programs, which are passed through the Advantage CA-Datcom Server COBOL Preprocessor. The COBOL preprocessor is completely compatible with the mainframe Advantage CA-Datcom/DB Version 10.0 COBOL SQL Preprocessor. When you preprocess a COBOL program with the Advantage CA-Datcom Server COBOL Preprocessor, Advantage CA-Datcom creates the plan which contains the executable form of the SQL statements on the mainframe just as it would if you were preprocessing the program on the mainframe. When the program is executed, the mainframe plan is executed. Advantage CA-Datcom Server automatically translates data between mainframe and workstation formats when it passes it between the workstation and mainframe.

With the Advantage CA-Datcom Server, you can write client-server programs, which use Advantage CA-Realia II Workbench. Advantage CA-Realia II Workbench can be used to execute the business logic programs and access the mainframe Multi-User Facility. You can port existing COBOL programs that access Advantage CA-Datcom/DB data from the mainframe to the workstation and create client-server applications.

This approach has a performance advantage over using a preprocessor that embeds ODBC calls in the application. ODBC calls include the binding of the SQL statements at execution time. The Advantage CA-Datcom Server Preprocessor creates static plans before the application executes. Therefore, when the program is executed, there is no binding overhead at all. Advantage CA-Datcom Server SQL programs also take full advantage of the Advantage CA-Datcom SQL Block Transfer feature to minimize communication time.

COBOL SQL Preprocessor Setup

Use Explorer or a similar tool to copy the following files to the specified location.

The default install directory for the Advantage CA-Datcom Server Advantage CA-Realia II Workbench components is:

```
C:\Program Files\CA\AdvantageDatcomServer5.0\Realia Workbench Precompiler.
```

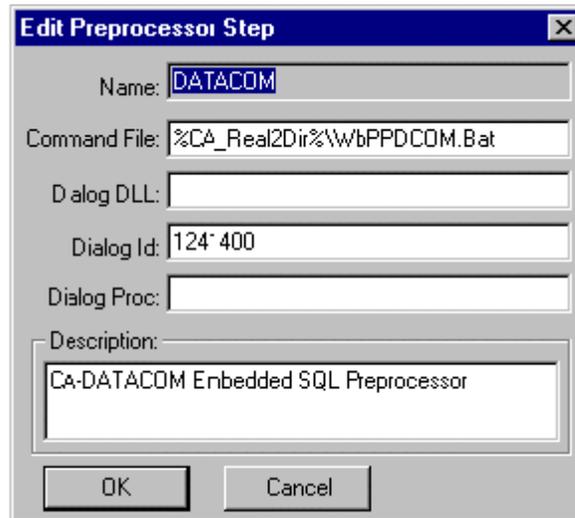
The default install directory for the Advantage CA-Realia II Workbench is:

```
\CAWB30 or CAWB31
```

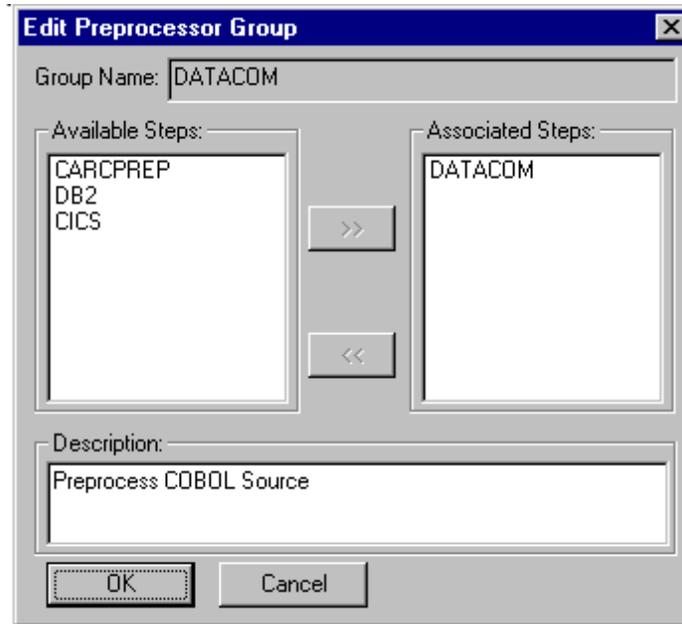
If you have selected a different install directory, make the appropriate adjustments.

Copy From	Copy To
Severwbininstalldirectory\realdcom.exe	workbenchinstalldirectory\realdcom.exe
Severwbininstalldirectory\Wbppdcom.bat	workbenchinstalldirectory\Wbppdcom.bat
Severwbininstalldirectory\cadcrw32.lib	workbenchinstalldirectory\lib\cadcrw32.lib

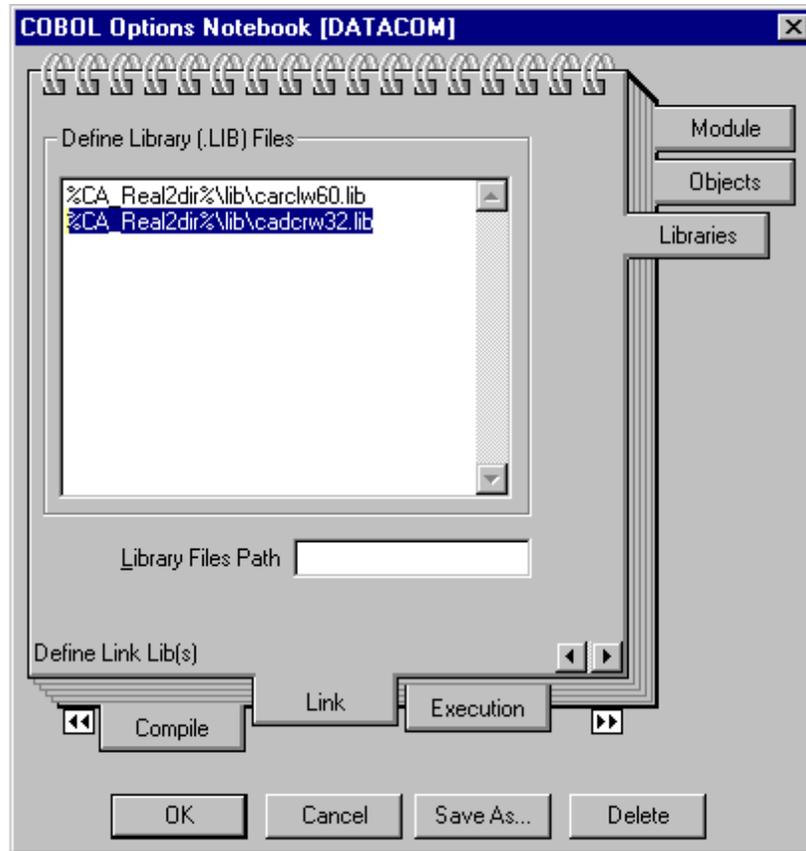
The workbench needs to be enhanced to include proper setup to use the correct options. Select the Tools->Preprocessor Maintenance->Edit Steps function to create a step with the name DATACOM.



Then select the Tools->Preprocessor Maintenance->Edit Groups function to create an *Advantage CA-Datacom* group.

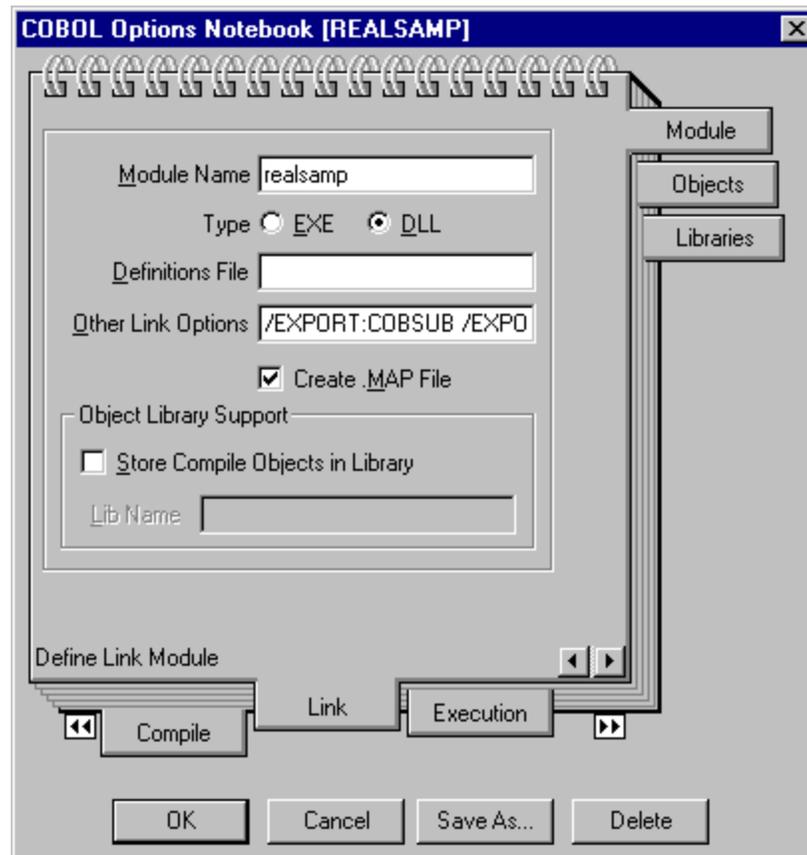


You must add the Datacom library to the link step setup. The library name is `cadcrw32.lib`. For simplicity, this can be added to the system options file.



If you are linking a COBOL .dll that will be called from another program, such as Visual Basic or Visual C++, enter a parameter as follows:

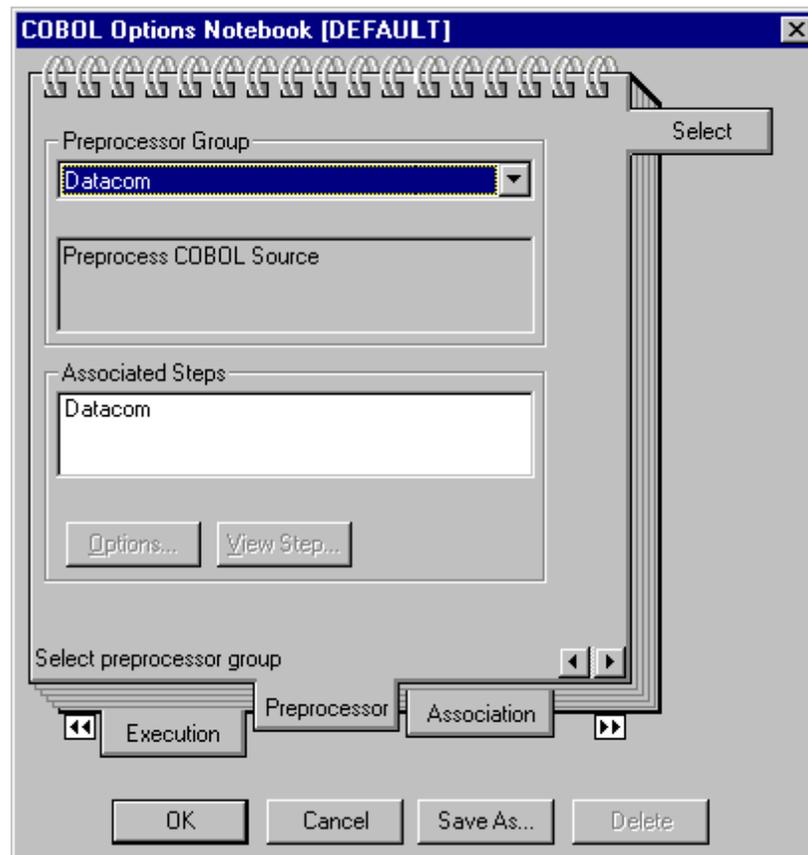
- Click the Module tab.
- In the Other Link Options field, enter `/EXPORT:entrypt1` where `entrypt1` is the name of your entry point. You can add more parameters in the same field by leaving a space between each one.



Using the COBOL SQL Preprocessor

Once installed into your Advantage CA-Realia II Workbench environment, the Advantage CA-Datcom Server Preprocessor is selectable as a Preprocessor Group on the Preprocessor tab of the Options dialog. When using Advantage CA-Realia II Workbench to compile a COBOL program with embedded SQL, select Advantage CA-Datcom Server as the Program Type to create a (mainframe) Advantage CA-Datcom/DB plan. When building the program, Advantage CA-Realia II Workbench invokes the Advantage CA-Datcom Server Preprocessor before invoking the Advantage CA-Realia II Workbench COBOL.

When the Preprocessor starts to execute, Advantage CA-Datcom Server connects to the mainframe Data Source you have selected (see "Preprocessor Options" following). You must have your workstation's communications software loaded and ready to use before you start to build the program.



COBOL Preprocessor Options

Once you select Advantage CA-Datacom Server as the preprocessor, you must set the preprocessor options. Programs using embedded SQL must be processed by the Advantage CA-Datacom preprocessor named RealDCOM.EXE. You can modify your source code by inserting a ***\$DBSQLOPT SOURCE=ODBCName** statement into the beginning of the program source. Or, you can edit your program or application .OPT file and include a RealDCOM preprocessor section. Other options can be set using this method as well.

```
[RealDCOM Preprocessor]
```

```
source=ODBCName
```

Note: All the mainframe Advantage CA-Datacom/DB Preprocessor options are available except the SQLMODE=DB2 and SQLMODE=DB2A86 options. Specify preprocessor options in the COBOL program source itself as \$DBSQLOPT lines in the same way as you would in a mainframe COBOL program. **We recommend specifying PLNCLOSE=T in your preprocessor options.** For details on preprocessor options, see the *Advantage CA-Datacom/DB SQL User Guide*.

The following options are specific to the workstation and are not available for the mainframe.

Connection Type	Specify Pass, Prompted, or Silent.
conntype=	<p>The Pass form of connection functions similar to the Silent connection (see following). The difference in functionality is that the User ID and Password can be passed to the program at runtime. This connection type is useful for applications that allow a user to signon and pass the User ID and Password to the COBOL program.</p> <p>If you specify Prompted, the Advantage CA-Datcom/DB Logon window (documented in the chapter, "ODBC Driver Configuration and Administration") can still be suppressed by specifying a Data Source name (see the following) and by supplying all the information needed for connection to that data source in its control file.</p> <p>For the Silent form of connection, the same information is used by the ODBC Driver when connecting, but no Logon window is presented. If the Data Source name is not specified, the ODBC Driver uses the Default Data Source (see "ODBC Driver Configuration"). All items except UserID and Password may be specified for a Data Source in the ODBC Administration Configuration window. If the Multi-User Facility you are connecting to is secured or if the server you are connecting to has SIGNON=YES in its start-up JCL, the UserID and Password fields are required.</p> <p>Valid Entries: PASS, PROMPTED, or SILENT</p> <p>Default Value: SILENT</p>
UserID	The user ID required to log on to the mainframe.
userid=	<p>Valid Entries: 1- to 8-byte user ID</p> <p>Default Value: None</p>
PASSWORD	The password value to logon to the mainframe.
password=	<p>Valid Entries: 1- to 20-byte password</p> <p>Default Value: None</p>

Data Source source=	The name of the ODBC Data Source to which the ODBC Driver should connect. This must be an Advantage CA-Datcom/DB Data Source. If the Data Source name is omitted, the ODBC Driver attempts to connect to the Default Data Source. (The Default Data Source is the one named Default in the Windows registry, ODBC.INI and DCODBC.INI files.)
------------------------	--

The same connection method and Data Source name that you pick on the Program Options window is embedded in the COBOL program you are preprocessing for use when the program starts to execute. If the program is executed on the same machine as the one on which it is preprocessed, the preprocessing serves as a preview of exactly what happens to the end user when the program is executed. If the program is executed on a different machine, you must tell the workstation user about the Data Source name embedded in the program and make sure that the end user has configured that Data Source to point to the same Advantage CA-Datcom Server that was used to create the plan. If you have not specified a Data Source name and your Default Data Source was used, when the program executes, it uses the Default Data Source defined on the workstation on which it is executing.

Note: If the wrong Advantage CA-Datcom Server is selected when the program is executed (that is, a different Server than the one on which the program was preprocessed), the program gets a PLAN TIMESTAMP MISMATCH error or PLAN DOES NOT EXIST error.

Valid Entries: A valid Data Source name

Default Value: (blanks)

COBOL Preprocessor Listing

If there are any preprocess-time errors in your COBOL program, Advantage CA-Realia II Workbench stops the build process and displays the Error Listing file. This file is similar to the mainframe SQL Preprocessor listing file, and shows each line of the input COBOL program with any errors shown where they occurred when preprocessing. The error and warning summaries are at the end of the listing.

Warning Level

The Advantage CA-Datcom Server Preprocessor modifies the input COBOL program, adding various variable declarations and procedural code. One of the VALUE clauses that it adds causes a Warning error 0279 because the value is technically too large for a PIC S9(9) data item. These warnings can be ignored, and you should set the Minimum Error Level to 'E' instead of 'W' in the Listing Options on the Program Options window, to keep the build from terminating when the warning errors are issued by the COBOL compiler.

Linking Your DLL

In Advantage CA-Realia II Workbench, COBOL programs are linked together into DLLs or EXEs, which are then executed or called from other programs. Since the COBOL programs as output by the Advantage CA-Datcom Server Preprocessor include calls to DBSQLE and DBXHVPK just as they do on the mainframe, those entry points must be resolved when the DLLs or EXEs are linked. This is accomplished by importing three Advantage CA-Datcom Server CADCRW32.LIB files into your Advantage CA-Realia Workbench project and positioning them under the DLL name (same level as the COBOL programs).

Connecting and Disconnecting

The DBSQLE module connects to the mainframe Advantage CA-Datcom Server indicated by the Data Source name chosen at preprocessor time when the first SQL statement is executed. The DBSQLE module recognizes when the first statement is executed, and makes the connection. All subsequent SQL statements in that same DLL go through that same connection. This means that, as on the mainframe, all COBOL programs which end up calling the same copy of the DBSQLE module use the same connection to the Multi-User Facility, and therefore a COMMIT WORK from any of the COBOL programs commits the work done by all of the COBOL programs up to that point. If you have multiple COBOL programs linked together in the same DLL, use the USRNTY=NONE plan option to prevent the preprocessor from generating the DBMSCBL entry point in all of the COBOL programs.

Locking, Transactions, and Abends

COMMIT WORK and ROLLBACK WORK statements work as they do on the mainframe to delimit transactions. As in mainframe DBMS programming, you should issue COMMIT WORK statements at application-specific check points, in case the PC abends or power is lost, and to avoid having too many concurrent locks on the mainframe. Remember that your PC program is actually accessing the mainframe Multi-User Facility in real time.

If your Advantage CA-Realia II Workbench program ends before the CLOSE command is issued, its task remains “stranded” in the mainframe server. This is because the server does not know that your PC has abended. The stranded task can be seen by using the mainframe Server Communication Utility (SVCOMPR) to get a status of the server. The stranded task can then be canceled, or it may time out and be canceled automatically. In either case, a ROLLBACK WORK is issued on behalf of that task to roll back any uncommitted changes.

Mixing SQL with Navigational Commands in COBOL Programs

No restrictions exist for mixing SQL statements and navigational commands in the same COBOL program. The execution of an SQL COMMIT WORK statement is the equivalent of the navigational COMMIT command, and either way of doing a commit commits all changes made by that program since the last commit point, whether the changes were made using navigational commands or SQL statements. Similarly, the SQL ROLLBACK WORK statement is the equivalent of the navigational ROLBK command.

To disconnect from the mainframe, the application can call either DBSQLE or DBNTRY with a CLOSE command. This terminates the application with an implicit COMMIT WORK.

C Preprocessor

The Advantage CA-Datacom Server supports access to mainframe *Advantage CA-Datacom/DB* data through the C language containing embedded SQL statements.

Note: The Advantage CA-Datacom Server C Preprocessor at this time cannot support the creation of stored procedures. The stored procedures must be created on the same platform as Advantage CA-Datacom/DB.

The C preprocessor is compliant with any ANSI C compiler. The C code is run through the preprocessor to produce ANSI C code that can then be input into the C compiler of choice.

Using SQL Statements

The Advantage CA-Datcom Server supports embedding Advantage CA-Datcom/DB SQL statements in C programs, which are passed through the Advantage CA-Datcom Server C Preprocessor. The Advantage CA-Datcom Server C Preprocessor is completely compatible with the mainframe Advantage CA-Datcom/DB Version 10.0 C SQL Preprocessor. When you preprocess a C program with the Advantage CA-Datcom Server C Preprocessor, Advantage CA-Datcom creates the plan which contains the executable form of the SQL statements on the mainframe just as it would if you were preprocessing the program on the mainframe. When the program is executed, the mainframe plan is executed. Advantage CA-Datcom Server automatically translates data between mainframe and workstation formats when it passes it between the workstation and mainframe.

With Advantage CA-Datcom Server, you can write client-server programs that use C as the language. These programs can be used to execute the business logic and access the mainframe Multi-User Facility.

This approach has a performance advantage over using tools that dynamically prepare and execute SQL statements. ODBC calls, that dynamically prepare and execute SQL statements, include the binding of the SQL statements at execution time. The Advantage CA-Datcom Server C Preprocessor creates static plans before the application executes. Therefore, when the program is executed, no binding overhead is present at all. Advantage CA-Datcom Server SQL programs also take full advantage of the Advantage CA-Datcom SQL Block Transfer feature to minimize communication time.

Using Advantage CA-Datacom Navigational Commands

Support is also provided in the Advantage CA-Datacom Server for the execution of Advantage CA-Datacom/DB navigational commands from Advantage CA-Realia II Workbench COBOL programs, and from other programming languages such as Microsoft C/C++. The DBNTRY entry point is supported. The DBNTRY application programming interface includes a rich set of navigational commands including index-only retrieval, indexed row retrieval, update, insert, and delete commands, and the Compound Boolean Selection set-at-a-time commands (SELFR, SELNR, SELPR). See the *Advantage CA-Datacom/DB Programmer Guide* for details on the commands.

Since the navigational interface to Advantage CA-Datacom Server is provided in the form of Dynamic Link Libraries (DLLs), any workstation programming language that can call DLL functions can use the interface. However, when calling the DLL functions directly, you must provide a “handle” parameter so the DLL code can distinguish the various concurrent connections. The function that takes the extra “handle” parameter is WDBNTRY and its function prototype is in CADCDBN.H.

Unsupported Navigational Commands

All Advantage CA-Datacom/DB navigational commands are supported except:

- LOGDR
- LOGDW
- LOGIT
- LOGLB
- GETIT
- GETPS
- GSETL
- GSETP

Processing of Navigational Commands

When navigational commands are executed, the commands are routed to the mainframe Multi-User Facility and actually execute there. Data is automatically translated between mainframe and PC formats based on the current Datadictionary definitions of elements, keys, and tables. User Requirements Tables can be generated automatically by Advantage CA-Datacom Server, or the Advantage CA-Datacom Server administrator can specify that the server preload and use existing User Requirements Tables. (See the *Advantage CA-Datacom Database Server Option Installation and Maintenance Guide* for details on URTs.)

Using Navigational Commands with C

A sample program written in Microsoft C appears following. This sample program shows how to call various functions. Link the program with CADCMF32.LIB and CADCDB32.LIB.

Note: When you installed Advantage CA-Datcom Server, the workstation installation program placed the .LIB files in the \Program Files\Computer Associates \Advantage CA-Datcom Server 5.0\ subdirectory (if you allowed the directory to default) on your PC.

Sample Program

```
/*
CADCSMP1.C - Sample program to demonstrate using navigational
commands from C with Advantage CA-Datcom Server Release 5.0.

Compile and link as a QuickWin application (Microsoft's
Visual C/C++ 1.0 or later is required for this.) Link in
CADCMF32.LIB and CADCDB32.LIB.

Change the value of szServId below to connect to your server.

*/
#include <ctype.h>
#include <dos.h>
#include <malloc.h>
#include <stdarg.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

#define STRICT
#include "windows.h"
/*
CA Header Files
*/
#include <cadcdrv.h> /* for CADCOMMF.DLL functions */
#include <cadcdbn.h> /* for CADCOMDB.DLL functions */

static HATCH hAttach;
static char sUib[ 33 ] = "RAAT.C Sample Program ";
static char sel[ ] = "FULL \0 ";
static DBREQARA uRa;
typedef struct _CUST
{
char cStatus; /* +000 CHAR, 1 */
char sId[ 4 ]; /* +001 CHAR, 4 */
char sName[ 30 ]; /* +005 CHAR, 30 */
};
```

```

char sCompany[ 30 ];          /* +035 CHAR, 30 */
char sAddress[ 30 ];        /* +065 CHAR, 30 */
char sCity[ 15 ];          /* +095 CHAR, 15 */
char sState[ 2 ];          /* +110 CHAR, 2 */
char sZip[ 9 ];            /* +112 CHAR, 9 */
char cFlag;                /* +121 CHAR, 1 */
char sS1[ 3 ];             /* +122 CHAR, 3 */
char sS2[ 3 ];             /* +125 CHAR, 3 */
char sS3[ 4 ];             /* +128 CHAR, 4 */
char sBalance1[ 5 ];       /* +132 PACKED, 5 */
char sBalance2[ 5 ];       /* +137 PACKED, 5 */
char sS4[ 2 ];             /* +142 CHAR, 2 */
char sS5[ 2 ];             /* +144 CHAR, 2 */
char sS6[ 2 ];             /* +146 CHAR, 2 */
char sS7[ 5 ];             /* +148 CHAR, 5 */
} CUST;
static CUST uCust;

/*
 * Connection variables
 */
static char      szServId[ ] = "TSO4REL3";
typedef struct QCOMCA SQLCA;
static HODBC     hEnv;
static HODBC     hDbc;
static HODBC     hStmt;
static void FAR *hDriver;
static UB1       szSqlState[ 6 ];
static B4        nNativeErrorFlags;
static B2        nErrorMsgMax = SQL_MAX_MESSAGE_LENGTH - 1;
static B2        nErrorMsgSize;
static UB1       szErrorMsg[ SQL_MAX_MESSAGE_LENGTH ];

/*
 * initRa ( ) - macro to initialize the Request Area
 */
#define initRa(puRa) memset(puRa,0x00,sizeof(DBREQARA))

/*
 * disconnectFromServer ( ) - disconnect from a Server.
 */
void disconnectFromServer ( )
{
    SQLDisconnect ( hDbc );
    SQLFreeConnect ( hDbc );
    SQLFreeEnv ( hEnv );
    return;
}

/*
 * connectToServer ( ) - connect to a Server.
 */
int connectToServer ( )
{
    int nRet;

    /*
     * Allocate the environment.
     */
    nRet = SQLAllocEnv ( &hEnv );

```

```
if ( nRet != SQL_SUCCESS )
{
    nErrorMsgSize = nErrorMsgMax;
    SQLError( 0, 0, 0,
              szSqlState, &nNativeErrorFlags, szErrorMsg,
              nErrorMsgMax, &nErrorMsgSize
            );
    printf( "SQLAllocEnv failed (%s-%s)\n", szSqlState,
           szErrorMsg );
    return ( 1 );
}

/*
Allocate a mainframe connection.
*/
nRet = SQLAllocConnect( hEnv, &hDbc );
if ( nRet != SQL_SUCCESS )
{
    nErrorMsgSize = nErrorMsgMax;
    SQLError( hEnv, 0, 0,
              szSqlState, &nNativeErrorFlags, szErrorMsg,
              nErrorMsgMax, &nErrorMsgSize
            );
    printf( "SQLAllocConnect failed (%s-%s)\n", szSqlState, szErrorMsg );
    return ( 2 );
}

/*
Set the connection options. Option 1100 suppress
construction of the SQL access plan.
*/
nRet = SQLSetConnectOption( hDbc, 1100, 1 );
if ( nRet != SQL_SUCCESS && nRet != SQL_SUCCESS_WITH_INFO )
{
    nErrorMsgSize = nErrorMsgMax;
    SQLError( hEnv, hDbc, 0,
              szSqlState, &nNativeErrorFlags, szErrorMsg,
              nErrorMsgMax, &nErrorMsgSize
            );
    printf( "SQLSetConnectOption failed (%s-%s)\n",
           szSqlState, szErrorMsg );
    return ( 3 );
}

/*
Establish the connection. SQLDriverConnect( ) may be
used here if needed.
*/
nRet = SQLConnect( hDbc, ( UBI * )szServId, SQL_NTS,
                  NULL, 0, NULL, 0 );
if ( nRet != SQL_SUCCESS && nRet != SQL_SUCCESS_WITH_INFO )
{
    nErrorMsgSize = nErrorMsgMax;
    SQLError( hEnv, hDbc, 0,
              szSqlState, &nNativeErrorFlags, szErrorMsg,
              nErrorMsgMax, &nErrorMsgSize
            );
    printf( "SQLConnect failed (%s-%s)\n", szSqlState,
           szErrorMsg );
    return ( 4 );
}

/*
Obtain the low-level connection (attachment) handle to
the Server.
```

```

*/
nRet = DBGetHandle( hDbc, &hAttach );
if ( nRet != SQL_SUCCESS )
{
    printf( "DBGetHandle failed\n" );
    return ( 6 );
}

return ( 0 );
}

/*
MAIN() - mainline code for RAAT & CBS test driver.
*/
int main( )
{
    int nRet;

    /*
    Connect to the server.
    */
    printf( "Connecting to %s\n", szServId );
    nRet = connectToServer( );
    if ( nRet != 0 )
        goto quit;

    /*
    Construct the Request Area.
    */
    initRa( &uRa );
    memcpy( uRa.command, "REDKG", 5 );
    memcpy( uRa.tbl_name, "CUS", 3 );
    memcpy( uRa.key_name, "CUSID", 5 );
    uRa.dbid.i = 10;

    /*
    Read the database.
    */
    for ( ;; )
    {
        Wdbntry( hAttach, sUib, &uRa, ( char * )&uCust,
                sEl, NULL );
        if ( memcmp( uRa.ret_code, " ", 2 ) != 0 )
            break;
        printf( "%-4.4s %-30.30s %-30.30s\n",
                "%-15.15s %-2.2s\n\n",
                uCust.sId, uCust.sName, uCust.sCompany,
                uCust.sCity, uCust.sState
                );
        memcpy( uRa.command, "REDNX", 5 );
    }

    /*
    Print out appropriate message for unexpected errors.
    */
    if ( memcmp( uRa.ret_code, "14", 2 ) == 0 )
    {
        printf( "End of table reached\n" );
    }
    else
    {
        printf(
            "Unexpected return code (ret=%-2.2s, rsn=%2.2d)\n",
            uRa.ret_code, uRa.iret_code
        );
    }
}

```

```
    }  
  
    /*  
    Hang up.  
    */  
quit:  
    disconnectFromServer ( );  
    printf( "Disconnected\n" );  
    return ( 0 );  
}
```

Using Navigational Commands with Advantage CA-Realia II Workbench

DBNTRY Programming

Programs using native Advantage CA-Datacom/DB calls must ensure that all programs are compiled with ASCII code set enabled and COMP-5. Compiling can be ensured by using the Advantage CA-Realia II Workbench setup.

Coding Your COBOL Programs

Write your COBOL programs as you would write mainframe COBOL programs that include Advantage CA-Datacom/DB navigational commands. The commands are routed to the mainframe for execution. However, remember that all data is in workstation format, not mainframe format.

- Character data is translated between EBCDIC and ASCII. Integers are byte-swapped.
- NUMERIC (zoned decimal), DECIMAL (packed decimal) and FLOAT values also have different formats on the PC than on the mainframe.
- DATE, TIME, and TIMESTAMP values are the same except for the ASCII-EBCDIC translation.

These differences in format could impact any program statements which assume a certain physical representation of data, such as assuming that the right most byte of a PIC S9(9) COMP field is the low-order byte (on the workstation, it is the high-order byte). Or checking for X'40' when looking for a blank (on the workstation, a blank is X'20'). Programs which were originally written for the mainframe may have to be "ported" to the workstation. During the "port" process, change any platform-specific code into portable code (or, as a last resort, change it to be specific to the workstation instead of the mainframe).

Linking Your COBOL Programs

You must link your navigational COBOL programs with the CADCRW32.LIB, file provided.

You do not have to link a User Requirements Table (URT). URTs are handled entirely on the mainframe by the Advantage CA-Datcom Server program. See the *Advantage CA-Datcom Database Server Option Installation and Maintenance Guide* for the options for URT administration.

Executing Your COBOL Programs

When the first navigational command is executed, connection to the mainframe is automatically accomplished by the CADCRW32 module. By default, it connects to the Default Data Source. However, you can add a call to the DBOPTS program to your program to control the connection. Code the DBOPTS call so that it is executed before the first navigational command is executed. The DBOPTS program is called with one parameter, a control block that looks like the following sample:

```
01 NAV-OPTIONS.
   05 NAV-CONNTYPE      PIC X.
   05 NAV-DSN           PIC X(32) .
   05 NAV-USERID        PIC X(8) .
   05 NAV-PASSWORD      PIC X(20) .
   05 NAV-FILL1         PIC X(52) .
```

Type	Definition
NAV-CONNTYPE	Should be S for Silent connection (no prompting for UserID and Password), P for Prompted, or A for Passed.
NAV-DSN	Is the name of the Data Source.
NAV-USERID	Is the user identification. Used for Passed.
NAV-PASSWORD	Is the user password. Used for Passed.
NAV-FILL1	Should be initialized to LOW VALUES and is reserved for future use.

If there is an error while connecting, your program receives a return code 68, and the internal return code indicates the problem more specifically. See the appendix, "Messages," for a list of possible internal return codes.

If you get a 68(2) or a 68(196) you may get more information by calling the DBERROR program, which takes one parameter, a control block which looks like the following sample:

```
01 ERROR-INFO.  
   05 SQL-STATE          PIC X(5) .  
   05 NATIVE-ERROR      PIC S9(9) BINARY .  
   05 ERROR-MSG         PIC X(132) .  
   05 CONNTYPE          PIC X .  
   05 DSN                PIC X(32) .  
   05 FILL1             PIC X(80) .
```

The information returned corresponds to the ODBC error information documented in the appendix “Messages” and to the information, if any, provided in the previous DBOPTS call.

Advantage CA-Realia II Workbench COBOL programs automatically disconnect from the mainframe when the program terminates.

Compatibility Levels

ODBC API Conformance Levels

The Advantage CA-Datacom/DB ODBC Driver conforms to the Core level of the Microsoft ODBC API specification. It also provides one function from Level 1: SQLBrowseConnect.

JDBC API Conformance Levels

The Advantage CA-Datacom/DB JDBC Driver conforms to the JDBC 1.2 specification

ODBC SQL Conformance Levels

The Advantage CA-Datacom/DB ODBC Driver conforms to the Microsoft Core SQL Grammar specification, with the following exceptions:

Statement	Unsupported Keywords
DROP TABLE	CASCADE, RESTRICT
DROP VIEW	CASCADE, RESTRICT
GRANT	REFERENCES
REVOKE	REFERENCES, RESTRICT
	RESTRICT is implied when CASCADE is not specified.

ODBC Data Type Support

SQL Data Types

Advantage CA-Datcom Server supports all of the Core SQL data types. In addition, the server supports the following Extended SQL data types:

- SQL_DATE
- SQL_TIME
- SQL_TIMESTAMP
- SQL_BINARY
- SQL_VARBINARY
- SQL_LONGVARBINARY

Advantage CA-Datcom Server maps the SQL_BINARY data type to the Advantage CA-Datcom/DB CHAR data type with a subtype of FOR BIT DATA. This prevents ASCII/EBCDIC translation from occurring when Advantage CA-Datcom Server transfers the data between the PC and mainframe. Similarly, Advantage CA-Datcom Server maps the SQL_VARBINARY data type to the Advantage CA-Datcom/DB VARCHAR FOR BIT DATA data type and SQL_LONGVARBINARY to VARCHAR FOR BIT DATA.

The maximum length for a column of data type SQL_LONGVARBINARY is approximately 32,000 bytes. However, the actual maximum depends on the Advantage CA-Datcom/DB data area in which the table resides and the lengths of other columns in the table. See the discussion of LONG VARCHAR in the section on data types in the *Advantage CA-Datcom/DB SQL User Guide* for details.

C Data Types

Advantage CA-Datcom Server supports all of the Core C data types. In addition, the Server supports the following Extended C data types:

- SQL_DATE
- SQL_TIME
- SQL_TIMESTAMP
- SQL_BINARY

Advantage CA-Datcom/DB SQL Preprocessor Options

All the mainframe Advantage CA-Datcom/DB SQL Preprocessor Options are available except:

- SQLMODE=DB2
- SQLMODE=DB2A86

Note: Be sure to specify PLNCLOSE=T when coding the preprocessor options.

Advantage CA-Datcom/DB Navigational Command Support

From C programs and Advantage CA-Realia II Workbench COBOL programs, all Advantage CA-Datcom/DB navigational commands are supported except:

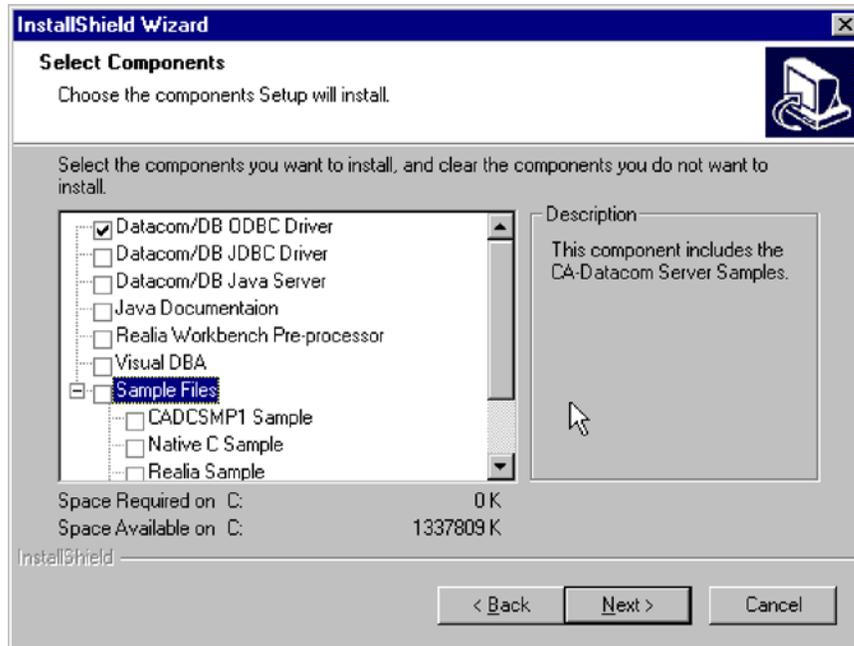
- LOGDR
- LOGDW
- LOGIT
- LOGLB
- GETIT
- GETPS
- GSETL
- GSETP

Workstation Templates

Reproduce the templates of the screens provided in this appendix and use the reproduced pages to specify what you want the workstation user to enter.

InstallShield Wizard

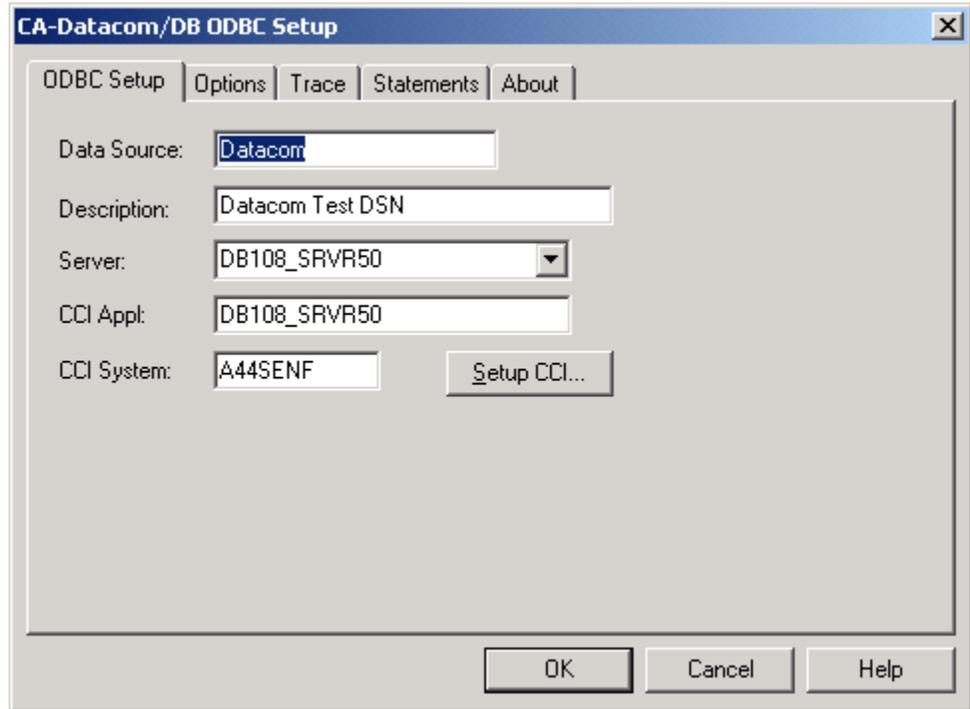
The following screen appears when you select Custom installation or when you run the installation after an initial installation. It tells you which components will be installed, how much space is required for the components you select, and how much space you have.



Additional Instructions:

Advantage CA-Datcom/DB ODBC Setup

Use the entries provided on this template to provide the ODBC Driver Manager with information needed to configure your data source. Do not vary the entries provided unless authorized to do so.



Additional Instructions:

Connection Window Template

CA-Datcom/DB Logon

Logon

User ID:

Password:

New Password:

Verify Password:

Server Name:

System ID:

Appl ID:

OK Cancel

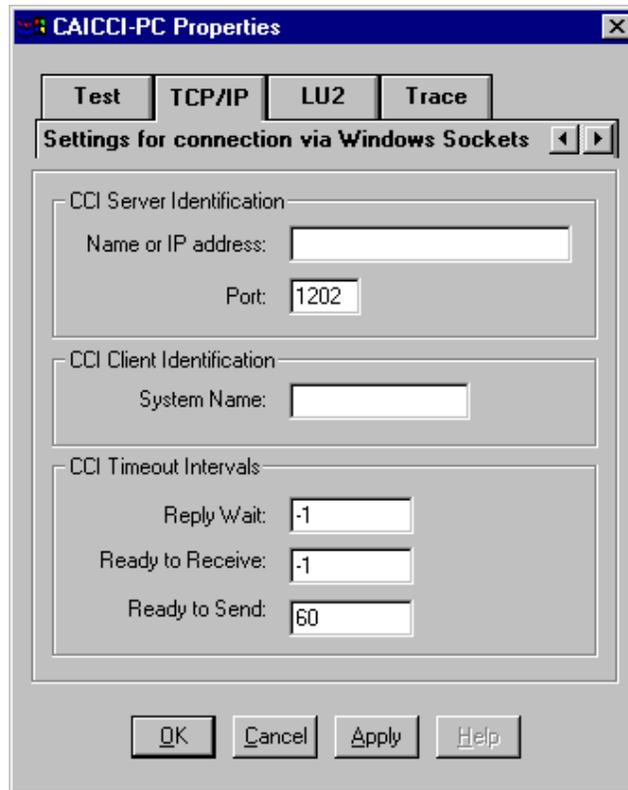
Additional Instructions:

CAICCI Window Template - Test



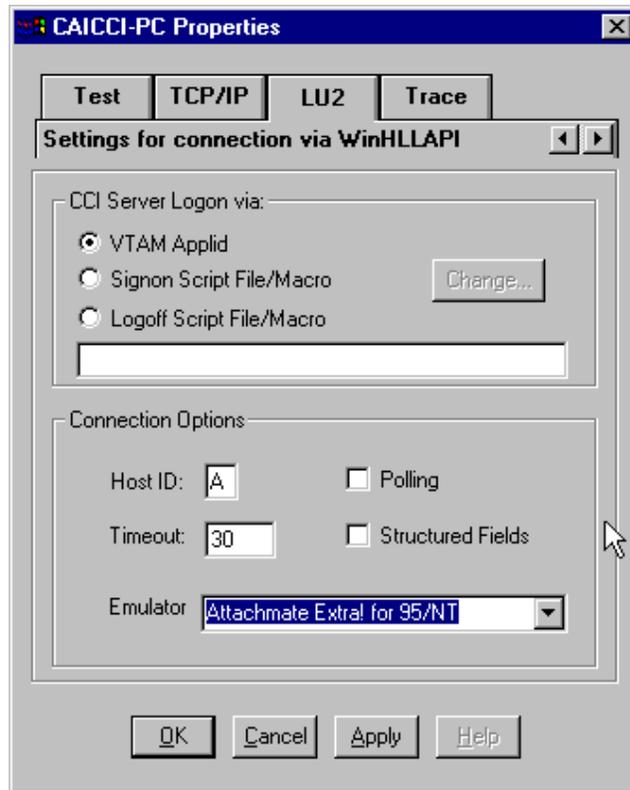
Additional Instructions:

CAICCI Window Template - TCP/IP



Additional Instructions:

CAICCI Window Template - LU2



Additional Instructions:

Workstation Messages

Errors generated by Advantage CA-Datacom Server are reported to the application as message text and a "native error code." The error message and code may or may not be displayed to you by the application. Contact your server administrator if you are not able to diagnose a problem from the information the application provides.

Note: If a message is longer than the maximum length of 110 characters, Advantage CA-Datacom Server truncates the message.

Message Prefixes

Since connecting to a mainframe server involves many layers of software, the ODBC error messages include identifiers that indicate where an error originates. Messages from the Advantage CA-Datacom/DB ODBC driver are prefixed with [CA][ODBC CA-Datacom/DB Driver]. The first part of the prefix indicates the vendor of the ODBC driver, in this case CA for Computer Associates. The next part of the prefix indicates that the Driver code issued the error message.

If the error was generated by the Microsoft ODBC driver Manager, the prefix would have been [Microsoft][ODBC DLL].

Messages from the Advantage CA-Datacom/DB mainframe server or from its workstation-resident interface module, or from the mainframe Multi-User Facility, are prefixed with [CA][CA-Datacom/DB ODBC driver][CA-Datacom /DB]. This prefix indicates that the error was reported by the "data source" itself, not the Driver Manager or the Driver.

Messages Not in This Manual

Messages with negative numbers that are not included in this appendix are SQL codes issued by mainframe Advantage CA-Datacom/DB. For documentation on these codes, see the *Advantage CA-Datacom/DB Message Guide*.

Messages with Positive Message Numbers

1001 Out of memory

Explanation: An ODBC driver function could not allocate memory.

User Response: Make more memory available to the ODBC driver modules and retry the operation.

1002 Invalid function sequence

Explanation: ODBC functions have been called in an illogical order.

User Response: The application must call the ODBC functions according to the rules for function call ordering.

1003 Truncation error

Explanation: The selected value cannot be copied to the provided host variable because truncation of digits to the right of the decimal place would occur.

User Response: The application must provide a larger host variable.

1004 Significant digits would be truncated

Explanation: The selected value cannot be copied to the provided host variable because truncation of significant digits would occur.

User Response: The application must provide a larger host variable.

1005 Invalid string form of a number

Explanation: The selected value cannot be returned in the numeric host variable provided by the application because truncation of significant digits would occur. Or, non-numeric characters are in the value.

User Response: The value cannot be retrieved using a numeric host variable. Correct the data or change the application.

1006 NULL argument

Explanation: A NULL argument has been passed to an ODBC function when a non-NULL argument is required.

User Response: The application must provide a non-null argument.

1007 Truncation error

Explanation: The selected value cannot be copied to the provided host variable because truncation of digits to the right of the decimal place would occur.

User Response: The application must follow the rules for calling ODBC functions.

1008 Cannot prepare without parameter type(s), use SQLSetParam

Explanation: The SQL statement has parameter markers but SQLSetParam has not been called for at least one of them.

User Response: The application must call SQLSetParam for each parameter marker in the SQL statement being prepared.

1010 Table qualifier is not supported

Explanation: The ODBC concept of qualifier is not supported. Advantage CA-Datcom/DB names consist of an authorization ID and a name, separated by a period.

User Response: Do not attempt to provide a qualifier.

3002 Data source missing or too long

Explanation: A data source name is either blank or too long.

User Response: Correct the invalid name and retry the connection.

3003 User ID missing or too long

Explanation: A user ID is either blank or too long.

User Response: Correct the invalid name and retry the connection.

3004 Password missing or too long

Explanation: A password is either blank or too long.

User Response: Correct the invalid password and retry the connection.

3007 Advantage CA-Datacom/DB error during QEXEI

Explanation: An Advantage CA-Datacom/DB error occurred during a QEXEI command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

3008 Advantage CA-Datacom/DB error during QPRPD

Explanation: An Advantage CA-Datacom/DB error occurred during a QPRPD command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

3009 Advantage CA-Datacom/DB error during QEXEC

Explanation: An Advantage CA-Datacom/DB error occurred during a QEXEC command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

3010 Cannot disconnect--transaction in progress

Explanation: You have attempted to disconnect from a data source before all transactions have completed.

User Response: Wait until all transactions are complete before attempting to disconnect.

3012 Advantage CA-Datcom/DB error during OPENP

Explanation: An Advantage CA-Datcom/DB error occurred during an OPENP command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datcom/DB SQLCODE is the "native error code," which can be displayed by your application.

3013 Advantage CA-Datcom/DB error during CLOSE

Explanation: An Advantage CA-Datcom/DB error occurred during a CLOSE command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datcom/DB SQLCODE is the "native error code," which can be displayed by your application.

3014 Attempting to disconnect before connecting

Explanation: You are attempting to disconnect before you have connected to a data source.

User Response: None. This message is for information only.

3015 Driver is not capable

Explanation: The ODBC driver cannot perform the operation you requested.

User Response: Do not attempt this operation. If you get this error while executing an application, contact your server administrator.

3021 SysID and Applid required for connection

Explanation: Both the CCI System and Application IDs are required for the ODBC driver to connect to a mainframe server.

User Response: Specify both names.

3022 Advantage CA-Datcom/DB error during WServerList

Explanation: An Advantage CA-Datcom/DB error occurred during a WGetServerlist command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datcom/DB SQLCODE is the "native error code," which can be displayed by your application.

4001 Advantage CA-Datcom/DB error during QHOST

Explanation: An Advantage CA-Datcom/DB error occurred during a QHOST command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datcom/DB SQLCODE is the "native error code," which can be displayed by your application.

4003 Invalid cursor name

Explanation: An invalid cursor name has been specified.

User Response: Verify that the name conforms to SQL rules.

4004 Invalid data type

Explanation: An invalid data type has been detected in an ODBC function.

User Response: This is an internal system error. Contact CA Technical Support.

4005 SQLExecute called before SQLPrepare

Explanation: The SQLExecute function has been called before SQLPrepare has been called.

User Response: The application must call SQLPrepare before SQLExecute.

4006 Conversion not supported

Explanation: A data type conversion was requested that is not supported.

User Response: The application must use only those data type conversions supported by the Advantage CA-Datacom/DB ODBC driver.

4007 Invalid SQL statement text

Explanation: You have specified an empty string as an SQL statement text.

User Response: SQL statements must not be the empty string.

4009 Invalid column number

Explanation: A column number that was specified as a parameter for an ODBC function is not valid for the SQL statement.

User Response: This is an application error. Contact your server administrator.

4011 Advantage CA-Datacom/DB error during QPRPD

Explanation: An Advantage CA-Datacom/DB error occurred during a QPRPD command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

4016 Advantage CA-Datacom/DB error during QEXEC

Explanation: An Advantage CA-Datacom/DB error occurred during a QEXEC command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

4020 Invalid SQL syntax

Explanation: Invalid SQL syntax has been detected in an SQL statement sent to Advantage CA-Datcom/DB.

User Response: Recode the SQL statement to follow Advantage CA-Datcom/DB syntax and retry.

4021 Column type invalid or not supported

Explanation: An invalid data type has been detected in an ODBC function.

User Response: The application must use only those data types supported by the Advantage CA-Datcom/DB ODBC driver.

4022 Cannot use SQLGetData on bound column

Explanation: The SQLGetData function was passed to the identifier of a column for which SQLBindColumn has been called.

User Response: The application must not use SQLGetData on columns which have been bound to an address.

4028 Parameter truncation error

Explanation: A parameter would have to be truncated and is not allowed.

User Response: This is an internal system error. Contact CA Technical Support.

4029 Advantage CA-Datcom/DB error during WGetTables

Explanation: An Advantage CA-Datcom/DB error occurred during a WGetTables command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datcom/DB SQLCODE is the "native error code," which can be displayed by your application.

4030 Invalid Table type

Explanation: A parameter to the SQLTables function is invalid.

User Response: The application must use either 'TABLE,' 'VIEW,' 'SYSTEM TABLE,' 'SYNONYM,' or 'ALIAS' as the table type.

4031 Advantage CA-Datacom/DB error during WGetColumns

Explanation: An Advantage CA-Datacom/DB error occurred during a WGetColumns command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

4032 Advantage CA-Datacom/DB error during WGetSpecialColumns

Explanation: An Advantage CA-Datacom/DB error occurred during a WGetSpecialColumns command.

User Response: Contact your server administrator to diagnose the problem. The Advantage CA-Datacom/DB SQLCODE is the "native error code," which can be displayed by your application.

4035 Missing table name

Explanation: The table name is missing on an ODBC function which requires one (for example, SQLGetColumns).

User Response: Specify a table name.

4037 Invalid or unsupported Scope option

Explanation: An invalid value was specified for the Scope parameter for the SQLSpecialColumns function.

User Response: Only SQL_SCOPE_CURROW is supported.

4038 Driver is not capable

Explanation: The ODBC driver cannot perform the operation you requested.

User Response: Do not attempt that operation. If you get this error while executing an application, contact your server administrator.

4050 Read only Server, no updates allowed

Explanation: You are using a read-only server and attempted to issue an update transaction.

User Response: None required.

Messages with Negative Message Numbers

Messages with negative numbers which are not included in this appendix are SQL codes issued by mainframe Advantage CA-Datcom/DB. For documentation on these codes, see the *Advantage CA-Datcom/DB Message Guide*.

-5001 NULL HANDLE

Explanation: A NULL HANDLE pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5002 NULL SQLWA

Explanation: A NULL SQLWA pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5003 NULL SQLCA

Explanation: A NULL SQLCA pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5004 INVALID QCMD

Explanation: An invalid "Q" command was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5005 CANNOT ALLOCATE ATTACH

Explanation: A WDBSQLE function could not allocate memory.

User Response: Make more memory available to the ODBC driver modules and retry the operation.

-5006 INVALID HANDLE

Explanation: An invalid HANDLE pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5007 NO MEMORY AVAILABLE

Explanation: A WDBSQLE function could not allocate memory.

User Response: Make more memory available to the ODBC driver modules and retry the operation.

-5009 DATA VARIABLE NOT SUPPLIED

Explanation: A NULL host variable pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5010 INVALID DATA TYPE

Explanation: An invalid data type code was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5011 INDICATOR VARIABLE NOT SUPPLIED

Explanation: A NULL indicator variable pointer was passed to a WDBSQLE function when an indicator variable pointer value was required.

User Response: This is an internal system error. Contact CA Technical Support.

-5012 INVALID PLAN OPTIONS STRUCTURE

Explanation: An invalid plan option structure was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5013 SQLDA REQUIRED

Explanation: A NULL SQLDA pointer was passed to a WDBSQLE function when an SQLDA was required.

User Response: This is an internal system error. Contact CA Technical Support.

-5014 INVALID TRANSLATION ACTION

Explanation: An invalid translation action was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5015 CCIINIT FAILURE: xxxxxx

Explanation: A CAICCI error has been encountered while trying to initialize a connection. The specific CCI error message is reported in xxxxxx.

User Response: You may need to exit Windows to clear the error condition.

-5016 CCICONV FAILURE: xxxxxx

Explanation: A CAICCI error has been encountered while trying to communicate with the mainframe. The specific CCI error message is reported in xxxxxx.

User Response: A "time-out" error indicates that the mainframe took longer to respond than expected. You may need to adjust the CCI time-out value. You may have to exit Windows to clear the error condition.

-5018 NULL SQLDA

Explanation: A NULL SQLDA pointer was passed to a WDBSQLE catalog function.

User Response: This is an internal system error. Contact CA Technical Support.

-5020 MISSING SYSTEM ID ON CONNECT

Explanation: A NULL CCI System ID pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5021 NULL STATEMENT

Explanation: A NULL STATEMENT pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5022 MSG PART OVERFLOW

Explanation: An invalid message length was returned from the mainframe.

User Response: This is an internal system error. Contact CA Technical Support.

-5023 MISSING APPL ID ON CONNECT

Explanation: A NULL CCI Application ID pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5025 NULL SERVER LIST

Explanation: A NULL Server List pointer was passed to a WDBSQLE function.

User Response: This is an internal system error. Contact CA Technical Support.

-5026 INVALID CCI VERSION

Explanation: The version of CAICCI that you are using on the workstation is older than the required version and is incompatible.

User Response: Make sure the CAICCI (LCCIWIN.DLL in the WINDOWS directory) version installed is the same version as the one provided with the ODBC driver. See your mainframe server administrator for assistance. There should not be an LCCIWIN.DLL in the directory that contains your application. If there is, it is loaded instead of the one in the WINDOWS directory.

-5028 NOT CONNECTED

Explanation: The CAICCI connection failed.

User Response: Retry the application. If this error persists, contact CA Technical Support.

-5072 MAINFRAME SERVER ERROR: NO MEMORY AVAILABLE

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5073 MAINFRAME SERVER ERROR: INVALID REQUEST TYPE

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5074 MAINFRAME SERVER ERROR: NO CONNECTION HANDLE

Explanation: The mainframe server has released the connect to the PC. This is caused by exceeding the TIMEOUT= parameter value on the server startup.

User Response: Reconnect to the server.

-5075 MAINFRAME SERVER ERROR: UNABLE TO LOAD DBINFPF

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5076 MAINFRAME SERVER ERROR: UNRECOVERABLE CCI ERROR

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5078 MAINFRAME SERVER ERROR: BAD STMT ID ON CATALOG FUNCTION

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5079 MAINFRAME SERVER ERROR: BAD SQLDA ON CATALOG FUNCTION

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

-5080 MAINFRAME SERVER ERROR: REQUESTED SERVER NOT AVAILABLE

Explanation: The Server you selected is not found in the server group of the server identified by the CCI System and Application IDs you provided.

User Response: Contact your server administrator to verify the server name.

-5081 MAINFRAME SERVER ERROR: SIGN-ON FAILURE

Explanation: The server you are attempting to log on to has rejected your request.

User Response: Verify that the user ID and password you are using are correct. Contact your mainframe server administrator for assistance.

-5082 MAINFRAME SERVER ERROR: SERVER LIST BUFFER TOO SMALL

Explanation: The mainframe server has reported an internal error.

User Response: This is an internal system error. Contact CA Technical Support.

Mainframe Messages

Errors generated by the mainframe component of Advantage CA-Datcom Server are reported to the system console. Each message's number ends with a severity code :

- E** Error.
- I** Information.
- L** Literal (provided to facilitate translation). Literal messages can appear in Advantage CA-Datcom Server output without a preceding error message number.

DSV00001E Unable to load module SVCATPR. Job ended.

Explanation: Module SVCATPR cannot be loaded. You may have coded the startup JCL incorrectly.

User Response: Check the batch job, the input parameter requirements. When an error is found, make the necessary correction and resubmit the job.

DSV00002E Error reading system input file. Job ended.

Explanation: Advantage CA-Datcom Server has encountered an error while attempting to read the system input (SYSIN on MVS, SYSIPT on VSE) file. This is a system problem.

User Response: Recreate the system input file.

DSV00003E Invalid SERVERNAME on control statement. Job ended.

Explanation: The SERVERNAME control statement contains an invalid entry. The name may have been coded using characters that do not conform to SQL naming conventions.

User Response: Examine the entry for errors, make corrections, and resubmit the job.

DSV00004E Invalid APPLID on control statement. Job ended.

Explanation: The APPLID in the startup job is not known or contains invalid characters.

User Response: Correct the APPLID and resubmit the job.

DSV00005I Invalid AUTHID on control statement. SYSUSR used.

Explanation: The AUTHID in the startup job is not a known AUTHID or was given in an invalid format. The system SQL AUTHID was used instead.

User Response: Examine the control statement for errors.

DSV00006I Invalid SQLMODE on control statement. DATACOM used.

Explanation: The SQLMODE entry was invalid and could not be used.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the SQLMODE entry, and resubmit the job. Server does not support SQLMODE=DB2 and SQLMODE=DB2A86.

DSV00007I Invalid CBSIO on control statement. Zero used.

Explanation: The SQL CBSIO entry was invalid and could not be used.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the CBSIO entry, and resubmit the job.

DSV00008I Invalid PRTY on control statement. Seven (7) used.

Explanation: The SQL PRTY entry was invalid and could not be used.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the PRTY entry, and resubmit the job.

DSV00009I Invalid WORKSPACE on control statement. 16 used.

Explanation: The SQL WORKSPACE entry was invalid and could not be used.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the WORKSPACE entry, and resubmit the job.

DSV00010I Invalid TIMEMIN on control statement. Zero used.

Explanation: The SQL TIMEMIN entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the TIMEMIN entry, and resubmit the job.

DSV00011I Invalid TIMESEC on control statement. Zero used.

Explanation: The SQL TIMESEC entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the TIMESEC entry, and resubmit the job.

DSV00012I Invalid PLNCLOSE on control statement. "T" used.

Explanation: The SQL PLNCLOSE entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the PLNCLOSE entry, and resubmit the job.

DSV00013I Invalid ISOLEVEL on control statement. "C" used.

Explanation: The SQL ISOLEVEL entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the ISOLEVEL entry, and resubmit the job.

DSV00014I Invalid MSG on control statement. "NN" used.

Explanation: The SQL MSG entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the SQLMSG entry, and resubmit the job.

DSV00015I Invalid OPT on control statement. "P" used.

Explanation: The SQL OPT entry was invalid and could not be used.

User Response: See the *Advantage CA-Datacom/DB SQL User Guide* for valid SQL plan options, correct the SQLOPT entry, and resubmit the job.

DSV00016I Invalid DATE on control statement. "ISO" used.

Explanation: The SQL DATE entry was invalid and could not be used. Server used the default. DATE=ISO.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the DATE entry, and resubmit the job.

DSV00017I Invalid TIME on control statement. "JIS" used.

Explanation: The SQL TIME entry was invalid and could not be used. Server used the default. DATE=JIS.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the TIME entry, and resubmit the job.

DSV00018E Invalid PLANNAME on control statement. Job ended.

Explanation: The SQL plan name named on the startup job PLANNAME control statement is not a valid name. This must be a four-character prefix to be used for all plan names generated by this server.

User Response: See the *Advantage CA-Datcom/DB SQL User Guide* for valid SQL plan options, correct the PLANNAME entry, and resubmit the job.

DSV00019E Invalid DBUSERS on control statement. Job ended.

Explanation: The number of Advantage CA-Datcom/DB threads you entered on the startup job DBUSERS control statement is not a valid number. The job was canceled.

User Response: Valid numbers are between 1 and 9999. Change the statement and resubmit the job.

DSV00020I Invalid DECPOINT on control statement. Period used.

Explanation: The parameter value is invalid. The default was used.

User Response: See *Advantage CA-Datcom/DB SQL User Guide* for valid options. Make corrections and resubmit the job.

DSV00021I Invalid STRDELIM on control statement. Apostrophe used.

Explanation: The parameter value is invalid. The default was used.

User Response: See *Advantage CA-Datacom/DB SQL User Guide* for valid options. Make corrections and resubmit the job.

DSV00022I Invalid TIMEOUT on control statement. 120 minutes used.

Explanation: The length of time a workstation can remain idle without a time out is invalid. Valid entries are 1 to 1440.

User Response: Examine the entry for errors, make corrections, and resubmit the job.

DSV00023E Unable to load SVMSGPR module. Initialization ended.

Explanation: The module could not be loaded, probably because invalid load libraries were specified.

User Response: Check the STEPLIB or LIBDEF statement for correct load libraries.

DSV00024IEUnable to load DBINFPR module. Initialization ended.

Explanation: The module could not be loaded, probably because invalid load libraries were specified.

User Response: Check the STEPLIB or LIBDEF statement for correct load libraries.

DSV00025E Unable to obtain Server storage. Job ended.

Explanation: The function you requested failed because there is insufficient storage in the batch region.

User Response: Increase the region size if possible. If unsuccessful, try to run the job at a time when there is less demand on the system.

DSV00026E Database open failure. Server initialization ended.

Explanation: Unable to open a dynamic User Requirements Table for the DDD table and SQL access.

User Response: If the DBID for the DDD-DATABASE database is not 015, check that you have an input control statement DDDID=xxx specifying the correct DBID.

DSV00027I Invalid DDDID on control statement. Default 015 used.

Explanation: The DDDID= control statement specifies an invalid DBID. Server is using 015 as the DDD-DATABASE database ID.

User Response: Correct your input control statement DDDID=xxx to specify the correct DBID for the DDD-DATABASE database.

DSV00028E Server ended, CCI failure in function: xxxxxxxx.

Explanation: A problem has been experienced in CAICCI, the facility that links a workstation to a server. The message specifies which function failed. The Server job has ended.

User Response: Check the subsequent message for return codes and check error text for the specific problem or problems. Consult the CAICCI documentation for the meaning of the return codes.

DSV00029E Server ending because Multi-User not available.

Explanation: The Multi-User Facility specified in the startup JCL and through which the Server is communicating has abended or is not available.

User Response: If submitting JCL to bring up the server, examine the STEPLIB or LIBDEF statements to ensure that there are no errors. If the message is received after the server has been successfully initiated, then the Multi-User Facility has abended. Restart the server when the Multi-User Facility is again available.

DSV00030E Support for DATACOM Server not requested in Multi-User startup options.

Explanation: Multi-User Startup Options must include the statement DATACOM DTCMSRVR.

User Response: Resubmit the job when Multi-User Facility has been started with the correct options.

DSV00031E DATACOM Server incompatible with current version of database.

Explanation: The current version of the Advantage CA-Datcom Server cannot be used with the version of the Multi-User Facility.

User Response: Database must be at Version 9.0 or higher for the server to run.

DSV00032E Invalid LOGON on control statement. YES used.

Explanation: The parameter was invalid. The default was used.

User Response: Valid entries are NO and YES. Correct the statement and resubmit the job.

DSV00033I Cancel requested for a workstation connection.

Explanation: A request to cancel a connection was received from the communication utility. The task number follows this message.

User Response: No action is required.

DSV00034I The requested connection could not be canceled.

Explanation: The task number named a connection that did not exist or which was actively processing and could not be canceled.

User Response: Review the status report from the communications utility to verify the connection's task number and the status of the connection. Retry if necessary.

DSV00035I Workstation connected to Server.

Explanation: A workstation was successfully connected to this server. Task number, user ID, date, and time follow this message.

User Response: None. This message is for information only.

DSV00042I Invalid SQL option on control statement. Default YES used.

Explanation: The SQL= control statement specifies an invalid value. The valid values are NO or YES. Server is using YES as the option.

User Response: Correct your input control statement to specify a valid value.

DSV00043E Invalid MAXURTS on control statement. Job ended.

Explanation: The MAXURTS= control statement specifies an invalid value. Valid values are 1 through 999.

User Response: Correct your input control statement to specify a valid value.

DSV00044I Invalid URTPRTY on control statement. Seven (7) used.

Explanation: The URTPRTY= control statement specifies an invalid User Requirements Table priority. Valid priority is an integer between 1 and 15, inclusive. Server is using a priority of 7.

User Response: Correct your input control statement to specify a valid value.

DSV00045I Invalid DYNURT on control statement. Default NO used.

Explanation: The DYNURT= control statement specifies an invalid value. Valid values are YES or NO. Server is using DYNURT=NO.

User Response: Correct your input control statement DYNURT= to specify a valid value.

DSV00046E Invalid LOADURT control statement. Job ended.

Explanation: The LOADURT= control statement syntax is invalid.

User Response: Correct your input control statement LOADURT= and resubmit.

DSV00047E Unable to load requested URT from control statement. Job ended.

Explanation: The User Requirements Table specified on LOADURT= control statement is invalid or does not exist.

User Response: Correct the program module name specified or the concatenated library list and resubmit.

DSV00048E URTS requested on LOADURT exceed MAXURTS value. Job ended.

Explanation: The number of User Requirements Tables to load specified by the LOADURT control statement was larger than the value specified in the MAXURTS control statement.

User Response: Correct your input control statement MAXURTS or LOADURT to reflect consistent input.

DSV00049I Advantage CA-Datcom Server INITIALIZED - Server-name.

Explanation: The named server has been successfully started.

User Response: None. This message is for information only.

DSV00050E Unable to load module SVURTPR. Server ended.

Explanation: Advantage CA-Datcom Server could not load the module SVURTPR.

User Response: Verify that the concatenated library list is correct. If it is not correct, correct it and resubmit. If it is correct, contact CA Technical Support.

DSV00051E Request failed. Unable to obtain storage.

Explanation: Advantage CA-Datcom Server could not obtain the required storage.

User Response: Increase the available storage and resubmit.

DSV00052I nnnnnnnn Successfully loaded from library.

Explanation: Advantage CA-Datcom Server successfully loaded User Requirements Table nnnnnnnn.

User Response: None. This message is for information only.

DSV00053E nnnnnnnn Could not be loaded from library.

Explanation: Advantage CA-Datcom Server tried to load User Requirements Table nnnnnnnn and could not.

User Response: Correct the program module name specified or the concatenated library list and resubmit.

DSV00054I nnnnnnnn Enabled; will be opened on first database request.

Explanation: Advantage CA-Datcom Server successfully enabled User Requirements Table nnnnnnnn. Server will open it when server needs it to satisfy a workstation's request.

User Response: None. This message is for information only.

DSV00055I nnnnnnnn Closed; will not be automatically reopened.

Explanation: Advantage CA-Datcom Server successfully closed User Requirements Table nnnnnnnn, as requested through SVCOMPR. Server will not open it again until it receives a SVCOMPR OPENURT= command.

User Response: None. This message is for information only.

DSV00056E nnnnnnnn Could not be closed. CLOSE failed.

Explanation: The Advantage CA-Datcom Server communication utility (SVCOMPR) could not close User Requirements Table nnnnnnnn.

User Response: Contact your database administrator.

DSV00057E Communications utility unable to get storage. Job ended.

Explanation: The Advantage CA-Datcom Server communication utility (SVCOMPR) could not obtain the required storage.

User Response: Increase the available storage and resubmit.

DSV00058E Communications utility has no input to process. Job ended.

Explanation: You submitted a job to the communication utility (SVCOMPR) of Advantage CA-Datcom Server that did not include any input statements.

User Response: Correct the job and resubmit.

DSV00059E Communications utility received invalid APPLID= name. Job ended.

Explanation: You submitted a job to the communication utility (SVCOMPR) of Advantage CA-Datcom Server that specified an invalid APPLID name. The APPLID= parameter should specify the APPLID of the server with which you want to communicate.

User Response: Correct the job and resubmit.

DSV00060I Communications utility input command received: cccccccc

Explanation: You submitted a job to the communication utility of Advantage CA-Datcom Server that included the command cccccccc.

User Response: None. This message is for information only

DSV00061E Communications utility received CCI failure in function: ffffffff

Explanation: CAICCI failed when the communication utility (SVCOMPR) or Advantage CA-Datcom Server tried to perform function ffffffff.

User Response: Correct the problem with CAICCI and resubmit.

DSV00062I Communications utility CCI system ID: ssssssss

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datcom Server is communicating with a CAICCI that has a system ID of ssssssss.

User Response: None. This message is for information only.

DSV00063E Communications utility received invalid input request. Not processed.

Explanation: You submitted a job to the communication utility (SVCOMPR) of Advantage CA-Datcom Server that contains an invalid command.

User Response: Correct the job and resubmit.

DSV00064E Communications utility has no CCI APPLID to communicate with Server.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datcom Server does not include a statement specifying the APPLID for CAICCI.

User Response: Correct the job and resubmit.

DSV00065I Communications utility is sending request to APPLID:

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datcom Server has passed the request to the APPLID for CAICCI.

User Response: None. This message is for information only.

DSV00066E Server could not process request. No memory was available.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datcom Server does not have adequate memory available to process the request.

User Response: Increase the memory allocated and resubmit the request.

DSV00067I FOLLOWING REPORT RETURNED FROM SERVER:

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has received the following information from the Server.

User Response: None. This message is for information only.

DSV00068I END OF REPORT

Explanation: All data received from the server has been reported..

User Response: None. This message is for information only.

DSV00069I Request sent to Server. No response expected.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has passed the request to server and the request will not generate a responding message from Server.

User Response: None. This message is for information only.

DSV00070I Communications utility reached the end of input file.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has read all of the statements in the input for this job.

User Response: None. This message is for information only.

DSV00071E Invalid option received on MODIFY card. Not processed.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has received an invalid option on a MODIFY statement in the input for this job.

User Response: Correct the job and resubmit.

DSV00072E LOADURT= option is specified incorrectly. Not processed.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has received a LOADURT= statement which is specified incorrectly in the input for this job.

User Response: Correct the job and resubmit.

DSV00073E OPENURT= or CLOSEURT= option is specified incorrectly. Not processed.

Explanation: The communication utility (SVCOMPR) of Advantage CA-Datacom Server has received an OPENURT= or CLOSEURT= statement which is specified incorrectly in the input for this job.

User Response: Correct the job and resubmit.

DSV00074L Advantage CA-Datacom Server COMMUNICATIONS UTILITY MESSAGES

Explanation: This literal precedes messages from the communication utility (SVCOMPR) of Advantage CA-Datacom Server.

User Response: None. This message is for information only.

DSV00075L Current Date:

Explanation: This literal precedes the current date on reports from the communication utility (SVCOMPR) of Advantage CA-Datacom Server.

User Response: None. This message is for information only.

DSV00076L Current time:

Explanation: This literal precedes the current time on reports from the communication utility (SVCOMPR) of Advantage CA-Datacom Server.

User Response: None. This message is for information only.

DSV00077L Advantage CA-Datacom Server PROGRAM MESSAGES

Explanation: This literal precedes messages from Advantage CA-Datacom Server.

User Response: None. This message is for information only.

DSV00078L WAITING FOR FREE DB THREAD

Explanation: This literal is used in the STATUS report when Advantage CA-Datacom Server is waiting for a free Advantage CA-Datacom/DB thread to process a request for this task.

User Response: None. This message is for information only.

DSV00079L PROCESSING REQUEST

Explanation: This literal is used in the STATUS report when Advantage CA-Datcom Server is processing a request for this task.

User Response: None. This message is for information only.

DSV00080L WAITING REQUEST-HOLDING THREAD

Explanation: This literal is used in the STATUS report when Advantage CA-Datcom Server is waiting for a request from this task and holding an Advantage CA-Datcom/DB thread to process it.

User Response: None. This message is for information only.

DSV00081L WAITING REQUEST-NO THREAD HELD

Explanation: This literal is used in the STATUS report when Advantage CA-Datcom Server is waiting for a request from this task and is not holding an Advantage CA-Datcom/DB thread.

User Response: None. This message is for information only.

DSV00082L CONNECTIONS TO SERVER

Explanation: This literal precedes a list of the workstations connected to the server.

User Response: None. This message is for information only.

DSV00083L SERVER GROUP

Explanation: This literal precedes a list of servers in the same server group.

User Response: None. This message is for information only.

DSV00084L NO TASKS ALLOCATED

Explanation: This literal specifies that no workstations are connected to this Server.

User Response: None. This message is for information only.

DSV00085L OPEN

Explanation: This literal is used in the URTSTATUS report when the named User Requirements Table has been opened by the server.

User Response: None. This message is for information only.

DSV00086L OPEN-CANNOT BE CLOSED

Explanation: This literal is used in the URTSTATUS report when the named User Requirements Table has been opened by the Server and cannot be closed. This literal will only appear with a User Requirements Table opened by the server to access the Advantage CA-Datacom/DB Datadictionary.

User Response: None. This message is for information only.

DSV00087L CLOSED-OPENED ON REQUEST

Explanation: This literal is used in the URTSTATUS report when the named User Requirements Table has been closed by the server and will be re-opened when the Server receives a request requiring it.

User Response: None. This message is for information only.

DSV00088L CLOSED-NOT OPENED ON REQUEST

Explanation: This literal is used in the URTSTATUS report when the named User Requirements Table has been closed by the Server and will not be reopened until the server receives a command through the communications utility (SVCOMPR) to reopen it.

User Response: None. This message is for information only.

DSV00089I Advantage CA-Datacom Server ENDING - nnnnnnnn

Explanation: Server nnnnnnnn is shutting down. nnnnnnnn is the logical server name for this server.

User Response: None. This message is for information only.

DSV00090E Invalid parameter on control statement. Job ended.

Explanation: Server did not recognize an optional startup parameter. This is usually due to misspelling the parameter.

User Response: Fix the control statement and resubmit the startup JCL.

DSV00091E Invalid ACCESS on control statement. Default W used.

Explanation: Server did not recognize the ACCESS parameter in the control statement and selected the default value, W (read/write access).

User Response: Fix the control statement and resubmit the startup JCL.

DSV00092E Duplicate SERVERNAME entered - servername

Explanation: A server with the name you specified in the SERVERNAME= control statement was already up and running.

User Response: Change your input control statement to specify the correct server name.

DSV00093I (msg) TIMESTAMP:

Explanation: Tracetime active.

User Response: None. This message is for information only.

DSV00094E Invalid SECEXIT= on control statement. Job ended.

Explanation: Server was unable to load the security exit program which you specified on the SECEXIT= input statement.

User Response: Correct the program module name specified on the concatenated library list and resubmit.

DSV00095E Unable to load the Security exit named in input. Job ended.

Explanation: Server was unable to load the security exit program which you specified on the SECEXIT= input statement.

User Response: Correct the program module name specified on the concatenated library list and resubmit.

DSV00096E CONNECT FAILED - Version Incompatibility

Explanation: The mainframe server region has detected an attempted logon from a workstation executing incompatible code. For example, a user at a workstation using Version 2.0 code has attempted to logon to a Version 3.0 mainframe region. The receive and feedback buffers are dumped following this message.

User Response: Install Advantage CA-Datcom Server Version 3.0 on the workstation.

Advantage CA-Datcom/DB Return Codes

The return codes documented following are those return codes **unique** to the client-server environment. All other return codes are documented in the *Advantage CA-Datcom/DB Message Guide*.

Return Code	Internal Return Code	Description
13	001	<p>Explanation: During the execution of the caller's database command, the client interface requested a data translation descriptor from the server. The request failed.</p> <p>User Response: Run a PXX report to obtain detailed information about the failure.</p>
37	001	<p>Explanation: A connection to the server has not been properly established for the caller. (An unknown attachment handle was passed to Wdbntry(.))</p> <p>User Response: Before issuing any database commands, make sure that an ODBC session has been properly established and that the correct handle was obtained from ODBC.</p>
	002	<p>Explanation: The Work Area where the results of the database command were supposed to be stored is not large enough to hold the result.</p> <p>User Response: Increase the size of the Work Area.</p>
68	002	<p>Explanation: An attempt to establish a connection to the server failed.</p> <p>User Response: Calling WGetErrInfo() will provide a detailed error message.</p>
	194	<p>(BAD_TRANS_ACTION)</p> <p>Explanation: An error occurred in data translation.</p> <p>User Response: Examine the data being sent to, or received from, the server and correct it.</p>
	196	<p>(CCI_CONV_PROBLEM)</p> <p>Explanation: An error occurred in conversing with the server.</p> <p>User Response: Calling WGetErrInfo() will obtain a detailed error message.</p>

Return Code	Internal Return Code	Description
68	202	(MSG_PART_OVERFLOW) Explanation: When receiving a reply from the server, the client interface received a message that contained a component longer than expected. User Response: Contact CA Technical Support.
	207	(BAD_LENGTH) Explanation: The total length of the message to be sent to the server exceeded 32767 bytes. User Response: Modify the request to reduce the number of bytes transmitted from the client to the server.
	209	(FLOAT_ERROR) Explanation: An error occurred during the translation of floating-point data. User Response: Examine the data being sent to, or received from, the Server and correct it.
	222	Explanation: The server was unable to obtain memory to fulfill the request. User Response: Increase the size of the partition/address space used by the server.
	223	Explanation: The request type received by the server was invalid. This could be caused by attempting to use navigational commands with a Verison 2.0 server. User Response: Contact CA Technical Support.
	224	Explanation: The attachment handle received by the server was unknown. User Response: Contact CA Technical Support.
	226	Explanation: The server encountered a communications (CAICCI) error. User Response: Contact your server administrator.
	233	Explanation: The server could not load the RAAT/CBS command processor (SVNATPR). User Response: Contact your server administrator.
	234	Explanation: The server could not load the Metadata command processor (SVMDTPR). User Response: Contact your server administrator.
	235	Explanation: The server could not load the URT processor (SVURTPR). User Response: Contact your server administrator.

Return Code	Internal Return Code	Description
89	002	<p>Explanation: During the execution of a database command, the client interface software attempted to allocate memory. The request failed.</p> <p>User Response: Increase the amount of memory available to the client interface software by closing unused applications (Windows). If the problem persists, contact CA Technical Support.</p>

Index

A

Administrator tasks, 5-1
API conformance levels, A-1
APPLID
 Appl ID, 5-17
 JDBC Setup Appl, 6-2
 ODBC Setup Appl, 5-4
APPLID (SVCOMPR command), 4-4
ASCII to EBCDIC translation, 8-2
Authid (ODBC Setup window), 5-10
Auto Commit, 5-6
Automatic rollback and disconnect, 2-9

B

BASEID
 DDBID= in startup JCL, 2-6
 DDDID= in startup JCL, 2-6

C

C programs, 10-1
CA-Datacom/DB
 navigational commands, 10-1, A-3
 threads, 2-5
CADCDB32.LIB, 10-2

CADCDBN.H, 10-1
CADCMF32.LIB, 10-2
Cadrw32.lib, 9-5
CAICCI
 Application ID
 Appl - JDBC setup, 6-2
 Appl - ODBC setup, 5-4
 startup JCL, 2-5
 IDs, 5-1
 System ID
 Connection Window, 5-17
 JDBC setup System, 6-2
 ODBC setup System, 5-4
 workstation requirement, 5-4
CANCEL (SVCOMPR command), 4-4
CANCEL command, 3-3
CA-Realia II Workbench
 using navigational commands, 10-1
 using the SQL Preprocessor, 9-2
CBSIO=, 2-8
CLOSEURT= (SVCOMPR command), 4-4
COBOL programs
 abends, 9-10
 connecting/disconnecting, 9-10
 linking a .dll, 9-5
 locking, 9-10
 mixed SQL and navigational, 9-11
 navigational commands in, 10-1
 precompiling, 9-6
 SQL statements in, 9-2
Code pages, 8-2
COMMIT WORK, 9-10
 specifying auto commit for a data source, 5-6
Communication utility
 input parameters, 4-4

- SVCOMPR batch job, 4-1
- SVCOMPR requirements, 4-2, 4-3
- VSE JCL, 4-3
- Configuration dialog, 5-1
- Conformance levels
 - API, A-1
 - navigational commands, A-3
 - SQL, A-1
- Connection
 - cancelling, 4-4
 - data source administration, 5-1
 - for COBOL programs, 9-10
- Connection methods
 - choosing, 5-11
 - determination, 5-11
 - SQLBrowseConnect, 5-13
 - SQLConnect, 5-14
 - SQLDriverConnect, 5-14
- Connection string
 - Connection Window, 5-13
 - SQLBrowseConnect, 5-13, 5-14
 - SQLConnect, 5-14
 - SQLDriverConnect, 5-14
- Connection type (Preprocessor option), 9-8
- Connection Window, 5-13

D

- Data source
 - connecting to mainframe, 5-1
 - default, 5-15
 - entering options, 5-14
 - ODBC Setup, 5-4, 5-12
 - Preprocessor option, 9-9
 - prompting, 5-14
 - SQLConnect, 5-14
- DATE=, 2-8, 2-9
- DB threads
 - reserving, 2-5
 - waiting, 3-3
- DBSQLE, 9-10
- DBUSERS=
 - reserving DB threads, 2-5
 - waiting for DB threads, 3-3

- DCODBC.INI, 5-10
- DDBID=, 2-6
- DDDID=, 2-6
- DECPOINT=, 2-8, 2-9
- Demonstrating successful installation, 2-1
- Diagnostics, 3-2
- Double-byte characters (DBCS), 5-6
- Drivers
 - JDBC, 6-1
 - ODBC, 5-13
- DTCMSRV, 2-1
- Dynamic User Requirements Tables
 - enabling, 2-6
 - priority, 2-10
- DYNURT=, 2-6

E

- EBCDIC to ASCII translation, 8-2
- EOJ (SVCOMPR command), 4-4
- EXPORT entrypoint, 9-5
- External Security
 - Server user logon, 2-7

I

- Installing the Java Server, 7-1
- ISOLEVEL=, 2-8

J

- Java
 - programmer documentation, 1-16
- Java Server, 7-1
- JCL, startup
 - description, 2-2, 2-3
 - statements, 2-2, 2-3, 4-2, 4-3

JDBC Driver, 6-1

L

Libraries, DSN statements, 2-3, 2-4, 4-2, 4-3

LOADURT=, 2-7

LOADURT= (SVCOMPR command), 4-4

LOGON=, 2-7

M

MAXURTS=, 2-7

Microsoft
programmer documentation, 1-16

MODIFY (SVCOMPR command), 4-4

MSG=, 2-9

Multi-User Facility
abend, 3-1
failure, 3-1
restarted, 3-1
startup, 2-1

N

Navigational commands
in C programs, 10-1
in COBOL programs, 10-1
unsupported commands, A-3

NT Registry editor, 7-4

O

ODBC Driver, 5-1

OPENURT= (SVCOMPR command), 4-4

Operation instructions, 3-1

OPT=, 2-8

OPTIONS (SVCOMPR command), 4-4

Overview, 1-1

P

PLANNAME=, 2-7

PLNCLOSE=, 2-8

Preprocessor
options, 9-7

Preprocessor options, 2-8

Priority (for dynamic URTs), 2-10

PRTY=, 2-8

R

Reconnect, 5-13

Restarting, 3-1

Rollback, 2-9, 3-1

ROLLBACK WORK, 9-10

S

Samples
VSE startup JCL, 2-3

Screens
Connection Window, 5-13
templates
CA-Datcom/DB ODBC Setup, B-3
CAICCI-PC Properties, B-5, B-6, B-7
Connection Window(CA-Datcom/DB
LOGON), B-4
Selected components, B-2

SECEXIT=, 2-8

Security, LOGON=, 2-7

Server
groups
changing without workstations, 3-2
designating, 2-5
JCL submission, 2-2, 2-4
Muti-User connection, 2-3, 2-4, 4-2, 4-3
JDBC setup, 6-1, 7-4

- listing, 3-2
- multiple Servers, 2-2, 2-4
- ODBC setup, 5-4, 5-12
- operation instructions, 3-1
- server name
 - SERVERNAME,JDBC Setup, 6-1, 7-4
- Server name
 - Connection Window, 5-17
 - SERVERNAME,ODBC Setup, 5-4, 5-12
- SERVERNAME=, 2-8
- user selection, 5-13

SERVERNAME=, 2-8

SQL

- conformance levels, A-1
- in CA-Realia II Workbench programs, 9-2
- Plan Options, 2-8
- Preprocessor
 - options, 9-7, A-3
 - usage, 9-2

SQLBrowseConnect, 5-13

SQLConnect, 5-14

SQLDriverConnect, 5-14

Startup

- JCL, 2-2, 2-3
- Multi-User, 2-1

STATUS (SVCOMPR command), 4-4

STATUS command, 3-3

STRDELIM=, 2-9

SVCOMPR, 3-3

SVCOMPR batch job, 4-1

SVDBSPR, 2-7

- SQL Plan Options
 - CBSIO, 2-8
 - DATE, 2-9
 - DECPOINT, 2-9
 - ISOLEVEL, 2-8
 - MSG, 2-9
 - OPT, 2-8
 - PLNCLOSE, 2-8
 - PRTY, 2-8
 - SQLMODE, 2-8
 - STRDELIM, 2-9
 - TIME, 2-9
 - TIMEMIN, 2-8
 - TIMESEC, 2-8
- SQL Plan WORKSPACEPTY, 2-8
- startup parameters

- ACCESS, 2-4
- APPLID, 2-5
- AUTHID, 2-5
- CONEXIT, 2-5
- DBUSERS, 2-5
- DDBID, 2-6
- DDDID, 2-6
- DYNURT, 2-6
- LOADURT, 2-7
- LOGON, 2-7
- MAXURTSS, 2-7
- PLANNAME, 2-7
- SECEXIT, 2-8
- SERVERNAME, 2-8
- SQL Plan Options, 2-8
- SQLMEM, 2-8
- SUPMSG, 2-9
- TIMEOUT, 2-9
- TIMEOUTWAIT, 2-10
- TRACEON, 2-10
- TRUSER1, 2-10
- TRUSER2, 2-10
- TRUSER3, 2-10
- TRUSER4, 2-10
- TRUSER5, 2-10
- URTPRTY, 2-10

System ID (Connection Window), 5-17

T

- Table (ODBC Setup window), 5-10
- Table caching, 5-6
- Table lists, limiting, 5-10
- Table masking, specifying for a data source, 5-5
- TableListMask= parameter (DCODBC.INI), 5-10
- Templates
 - CA-Datcom/DB ODBC Setup, B-3
 - CAICCI-PC, B-5, B-6, B-7
 - Connection Window(CA-Datcom/DB LOGON), B-4
 - Selected components, B-2
- Termination
 - communication utility, 3-1
 - Multi-User failure, 4-1
- TIME=, 2-8, 2-9
- TIMEMIN=, 2-8

TIMEOUT=, 2-9
TIMESEC=, 2-8
TRACEOFF (SVCOMPR command), 4-4
TRACEON (SVCOMPR command), 4-4
TRACEON=, 2-10
Translating double-byte characters, 5-6
Translation Table, specifying for a data source, 5-6
TRUSER1=, 2-10
TRUSER2=, 2-10
TRUSER3=, 2-10
TRUSER4=, 2-10
TRUSER5=, 2-10

U

URT, specifying maximum number, 2-7
URTPRTY=, 2-10
URTSTATUS (SVCOMPR command), 4-4

User ID (Connection Window), 5-16

User Requirements Tables

closing, 3-3
dynamic
 enabling, 2-6
opening, 4-7
priority for dynamic URTs, 2-10
status, 3-3

Using, 7-1

V

View lists, limiting, 5-10

W

WDBNTRY, 10-1

Wild card characters (in table masking), 5-10

WORKSPACE=, 2-8

Workstation
 monitoring, 3-3

