This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the “Documentation”) is for your informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, if you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or 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 YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2013 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.
CA Technologies Product References

This document references the following CA Technologies products:

- CA ACF2™ (CA ACF2)
- CA Detector® for DB2 for z/OS (CA Detector)
- CA Fast Load for DB2 for z/OS (CA Fast Load)
- CA Plan Analyzer® for DB2 for z/OS (CA Plan Analyzer)
- CA SQL-Ease® for DB2 for z/OS (CA SQL-Ease)
- CA Subsystem Analyzer for DB2 for z/OS (CA Subsystem Analyzer)
- CA Top Secret® (CA Top Secret)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.
Documentation Changes

The following updates have been made in the second edition of this documentation:

- Moved the batch reporting parameters to the new CA Detector Reference Guide.
- Moved the CA Detector table information to the "Summary of Objects" appendix in the Implementation Guide.

The following documentation updates have been made since the last release of this documentation:

- **Profile and Datastore Batch Utilities** (see page 26)—Clarified that CA Detector report and collection profiles created before r14 and datastores created before r15 must be carried forward using the provided batch utilities.
- **Tables**—New appendix that describes the columns in the following tables: PTI.PDT_STANDARD_#, PTI.PDT_OBJECT_#, PTI.PDT_STANTEXT_#, PTI.PDT_SQLERROR_#, PTI.PDT_ERRORTXT_#, PTI.PDT_ERRORVAR_#, PTI.PDT_DYNAMREQ_#, PTI.PDT_DYNAMTXT_#, and PTI.PDT_HOSTVARS_#.
- The EXPANS column has been removed from the CA Detector SQL metrics displays. This column relates to metrics that apply to DB2 versions that CA Detector no longer supports.
Contents

Chapter 1: Introduction 11
Benefits ......................................................................................................................... 11
Product Overview ......................................................................................................... 12
Architecture .................................................................................................................. 12
Control File ................................................................................................................ 13
Interproduct Functionality ......................................................................................... 14
CA Detector Main Menu ............................................................................................. 14
   PDT Global Command ............................................................................................ 15
   View and Select DB2 Subsystems ........................................................................... 15
   Customize the Panel Appearance .......................................................................... 16
   Date and Time Formats ......................................................................................... 16
   Masking ................................................................................................................... 18
How to Scroll and Expand Fields .............................................................................. 18

Chapter 2: Operational Considerations 19
Authorization and Security .......................................................................................... 19
   How to Control User Access Using DB2 Security ............................................... 20
   How to Control User Access Using External Security ........................................ 21
   Xmanager Started Task Authorization ................................................................ 21
   MVS Authorization Requirements ...................................................................... 22
Xmanager Preparation .............................................................................................. 23
Set Profile Options .................................................................................................... 23
DB2 Trace Requirements and Collection .................................................................. 25
Profile and Datastore Batch Utilities ........................................................................ 26
   Utility Requirements ........................................................................................... 27
   SYSIN Statement Examples .............................................................................. 27
Batch Reporting ......................................................................................................... 29
   How To Create Batch Reports ............................................................................ 29

Chapter 3: Using Collection Services 31
Collection Services ................................................................................................... 31
Start Collection .......................................................................................................... 32
   Overhead Considerations ..................................................................................... 35
   Collection Overhead ............................................................................................ 35
   Triggered SQL ....................................................................................................... 37
   Dynamic SQL Text Statistics .............................................................................. 37
Chapter 4: Viewing Collection Activity

Collection Activity .................................................................................................................. 63
Current and Historical Activity ............................................................................................... 64
View Standard Collection Activity .......................................................................................... 64
  View Collection Activity by Plan Name ............................................................................. 66
  View Collection Activity by Program ............................................................................... 73
  View SQL Calls .................................................................................................................. 80
  View Dynamic SQL Activity ............................................................................................. 87
  Evaluate Access Path Information .................................................................................... 89
  View Collection Activity by Reporting Profile ................................................................. 90
  View Application Profile History ..................................................................................... 91
  View Keys Collection Activity ......................................................................................... 92
View Exception SQL Collection Activity .................................................................................. 93
  View Exception SQL by User .......................................................................................... 95
  View Exception SQL By Program .................................................................................... 99
  View Exception SQL by Plan .......................................................................................... 101
  View Exception SQL by Reporting Profile ....................................................................... 103
  View Exception SQL By Correlation ID .......................................................................... 104
  View Exception SQL By Collection ID ............................................................................ 105
Chapter 5: Using Collection Profiles

Collection Profiles ............................................................................ 137
Create Collection Profiles .................................................................. 138
Global Defaults .................................................................................. 139
View/Modify Global Defaults ............................................................... 139
Application Groups ............................................................................ 141
  How Application Groups Are Applied ............................................. 142
View/Modify Application Groups ......................................................... 145
Create an Application Group ............................................................... 152
Delete an Application Group ................................................................. 154
Resource Groups .................................................................................. 154
View/Modify Resource Groups ............................................................... 154
Add Collection Profile Resource Groups ............................................... 156
View Collection Profile Resource Groups ............................................. 158
Update Collection Profile Resource Groups .......................................... 159
Exclusion Groups ................................................................................. 159
Limit Collection Activity ....................................................................... 160
Define SQL Error Groups ..................................................................... 162
Exclude SQL Error Codes ..................................................................... 163
Additional Keys Group ......................................................................... 164
Customize Additional Keys Group Collection ....................................... 165
Delete a Collection Profile .................................................................... 167

Chapter 6: Using Reporting Profiles ...................................................... 169
Reporting Profiles ................................................................................ 169
Space Requirements ............................................................................ 170
Create Reporting Profiles .................................................................... 170
Set the Current Profile ......................................................................... 171
View and Update Reporting Profiles ..................................................... 172
Add Profile Entries ............................................................................... 172
View Application Profile Entries .......................................................... 174
Update Application Profile Entries ....................................................... 174
Delete Profile Entries ........................................................................... 175
Delete Reporting Profiles ..................................................................... 176

Chapter 7: Using SQL Traces ................................................................. 177
SQL Traces ......................................................................................... 177
SQL Trace Prerequisites ...................................................................... 178
Distributed SQL Traces Collection ....................................................... 179
SQL Trace and DB2 Performance Trace ............................................... 179
Start an SQL Trace ............................................................................... 179
View SQL Trace Data ........................................................................... 181
View SQL Trace Program Data ............................................................. 185
View SQL Trace Workunit Flow Data .................................................... 187
View SQL Trace Sort Detail ................................................................. 189
View SQL Trace Program Sort Summary Data ..................................... 189
View SQL Trace Object Scan Data ....................................................... 189
View Application Lock Acquire Activity .............................................. 190
View Application Resource Use by SQL Statement Type ..................... 191
View SQL Activity Calls ...................................................................... 191
Chapter 8: Analyzing Current Subsystem Activity  

DB2 Threads ................................................................................. 201
View a DB2 Subsystem List .......................................................... 202
View Subsystem Threads .............................................................. 202
View Current Thread Activity ....................................................... 203

Chapter 9: Sample Scenarios  

How to View Application Activity .................................................. 205
Perform Access Path Analysis with CA Plan Analyzer .................. 206
Use SQL Trace ............................................................................ 209
  Initiate a Trace on Your Current SQL Statement ....................... 209
  Execute Test SQL ..................................................................... 210
  View SQL Trace Data .............................................................. 210
  View Event Application Flow .................................................. 211
  View Distributed Application Flow .......................................... 211
  Terminate the SQL Trace ......................................................... 212

Appendix A: Quick Reference  

Access the Tutorial ..................................................................... 213
Access CA Detector .................................................................... 213
Initialize a Datastore ................................................................... 214
Create Collection Profiles ......................................................... 217
Start Collection .......................................................................... 219
View Standard Activity Collection Data ...................................... 220
View Exception SQL Collection Data .......................................... 221
View SQL Error Collection Data .................................................. 222
View a History Display ................................................................ 223
Stop Collection ........................................................................... 224

Index ......................................................................................... 225
Chapter 1: Introduction

This section contains the following topics:

- Benefits (see page 11)
- Product Overview (see page 12)
- Architecture (see page 12)
- Control File (see page 13)
- Interproduct Functionality (see page 14)
- CA Detector Main Menu (see page 14)
- How to Scroll and Expand Fields (see page 18)

Benefits

There are many benefits to using CA Detector as your DB2 application analysis tool of choice. It helps you to:

- Understand application workload characteristics.
- Evaluate application performance at informative levels of granularity.
- Identify resource-intensive application components easily.
- Lower development costs and increase productivity.
- Evaluate the impact of potential SQL modifications.
- Take advantage of interproduct functionality.
- Monitor all DB2 subsystem activity in a Sysplex from a single TSO session.

If Thread Termination\Dynamic DSNZPARM is installed, you can also terminate a thread by selecting it from the list of active threads. You can use the CA Detector interface with Thread Termination\Dynamic DSNZPARM to cancel idle or runaway threads without impacting other transactions.
Product Overview

CA Detector is an innovative solution for the analysis of DB2 application performance. Database administrators, performance analysts, application developers, and system programmers can use it to evaluate and understand application characteristics and associated resource utilization. Robust displays, ease of use, and unique data collection techniques accelerate the task of identifying poorly performing application components. It complements your existing DB2 application performance tools by providing additional functionality that enhances your understanding of application workload and lets you expedite your application performance analysis efforts.

You can view DB2 accounting trace information from various application levels. You can view low overhead DB2 accounting information for DB2 applications, plans, DBRMs, packages, and SQL statements. Collection capabilities have been enhanced to interrogate SQL execution activity. Extended collection is optional and can be requested when collection is initialized. All major accounting trace performance information is collected at the SQL call level without the need for DB2 performance trace activity.

You can examine DB2 application use from an application, plan, and program point-of-view. By defining reporting profiles, you can observe application workload from an application perspective, so you can truly understand which applications account for the majority of your DB2 subsystem activity. You can also interrogate DB2 subsystem use from a plan or program perspective. Examining application activity from a program perspective makes it easy to identify heavily used programs and account for the majority of application SQL calls. In addition, you can easily identify which plans are using specific programs. Extensive online panels categorize application activity and resource use in a logical, effective fashion.

The collection and reporting facilities assist you in understanding application SQL error activity. You can view and evaluate application errors incurred as a result of abnormal SQL call return codes. Optionally, you can request SQL error collection when you initialize collection. Performance trace activity is not required for SQL error collection.

When you enable Xmanager cross-system communication support, you can transparently access any DB2 subsystem in a Sysplex from a single TSO session.

Architecture

The user interface and related panels are provided using TSO/ISPF. The panel logic is designed to interface seamlessly with other CA Database Management Solutions for DB2 for z/OS. You must also install and execute an MVS started task called Xmanager. Xmanager provides an execution environment used by CA Detector.

To use CA Detector, the Xmanager started task must be active. When you are viewing panels, communication occurs between your TSO session and the Xmanager started task.
Multiple TSO sessions can use CA Detector and relate to the Xmanager address space and the Xmanager address space relates to the various DB2 subsystems on your MVS system. Other products, such as CA Subsystem Analyzer and the Thread Termination\Dynamic DSNZPARM Value Pack component, also use Xmanager. Xmanager is structured to allow multiple products to coexist within a single Xmanager address space.

Various product features require asynchronous collection activity. For example, starting collection causes a request to be sent to the Xmanager started task. When the request is processed, collection starts and executes asynchronously to your TSO/ISPF session. If you logoff your TSO session, collection remains active within the Xmanager address space. The same type of processing occurs if an SQL trace request is issued. Other features communicate with Xmanager in a synchronous fashion.

Control File

CA Detector uses a control file that lets ISPF users perform the following tasks:

- Share related collection start variable information.
- View all collection datastores, including those created by other users.
- View and use all collection and reporting profiles, including those created by other users.

The control file must be created during the customization step of the post-installation process.

For more information about creating the control file, see the CA Database Management Solutions for DB2 for z/OS Implementation Guide.

If Xmanager cross-system communication support is enabled, all of the Xmanager tasks in the Sysplex must reference the same control file.
Interproduct Functionality

CA Detector is designed to interface seamlessly with CA Plan Analyzer to analyze DB2 access paths for packages, DBRMs, and SQL calls. CA Plan Analyzer provides important supplemental information about optimizing SQL access paths, physical designs, and plan enhancements. Global commands can also be used from other CA Database Management Solutions for DB2 for z/OS. For example, you can activate the CA Detector SQL Trace facility directly from within CA SQL-Ease or any other CA Database Management Solutions for DB2 for z/OS.

Seamless integration with CA Subsystem Analyzer lets you evaluate subsystem performance as well as application performance. You can jump between the products while viewing real-time or historical information. Collection data between CA Subsystem Analyzer and CA Detector is synchronized using a common collection interval.

You can use the CA Detector interface with the Thread Termination\Dynamic DSNZPARM Value Pack component to cancel idle or runaway threads without impacting other transactions. You can quickly and easily terminate a thread by selecting it from a list of active threads.

CA Detector Main Menu

The CA Detector Main Menu displays the available product options.

---

```
Rnn.nn ----------------- CA-Detector Main Menu ----------------- yy/mm/dd       hh:mm
OPTION ===> 
DB2 SSID ==> ssid      USERID ==> USERA01     DS Group ==> DSNGRP

----------------------------------------------------------
1  SSID current interval data  9  DS Group current interval data
2  SSID historical interval data 10  DS Group historical interval data
3  SSID collection status     11  DS Group collection status
4  Maintain collect/report profiles  P  Product profile
5  Initialize SSID collection   T  Tutorial
6  Terminate SSID collection   X  Exit Detector
7  Create/Initialize datastore
8  SSID, Active thread, SQL trace displays

----------------------------------------------------------
```

---

Copyright (c) 2009 CA All Rights Reserved
---
The DB2 SSID field identifies the currently attached DB2 subsystem. The default value is taken from the CA Database Management Solutions for DB2 for z/OS Main Menu; however, you can change it at any time to switch between subsystems. Leave this field blank to select from a list.

The USERID field specifies the TSO user ID of the user currently logged on to CA Detector.

The DS Group field specifies the DB2 data sharing group name used when invoking the data sharing group displays. This name is independent of the DB2 SSID value being used; meaning it does not have to be that of the specified DB2 subsystem. The DS group name is retained in your ISPF profile.

Enter a value in the Option field to begin using CA Detector.

**Note:** The online tutorial (option T) introduces new users to the features and capabilities quickly.

### PDT Global Command

You can use the PDT global command to jump directly to the CA Detector Main Menu from other products.

**Note:** This command is valid only when a CA Detector session is not already active.

### View and Select DB2 Subsystems

You can view a list of remote locations and subsystems and their load libraries and install system administrators (SYSADM). From this list, you can select the DB2 subsystem you want to manage.

**Note:** These subsystems are specified in the SETUP member of `high-level.CDBAPARM` during post-installation processing.

**Follow these steps:**

1. Leave the DB2 SSID field blank on the CA Detector Main Menu and press Enter.
   
   The CA-DB2 SSID/Location Selection panel appears.
   
   **Note:** Press F1 to access the online help panels for a description of the fields on this panel.

2. Type $ in the selection line before the SSID (or location) you want to use and press Enter.

   Your selection is saved and appears in the DB2 SSID field of the previous panel.

   **Note:** When you select a location, you automatically select its corresponding SSID.
Customize the Panel Appearance

You can use the FREEZE, EXCLUDE, and SORT commands to customize panels based on your site requirements. After the customization is performed, you can use the REPORT command to save the panel format changes.

The information saved includes the column order in the panel and any sort commands used to sort the displayed data. You can tailor the panels based on your requirements, verifying that you see information in the same format each time you access a panel without having to reissue the commands.

**Note:** For more information about these commands and scroll options, see the General Facilities Reference Guide.

**Follow these steps:**

1. Enter the following commands to customize the display of information on the panel:
   
   ```plaintext
   FREEZE
   EXCLUDE
   SORT
   ```
   
   The column and sort order changes on the panel are updated as specified.

2. Type `REPORT` on the command line and press Enter.
   
   The format of the current panel is saved to an ISPF profile variable. The information saved includes the column order on the panel and any sort commands used to sort the displayed data.

   **Note:** You can return to the default settings by entering `REPORT RESET` on the panel.

Date and Time Formats

Many fields contain date, time, or timestamp information. There are three field types:

**Date**

Indicates the date on which an event occurred or an interval began.

**Elapsed Time**

Indicates the time duration of an event or an interval.

**Timestamp**

Indicates the time at which an event occurred or an interval began.
The date and time formats use the following abbreviations:

- MM-(Month)
- DD-(Day)
- YY-(Year)
- HH-(Hours)
- MM-(Minutes)
- SS-(Seconds)
- Ss-(Hundredths of Seconds)
- Sss-(Thousandths of Seconds/Milliseconds)
- Sssss-(Millionths of Seconds/Microseconds)

The date and time formats are summarized in the following table:

<table>
<thead>
<tr>
<th>Time Data Type</th>
<th># of Bytes</th>
<th>Time Value</th>
<th>Time Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>8</td>
<td>(N/A)</td>
<td>MM/DD/YY</td>
</tr>
<tr>
<td>Elapsed Time</td>
<td>8</td>
<td>&lt; 1 hour</td>
<td>MM:SS.ss</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt; 1 hour</td>
<td>HH:MM:SS</td>
</tr>
<tr>
<td>Elapsed Time</td>
<td>12</td>
<td>&lt; 1 hour</td>
<td>MM:SS.sssss</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt; 1 hour</td>
<td>HH:MM:SS.sss</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt; 24 hours</td>
<td>DD-HH:MM.SS</td>
</tr>
<tr>
<td>Timestamp</td>
<td>8</td>
<td>(N/A)</td>
<td>HH:MM:SS</td>
</tr>
</tbody>
</table>

**Note:** Because time field contents are variable format they are not suitable for sorting. Instead use the corresponding percent field. For example, to order a display by CPU usage, sort the CPUPCT column not the INDB2_CPU column.
Masking

Use only the percent sign (%) and the underscore (_) to specify selection criteria as follows:

**Note:** This formatting is different than the selection criteria described in the *General Facilities Reference Guide*.

- **Percent sign (%)**—Use the percent sign only at the end of the string to represent any number of characters. The percent sign is a suffix match symbol. The string you enter is read until the percent sign is reached. Anything you enter after the percent sign is not read.  
  For example, if you enter DSN%123, DSN is read, but not 123. All members that begin with DSN are returned with no limits imposed on what follows the initial three characters. To mask an indeterminate number of characters, use a single percent sign.

- **Underscore (_)**—Use the underscore anywhere within the string to represent any single character for each underscore you enter. Blanks are not allowed.  
  For example, if you enter DSN _ _ _ _ ABC, you can receive DSN1234ABC but not DSN1ABC or DSN12ABC or DSN123ABC. In this example, only members with exactly four characters between DSN and ABC are returned.

**More information:**

*Add Profile Entries* (see page 172)

How to Scroll and Expand Fields

When your data values are longer than the display area, you can scroll and expand the fields and columns to view or input data as follows:

- For fields that cannot be scrolled, like header fields, place your cursor in the field to be expanded and specify the EXPAND command (typically, using a PF key setting).

- For scrollable fields, the greater than sign (>) appears for fields with truncated values when the column is not wide enough to display the long value. You can view or edit this data using scroll commands, or the EXPAND and SETWIDTH commands.

**Note:** For more information about these commands, enter a question mark (?) on the command line or see the *General Facilities Reference Guide*. 
Chapter 2: Operational Considerations

This section contains the following topics:

Authorization and Security (see page 19)
Xmanager Preparation (see page 23)
Set Profile Options (see page 23)
DB2 Trace Requirements and Collection (see page 25)
Profile and Datastore Batch Utilities (see page 26)
Batch Reporting (see page 29)

Authorization and Security

Global product authorization is implemented through DB2 security or external security using an external security manager product (like CA ACF2, CA Top Secret, or IBM RACF). Use the EXTERNALSEC option in the PDT parmlib member in hlq.CDBAPARM to control the type of security used. DB2 security is used by default.

For more information about editing product parmlib members, see the Implementation Guide.

For DB2 security, EXECUTE authority is required on the CA Detector plans for the following product functions. For external security, the security resource entity and required access that are listed for each function apply.

Detector

Allows the user to access any of the product functions. Authorization to use this function is required before access to other product functions can be provided.

Security resource entity: PDT.LEVEL1.ssid
Required access: READ

View Displays

Allows the user to view current and historical CA Detector displays.

Security resource entity: PDT.LEVEL2.ssid
Required access: READ

Batch Reporting

Allows the user to generate reports and unload data from a data store.

Security resource entity: PDT.LEVEL3.ssid
Required access: READ
View SQL Trace

Allows the user to view, start, or stop an SQL trace.

Security resource entity: PDT.LEVEL4.ssid

Required access: READ

Collection Services

Allows the user to start and stop a collection and create a collection data store.

Security resource entity: PDT.LEVEL5.ssid

Required access: READ

SQL Trace Admin

Allows the user to modify SQL trace collection performance-related parameters.

Security resource entity: PDT.LEVEL6.ssid

Required access: UPDATE

Coll Profile Update

Allows the user to create and modify collection profiles.

Security resource entity: PDT.LEVEL7.ssid

Required access: UPDATE

View Host Variables

Allows the user to view host variables.

Security resource entity: PDT.LEVEL8.ssid

Required access: READ

How to Control User Access Using DB2 Security

The Product Authorizations Facility lets you use DB2 security to control user access to the product functions.

All users must be granted execute authority on the product plans. Plan execute authority is granted on a DB2 subsystem basis. Execute authority must be granted on all DB2 subsystems on which the product is used.
Note: For more information about using the Product Authorizations Facility, see the General Facilities Reference Guide.

Follow these steps:

1. Type A in the Option line on the CA Database Management Solutions Main menu and press Enter.
   The Product Authorizations panel appears.
2. Type E (explode/implode) next to the product name or function and press Enter.
   A list of available plans appears for the selected product or component.
3. Grant permission to use the resources as needed.

How to Control User Access Using External Security

DB2 security is used by default. However, you can also control access to the product functions by using an external security manager product. An interface is provided to CA ACF2, CA Top Secret, and IBM RACF using the Security Authorization Facility (SAF). This interface provides a common means for identifying a user and authorizing user access to critical resources.

To enable external security, follow these steps:

1. Define the CADB2 resource class name to the security system.
2. Define access to resources using the available access levels (see page 19). These levels of security correspond to the DB2 plan authorizations you can set using the Product Authorizations Facility.
   Note: The resource names are not based on data set class security definitions even though the resource names look like data set names. Lowercase nodes in the resource name indicate that you can use a wildcard asterisk.
3. Set the product parmlib EXTERNALSEC parameter to Y.

Xmanager Started Task Authorization

Additional authorities for the Xmanager started task are required to use SQL Trace.

The Xmanager address space ID must be authorized to create threads on all DB2 subsystems on which the CA Detector SQL Trace facility is run. You might have to grant Xmanager access through your security system. For example, if you are using RACF general resource class DSNR profiles to control access to DB2, Xmanager must be granted access in the appropriate profiles within the DSNR resource class.

Note: For more information, see the PDTSQTGR member in the CDBASRC installation data set for more information.
MVS Authorization Requirements

Functions requested by TSO users can occasionally result in data set access requests on behalf of the user to occur in the Xmanager address space. To accomplish these tasks, the Xmanager address space might attempt to create a security environment for the users in the Xmanager address space using the MVS RACROUTE macro. The Xmanager address space might require authorization to create the security environment within its address space. If the facility class is active within the security subsystem, then users must be granted access to the Xmanager facility. You should contact your security administrator to determine if updates to the security subsystem are needed.

The following table shows the MVS authorizations that are required for CA Detector TSO users and for Xmanager.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>TSO User Access Required</th>
<th>Xmanager Access Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control file</td>
<td>Read</td>
<td>Read</td>
</tr>
<tr>
<td></td>
<td>Update</td>
<td>Update</td>
</tr>
<tr>
<td>Collection datastore</td>
<td>Create</td>
<td>Read</td>
</tr>
<tr>
<td></td>
<td>Read</td>
<td>Update</td>
</tr>
<tr>
<td>Collection profiles</td>
<td>Create</td>
<td>Read</td>
</tr>
<tr>
<td></td>
<td>Read</td>
<td>Update</td>
</tr>
<tr>
<td>Reporting profiles</td>
<td>Create</td>
<td>None required</td>
</tr>
<tr>
<td></td>
<td>Read</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Update</td>
<td></td>
</tr>
<tr>
<td>SQL trace control file</td>
<td>Create</td>
<td>Read</td>
</tr>
<tr>
<td></td>
<td>Read</td>
<td>Update</td>
</tr>
<tr>
<td>SQL trace data set</td>
<td>Create</td>
<td>None required</td>
</tr>
<tr>
<td></td>
<td>Read</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Update</td>
<td></td>
</tr>
<tr>
<td>Xmanager parmlib data set</td>
<td>None required</td>
<td>Read</td>
</tr>
<tr>
<td></td>
<td>Update</td>
<td></td>
</tr>
<tr>
<td>DB2 catalog data sets</td>
<td>None required</td>
<td>Read</td>
</tr>
</tbody>
</table>
**Xmanager Preparation**

CA Detector requires a component called Xmanager. Xmanager executes as a started task in its own address space and is used to establish and provide an execution environment for CA Detector. The execution environment is used to perform synchronous and asynchronous collection-related tasks. In a TSO environment, CA Detector communicates with Xmanager to perform tasks required by the various CA Detector panels and collection processes.

In a sysplex complex, Xmanager can be requested to provide a distributed communications environment. This provides CA Detector with transparent access to any DB2 subsystem in the sysplex complex executing on a z/OS platform that has an active Xmanager task. If you implement cross-system support, all Xmanager tasks must reference the same control file to access datastores across the sysplex. The control file is specified in the DTCFPARM member of the Xmanager parameter library (high-level.CDBAPXMP).

Xmanager also recognizes a set of MVS modify commands that allow you to automate SQL data collection and trace activity.

*Note:* For more information about setting up and operating an Xmanager, implementing cross-system support, and MVS modify commands you can specify with the Xmanager, see the *General Facilities Reference Guide*.

**Set Profile Options**

The CA Detector profile sets variables that are specific to CA Detector. It can be used to configure SQL trace performance parameters. These parameters specify the buffer sizes and read intervals used to collect trace information.

DTTR0000 is the default shipped member.

*Note:* For more information about how to set up SQL trace using the DTTRxx CDBAPXMP member, see the *Implementation Guide*.

**Important!** The default values for the SQL trace performance parameters have been carefully selected and should be appropriate for almost any, if not all, situations. Altering these values will affect the performance of SQL traces. If a value is not shown or found for any of the parameters, the defaults specified in DTTR0000 are used instead.

*Note:* You can also set global profile options that apply to all CA Database Management Solutions for DB2 for z/OS. For more information about setting global profile options, see the *CA Database Management Solutions for DB2 for z/OS General Facilities Reference Guide*. 
Follow these steps:

1. Type P in the Option line on the CA Detector Main Menu and press Enter.
   The Profile Menu panel appears.

2. Type 2 in the Option line and press Enter.
   The CA Detector Profile Menu panel appears. From this panel, you can view and modify CA Detector profile variables.

3. Type 1 in the Option line and press Enter.
   The SQL Trace Performance panel appears. From this panel, you can view and modify SQL trace performance parameters.

4. Complete the following fields to modify the values in the DTTRxx CDBAPXMP member:

   **Note:** These values apply to all traces on the specified subsystem.

   **DB2 SSID**
   Specifies the DB2 subsystem identifier for which you are setting parameters.
   **Default:** 0000
   The default member was created during installation and is a global CA Detector SSID that applies to all subsystems that do not have their own SQL trace parameters. If you have not previously defined parameters for a specific subsystem, this function creates an Xmanager parmlib member for that subsystem automatically.

   **Member**
   Identifies the name of the Xmanager parmlib member for the specified subsystem. Verify this value is correct.

   **Trace Buffer Size (TRACE_BUFSIZE)**
   Specifies the amount of memory, in kilobytes, you want allocated to the SQL trace buffer. Increasing the buffer size helps your system avoid possible buffer overrun during trace data collection.
   **Default:** 752 KB
   **Limits:** 300 to 1024

   **Read Interval (READ_INTERVAL)**
   Specifies the time interval during which the SQL trace is to collect data. This is the read interval. Enter the interval in 1/100 of a second increments. Increasing this interval reduces CPU overhead but could result in lost data. The minimum increment is 10 and the maximum is 99.
   **Default:** 25
Number of 32K Writer Buffer (WRITER_BUFFER_COUNT)

Specifies the number of 32 KB buffers to be allocated for writing the SQL trace data set.

**Limits:** A number between 10 and 99

**Default:** 25

Delete this member

Indicates whether to delete or retain the parmlib member identified in the Member field.

Press Enter.

The DTTRxx CDBAPXMP member is modified, the profile is saved and your selections are processed.

---

**DB2 Trace Requirements and Collection**

Most DB2 performance information is collected when DB2 accounting trace is active. DB2 collects this information internally, whether DB2 accounting trace is active or not. Activation of accounting class 1 merely informs DB2 that you would like this information externalized to a selected DB2 trace destination. Activation of accounting class 2 (INDB2 times) and class 3 (wait times and counts) causes additional data collection activity to occur.

CA Detector collection activity relies on accounting trace information. When collection is active, accounting data values are made available for viewing from the application, plan, package, DBRM, and SQL statement levels. Collection functions and provides information in its various panels even if accounting trace class 1 is not active. This activity occurs because DB2 collects most accounting class data values internally, even if DB2 accounting class 1 is not active.

The following areas within collection have DB2 trace implications:

- **Wait times and counts**

  CA Detector panels contain various data values that originate from DB2 accounting class 3 collection. If accounting class 3 or monitor class 3 is not active on the DB2 subsystem where collection is active, wait time and wait counts populated by class 3 collection are zero. Monitor class 3, which essentially causes the same trace collection activity to occur within DB2 as accounting class 3, is frequently relied on by various performance monitors. Therefore, if monitor class 3 is active and accounting class 3 is not, the wait times and wait counts are available to you. CA Detector collection does not enable performance trace activity.
Thread package or DBRM level accounting information

CA Detector panels let you access Thread Termination\Dynamic DSNZPARM panels to view thread package or DBRM level accounting information. If accounting class 7 or monitor class 7 is not active on the DB2 subsystem where collection is active, thread package or DBRM counts populated by class 7 collection are not available.

Profile and Datastore Batch Utilities

You can use the profile and datastore batch utilities to create, copy, and delete CA Detector collection profiles, report profiles, and datastores.

Note: Profiles that are created before r14 and datastores that are created before r15 are not compatible with the current release. To carry them forward, use these utilities.

Sample JCL is provided in the following members of the high-level.CDBASRC data set:

PDTCPALC
  Creates collection profiles in batch.

PDTCPCPY
  Copies collection profiles in batch. You can use this utility to migrate collection profiles from an earlier product release and manage profiles for all of your DB2 subsystems.

PDTCPDEL
  Deletes collection profiles in batch.

PDTRPALC
  Creates report profiles in batch.

PDTRPCPY
  Copies report profiles in batch. You can use this utility to migrate report profiles from an earlier product release and manage profiles for all of your DB2 subsystems.

PDTRPDEL
  Deletes report profiles in batch.

PDTDSALC
  Creates the datastores in batch.

Note: This JCL is the same utility JCL that is generated when you select Batch JCL = Y on the Detector Create Datastore Display panel.
**PDTDSCPY**

Copies the datastores in batch. To migrate datastores from an earlier product release and copy an existing datastore to a larger datastore or to one with more intervals, use this utility.

**PDTDSDEL**

Deletes the datastores in batch.

### Utility Requirements

The batch utilities have the following requirements:

- The create and delete utilities require Create-level access to the profile and datastore data sets from your security subsystem.
- The copy utilities require Read and Update access to the profile and datastore data sets from your security subsystem.

In addition to the data set access requirements, the report profile and datastore copy and delete utilities require CA Detector main plan authorization, and the collection profile copy and delete utilities require CA Detector collection profile update authorization.

### SYSIN Statement Examples

The following examples include some parameters that could appear in the batch utility SYSIN statements.

**Note:** For detailed descriptions of all SYSIN statements, see the sample JCL for each utility.

#### Example: Report Profile Create Parameters

The following example creates a plan report profile. The SYSIN parameters correspond to the fields on the Detector Create Reporting Profile Display. In this example, the report data set allocation is directed to a specific volume; however, you can specify SYSIN parameters to create either SMS or non-SMS managed data sets.

```
SSID(D81B)
PROF(ACCTING)
VCAT(CA.DB2.TOOLS)
DESC('Accounting Plans')
TYPE(P)
UNIT(TRKS)
PRI(5)
SEC(5)
VOL(VOL001)
```
Example: Collection Profile Copy Parameters

The following syntax copies the contents of a collection profile to another collection profile:

```
ISSID(DF3G)
IVCAT(PDT.R11)
IPROF(OLDCPROF)
OSSID(DF1G)
OPROF(NEWCPROF)
```

Note the following considerations:

- The input and output profiles can be defined to the same subsystem or different subsystems.
- The input profile can be for the current release or a prior release. Because the input profile can be defined to a control file for a prior release (to which the utility does not have direct access), the IVCAT parameter is required to build the input profile data set name.
- The output profile must already exist on the current control file and is always set to the current release, so a corresponding OVCAT parameter is not required. The contents of the output profile are completely replaced by the contents of the input profile.

Considerations are similar when copying a datastore:

- The contents of the output datastore are completely replaced by the contents of the input datastore. However, the output datastore can be defined with a different number of collection intervals than the input datastore.
- If the output datastore has fewer intervals than the input datastore, the most current input intervals are copied, then number of intervals defined in the output datastore.
- When the output datastore has more intervals than the input datastore, the additional intervals are marked as unused.

Example: Datastore Delete Parameters

The following example deletes an existing datastore:

```
SSID(D71A)
VCAT(CA.DB2.TOOLS)
DST(DATASTOR)
```

Note: You can create multiple datastores with the same datastore name but different high-level qualifiers, so the VCAT parameter is required to completely specify the datastore to be deleted. Because report and collection profile names must be unique, the VCAT parameter is not required for the profile delete utilities.
Batch Reporting

Using the batch reporting facility, you can create reports in batch from a VSAM collection datastore that can then be reviewed online or in print. This lets you schedule DB2 activity reporting for off-peak times. You can also determine the optimal formatting of reports for your needs, set your parameters once, and reuse the JCL to ensure a consistent report format for reviewing your DB2 activity. This saves you the time and effort involved in customizing online panels when viewing datastore information.

You can also use the batch reporting facility to perform the following tasks:

- Generate reports that consolidate statistics across the members of a data sharing group
- Unload records from a datastore into a sequential file.
- Load data into DB2 tables. This lets you use SQL to query the tables for gathering DB2 activity information.
- Generate CA Plan Analyzer EXPLAIN control cards for your SQL which can then be used by CA Plan Analyzer to process the EXPLAINs.

How To Create Batch Reports

You can execute the batch reporting facility to generate reports in batch by submitting the JCL provided in high-level.CDBASRC(PDTBATCH). Using this job, you can specify what reports to generate, what time intervals to include, and what selection criteria to use.

Follow these steps:

1. Modify the PDTBATCH sample JCL using the batch reporting parameters and save your changes.

   The modifications must include a valid job card, the XMANID being used, libraries for your site, target data sets, and input parameters for the batch reports to be generated. Two significant parameters are UNLOAD and EXPLAIN: the UNLOAD parameter controls whether the filtered data is written to a sequential data set while the EXPLAIN card controls if CA Plan Analyzer EXPLAIN cards are generated. See the sample member for details about making these modifications.

2. Submit the JCL for execution.

Note: For more information about batch reporting, see the CA Detector Reference Guide.
Chapter 3: Using Collection Services

This section contains the following topics:

Collection Services (see page 31)
Start Collection (see page 32)
Stop Collection (see page 39)
Datastores (see page 39)
Exception SQL Text Table (see page 47)
View Collection Status (see page 47)
Collection Control Using MVS Modify Commands (see page 48)
How To Synchronize Collection (see page 59)

Collection Services

CA Detector collection services let you start and stop CA Detector collection within the Xmanager address space from an ISPF session, create a datastore, and view collection status. Collection is started and stopped on a DB2 subsystem basis. Multiple collectors can be active concurrently for multiple DB2 subsystems. While the collector is active, you can externalize the data collected.

Note: CA Detector collection is an asynchronous activity that executes within the Xmanager address space.

Use collection services to analyze SQL call activity to help you understand DB2 application activity and resource use. Collection analyzes application activity from plan, package, DBRM, and SQL call perspectives with low overhead and provides DB2 accounting trace information about plans, packages, DBRMs, and SQL calls.

When you initiate dynamic, exception, or SQL error collection, you use a collection profile to specify the amount of dynamic SQL text and the type and amount of exception SQL and SQL error data accumulated during collection. You can also use the collection profile to manage exception SQL activity based on resource thresholds you specify. For example, you can set CPU time, getpage, or rows returned thresholds to limit the exception SQL activity captured, or terminate or preempt an SQL request if it exceeds those thresholds. Your collection profile can also control SQL error collection by excluding SQL error conditions that are of no interest or by limiting the amount of SQL error information to be retained.

If you have CA Subsystem Analyzer installed, you can coordinate collection between CA Detector and CA Subsystem Analyzer to receive additional data on application and SQL performance.

Note: You can also control collection using MVS modify commands.
Start Collection

You can start collection to gather information about the processes running on your subsystem from an ISPF session or using MVS modify commands.

Only one collector can be active for a selected DB2 subsystem, although multiple collectors can be active concurrently on different DB2 subsystems.

Note: By design, the following dynamic statement types are not collected: COMMENT, COMMIT, CONNECT, EXPLAIN, GRANT, LABEL, RELEASE, REVOKE, ROLLBACK, and SET.

Follow these steps:

1. Type 5 (Initialize SSID collection) in the Option line on the CA Detector Main Menu and press Enter.

   The Start Collection Display panel appears.

2. Specify the collection options, including the interval time, whether the collection is externalized, and whether the collection is started or restarted automatically. Press Enter.

   Note: The sysplex interval is maintained with the Xmanager PLEXINTV command. For more information about the Xmanager PLEXINTV command, see the General Facilities Reference Guide.

   The next DETECTOR Start Collection Display panel appears.

3. Complete the SQL collection and exception collection options:

   Triggered SQL Collection

   Performs more monitoring for SQL executed from DB2 triggers and determines whether to capture SQL originating from native SQL stored procedures.

   Note: This type of collection introduces a slight additional overhead. If you do not have triggered SQL or do not want triggered SQL statistics, specify N. This option also applies to SQL from native stored procedures. These statistics do not include table and index data.
Plan Exclude/Include List

Includes or excludes collection for plans defined in the Plan Exclude/Include list. When this option is set to Y, the list is loaded from the specified collection profile. If any exception collection options are specified, plans in the list are included or excluded.

This processing can be activated even if the collection does not include any exception collection options.

Standard Activity

Initiates collection for static and dynamic SQL activity.

To capture dynamic SQL, exceptions, and data based on user-defined keys, select this option.

Dynamic SQL Stats

Collects SQL statistics for dynamic SQL text strings. When this option is enabled, you can collect the aggregate statistics for each unique dynamic SQL text string that is executed during the collection interval. This option is independent of SQL exception collection. A collection profile is required.

These statistics do not include SQL host variables. To include this information, enable SQL exceptions.

View By Keys

Collects data based on the values defined for the additional View By keys. When this option is enabled, more thread identification information is collected. This information lets you organize standard activity data by criteria such as user ID and correlation ID, in addition to the standard plan hierarchy view. The collection profile additional keys group identifies the specific thread information and connection types you want to collect information about in your collection.

This option is independent of SQL exception and SQL error collection. A collection profile is required.

Dynamic Exceptions

Initiates exception collection for dynamic SQL activity. When this option is enabled, exception data on dynamic SQL activity is collected.

If you want to externalize the data, specify a data store name that was created with the exception SQL option. A collection profile is required.

Static Exceptions

Initiates exception collection for static SQL activity. When this option is enabled, more monitoring for SQL performance exceptions for static SQL statements is performed.

Standard activity and dynamic SQL exception data must also be collected to collect static SQL data.
**Exception cache size**

Specifies the number of 1-MB blocks in the exception collection storage cache. This value can reduce the exception collection overhead at the cost of some additional storage.

Once established, this value can increase during times of high activity or can decrease with lower system activity.

**Limits:** Must be a positive number no greater than 20. If 0 is specified, the exception cache is not allocated.

**SQL Errors**

Collects data on SQL error activity.

To capture SQL error text, select this option.

**Note:** To externalize the data, specify a data store name that was created with the SQL errors option.

**SQL Error Text**

Collects only SQL error text data. All successful (nonnegative) SQL requests are excluded from the collection. SQL text that is associated with negative SQL return codes is collected even if CA Detector does not record a performance exception for the SQL statement. A slight increase in the overhead is expected when this option is enabled.

If you want to externalize the data, specify a data store name that was created with the SQL errors option. CA Detector always stores SQL call text for dynamic SQL statements in the data store exception data set. If SQL Error Text collection is enabled without enabling dynamic or static exception collection, older intervals of exception data can roll off the data store. Older intervals of exception data roll off the data store to accommodate intervals containing only SQL error-related SQL Call text.

**Host variables**

Initiates collection of host variables. This option applies to SQL exceptions and SQL errors.

You must have enabled at least one of dynamic or static SQL exceptions or SQL error text collection to enable host variable collection.
Collection Profile

Specifies the name of the collection profile to use when any options that require a collection profile have been enabled.

Press Enter.

A message appears to verify that collection has started successfully. Informational messages are also written to the Xmanager job log and the MVS syslog. These messages include the user ID of the person who started the collection.

Note: If an MVS modify command is used to start collection, the value of the user ID is CONSOLE.

Overhead Considerations

If you enable the additional SQL and SQL exception collection options when you start collection, you may experience additional overhead related to the options you select.

Collection Overhead

CA Detector SQL activity collection adds a small amount of fixed overhead to each SQL statement executed by your DB2 applications. CA Detector does not have any impact on your applications outside of their SQL activity. While this overhead is generally negligible, you should be aware of the overhead factors and the options available to minimize any application overhead from CA Detector collection activity:

- A significant amount of CA Detector SQL activity collection occurs in the Xmanager address space. Because of this, the dispatching priority of the Xmanager address space can have an impact on the overhead associated with CA Detector SQL activity collection. This is especially true if you have enabled SQL exception collection.

  Note: For more information about Xmanager dispatching requirements, see the General Facilities Reference Guide.

- If you have enabled SQL exception collection, you should consider enabling the exception collection storage cache, described earlier in this chapter. Without the cache enabled, CA Detector SQL exception collection activity requires access to the cross system lock for the Xmanager address space. In some environments, this locking activity can have an impact on the performance of DB2 applications. Use of the cache can significantly reduce or eliminate this locking activity. Enabling and monitoring the exception collection storage cache is discussed in detail in this chapter.
Because SQL exception collections add overhead beyond standard SQL activity collection, you may want to consider limiting SQL exception collection. One approach would be to use standard SQL activity collection to broadly monitor your DB2 application activity and use SQL exception collection to perform detailed monitoring of those applications that appear to represent problem areas. You can accomplish this by turning off use of the global exception defaults in the collection profile and establishing application and resource groups to perform SQL exception collection tailored to the applications of interest.

Do not set the global exception defaults or the maximum number of exceptions collected in an interval in the collection profile to zeros. Doing so may result in CA Detector attempting to generate an exception for every SQL statement executed by your DB2 applications, which may result in an unacceptable level of collection overhead.

Both dynamic SQL statistics collection and exception SQL collection require additional collection overhead. You should avoid concurrently enabling both collection options during a single collection session. If you want to enable both collection options concurrently you can reduce the collection overhead by disabling global exception collection on the collection profile global defaults and using application groups to limit exception SQL collection to targeted plans and programs.

You can effectively turn off CA Detector SQL activity collection and any associated overhead for the programs using a specific plan by adding the plan to the collection profile plan exclusion group. Since doing this turns off all SQL activity collection for applications, using the plan it is not a recommended action, however, some installations may find this useful for certain fetch intensive plans where application tuning is not a consideration such as DSNTIAUL. A better approach to this type of situation would be to create a collection profile application group for the plan that excludes the plan from static and dynamic SQL exception collection. This still allows the plan SQL activity to be monitored while eliminating the overhead of monitoring for SQL exceptions where there is not any interest.

If you choose to exclude a plan from SQL activity collection, you should be aware that the INDB2 time and CPU on the CA Detector SQL activity displays are based on the SQL actually monitored by Detector. For example, if you exclude DSNTIAUL and it accounts for 5% of your total DB2 application CPU, the INDB2 CPU Percent column on the Detector SQL activity displays still add up to 100% even though the INDB2 CPU only accounts for 95% of your total DB2 application CPU.

More information: Using Collection Profiles (see page 137)
Triggered SQL

Collecting triggered SQL activity introduces a slight additional overhead. If you have no triggered SQL or do not want triggered SQL statistics, do not enable this option when you start collection.

Note: This option also applies to SQL activity from native SQL stored procedures.

Dynamic SQL Text Statistics

You can collect SQL statistics during a collection interval for dynamically executed SQL statements when you start collection. Enabling this option can significantly increase storage use in the Xmanager address space.

If you are only interested in collecting statistics for individual SQL statements that exceed performance thresholds, enable SQL exception collection instead.

Important! Enabling this option and SQL exception collection concurrently may result in high collection overhead and is not recommended.

Additional View By Keys Storage Impact

Enabling additional keys collection can result in a significant increase in main storage requirements for active collections and DASD requirements for externalized historical activity due to the increased granularity. For example, if ten users execute a plan that executes a package containing two SQL statements, it will require four collection buckets during active collection or four datastore records for historical activity. If additional keys collection is enabled for the user ID key, it will require forty collection buckets or datastore records to record the same activity.

If you enable the Additional View By Keys SQL collection option, limit additional keys collection to one or two additional views and connect types that you consider to be the most important. Enabling this option for all additional keys and connection types will most likely result in an excessive amount of real storage and DASD storage for current and historical activity data.
Exception Collection Storage Cache

SQL exception collection requires the Xmanager address space cross memory lock. In some environments, contention for the cross memory lock can result in degraded application performance. If you believe that you have application performance issues related to enabling SQL exception collection, you can allocate an exception collection cache. In most cases, allocating a sufficiently large exception collection storage cache can alleviate any application performance issues related to enabling SQL exception collection. The exception cache limit is 20 MB (0020).

You can allocate a storage cache by specifying any non-zero value for the cache size. The size specified is the number of 1 MB blocks allocated to the pool. This storage is allocated in the Xmanager private region when collection is started.

If you allocate a storage cache, you can monitor its storage use and effectiveness with a modify command. As a general guideline, specify an initial cache size of 2 MB for a typical production DB2 subsystem.

More information:

PSTAT(DTR) Command—Monitor the Exception Cache Pool (see page 57)

Integrated Data Sharing Statistics

Data sharing integration lets you view the collection statistics of a data sharing group's members as an integrated whole and comparatively. To be included in this integration, the individual member collections must use the Sysplex Interval as their collection interval. If the collection data is being externalized, the member collections must also use the same datastore name and high level qualifier. We recommend using the data sharing group name as the common datastore name for the individual member collections.

Note: If you start collections for your data sharing group members without adhering to these requirements, the collections will not be included in the integrated displays.

If your data sharing group members are active on multiple LPARs, make sure you have activated cross-system support when you start your Xmanager address spaces; otherwise, you cannot integrate data sharing member activity across all LPARs in the sysplex.

For instructions on enabling Xmanager cross-system support and preparing Xmanager, see the General Facilities Reference Guide.
Stop Collection

You can stop collection on a specific subsystem at any time.

**Follow these steps:**

1. Type **6** (Terminate SSID collection) in the Option line on the CA Detector Main Menu and press Enter.
   
   The Terminate Collection Display appears.

2. Specify the DB2 subsystem ID on which you want to stop collection, and press Enter.
   
   When you terminate collection, informational messages are written to the Xmanager joblog and the MVS syslog. These messages include the user ID of the person who stopped the collection. It can take a few moments for collection to quiesce and terminate. A message displays to verify that collection has been successfully stopped.

   **Note:** If an MVS modify command is used to stop collection, the value of the user ID is CONSOLE.

Datastores

The collector can externalize the data collected when you create a datastore. Activity data collection and SQL error activity collection do not require use of a datastore. However, data must be externalized to collect exception SQL activity data.

If you are a first-time user, we recommend that you start collection without creating a datastore to externalize the data. This method lets you familiarize yourself with startup options and the various panels without creating a datastore.

Data is externalized on an interval basis. You specify the collection interval time when you start the collector. When you create the datastore, you specify how many intervals you want to retain. The datastore is a wrap file; it retains the $n$ most current intervals, where $n$ is the maximum interval value you specified when creating the datastore.

When you start collection without externalizing to the datastore, the data collected is retained only for the current collection interval. When the current collection interval expires, the data is discarded and a new collection interval starts.
You can view activity saved in a datastore or from the current collection interval. All panels available to view the current collection interval are also available to view data captured in the datastore.

**Notes:**

- Time values are stored in raw system clock format, which is in Greenwich Mean Time (GMT). The operating system stores a value that provides the offset from GMT to the local time. When the raw clock value is converted to a formatted time and date, the offset converts the GMT system clock value to the local time. Depending on when you set the dates on your datastore interval, the clock value can appear to gain an hour when daylight savings time begins and lose an hour when standard time is reinstated. Once the intervals that predate the time change are cycled off the datastore, the clock value returns to normal.

- The space estimates in the following topics are for *uncompressed* record data. When activity and exception SQL data records are externalized, CA Detector compresses them, which usually reduces the actual used space by 50 percent to 70 percent compared to the estimates. Datastores are created for use on a specified DB2 subsystem. You cannot create a datastore for use on multiple SSIDs.

---

**Create a Datastore**

You can create a datastore to save collection data so that it can be viewed and analyzed at a later time.

**Follow these steps:**

1. **Do one** of the following:
   - Type 7 (Create/Initialize Datastore) in the Option line on the CA Detector Main Menu and press Enter.
   - Type Y in the Create Datastore field on the Start Collection Display and press Enter.

   The Create Datastore Display panel appears.

2. **Complete the following fields:**

   - **Datastore SSID**
     - Specifies the DB2 subsystem ID for which the datastore is being created.

   - **High Level**
     - Specifies the high-level qualifier for the datastore.

   - **Datastore name**
     - Specifies the name to be used for the datastore. The name must be unique for the DB2 subsystem ID and will be used as a part of the data set names created during the create datastore process.
Max intervals

Specifies the maximum number of collection intervals to be retained in the datastore. Data is externalized on a collection interval basis. It is written to the datastore when the collection interval expires. For example, if you specify 10, and if 10 intervals exist within the datastore, then the oldest interval is deleted when the next interval is inserted. The interval time used is specified when collection is started.

Exception SQL

Specifies whether exception SQL is included in the datastore.

SQL Error

Specifies whether SQL errors are to be included in the datastore.

Batch JCL

Specifies whether to run the JCL in batch.

Press Enter.

The Create Datastore Display/Standard Activity Collection Data Set panel appears.

3. Press Enter to process your selections.

A message appears to indicate successful creation of the datastore data set.

**Note:** If you specified Y in the Exception SQL or SQL Error fields, a panel appears for each option so that you can specify data set allocation criteria.

**More information:**

[Space Requirements](#) (see page 43)
Retain Exception SQL Data

You can create the datastore to include exception SQL activity collection data.

Follow these steps:

1. Do one of the following:
   - Type 7 (Create/Initialize Datastore) in the Option line on the CA Detector Main Menu and press Enter.
   - Type Y in the Create Datastore field on the Start Collection Display and press Enter.
   The Create Datastore Display panel appears.

2. Type Y in the Exception SQL field to include exception SQL activity collection data and press Enter.
   
   **Note:** You must also specify datastore name information, the maximum number of collection intervals to be retained in the datastore, and whether to collect data on SQL errors.

   The Create Datastore Display/Standard Activity Collection Dataset panel appears.

3. Specify the data set allocation criteria and press Enter.
   
   The Create Datastore Display appears. The datastore will contain a data set for collecting exception SQL activity.

   **Note:** If you specified Y for SQL Error to include SQL error activity in the collection, the Create Datastore Display/SQL Error Collection Dataset panel appears instead. Specify the data set allocation criteria as needed.

Datastore Creation Example

Create a datastore with the following attributes:

- The high-level qualifier is DSNMx.
- The datastore name is COLLD81A.
- The subsystem is D81A.

In addition, specify that you want the datastore to be able to collect exception SQL and SQL error data.

The following data set names are generated:

<table>
<thead>
<tr>
<th>Data Set Name</th>
<th>Type of Collection</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNMx.COLLD81A.DTR0001.D81A</td>
<td>Activity</td>
</tr>
<tr>
<td>DSNMx.COLLD81A.DTR0002.D81A</td>
<td>Exception SQL</td>
</tr>
</tbody>
</table>
### Datastores

#### Chapter 3: Using Collection Services

<table>
<thead>
<tr>
<th>Data Set Name</th>
<th>Type of Collection</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNMx.COLLD81A.DTRW002.D81A</td>
<td>Exception SQL (work file)</td>
</tr>
<tr>
<td>DSNMx.COLLD81A.DTR0003.D81A</td>
<td>SQL error</td>
</tr>
</tbody>
</table>

When collection is externalized, CA Detector attempts to retain the number of intervals in the datastore that were specified when the datastore was created. When the interval time limit has been reached, the newest interval is added and the oldest interval is deleted. The collection logic attempts to reuse space that had been used by deleted interval data.

### Space Requirements

Before you create a datastore, be sure that you have enough allocated space to hold the collected data. Review the size of the exception SQL collection and SQL error collection data sets to help you allocate space accurately for the datastore. You can allocate up to 999,999 cylinders.

**Important!** The estimated space requirements are for uncompressed record data. CA Detector compresses the activity and exception SQL data records when they are externalized. This compression typically reduces the actual space used for these records by 50 to 70 percent from the estimated requirements.

Accurately estimating the amount of space required to retain activity collection data depends on a number of factors. When estimating space requirements, consider the following:

- The number of collection intervals to retain in the data set. You supply this value when the data set is created.
- The interval time; that is, how often collections are externalized to the data set. You supply this value when collection is started.
- The number of unique plans executed during a collection interval.
- The number of unique plan and program combinations executed during a collection interval.
- The number of unique plan, program, and SQL statement combinations executed during a collection interval.
- The number of additional keys collected and the types of DB2 connections, such as batch or CICS, that the additional keys are collected for.
Whether you specify extended data collection for exception SQL or SQL error data collection, specify these options when you create the datastore.

**Note:** Predicting application workload activity is difficult at the varying degrees of granularity shown in the previous list. Consequently, we recommend setting allocation values when you create the datastore and then increase or decrease those values to optimize the data set allocation size. For example, start with a primary allocation of 100 cylinders and a secondary allocation of 50 cylinders. Evaluate the amount of space required to retain the required data.

**More information:**

[Create a Datastore](see page 40)

### Data Sets Created by the Datastore

The datastore used to retain data is a collection of VSAM data sets. The number of data sets created depends on the collection options you specify. When you create the datastore, a data set is automatically created to contain activity collection information. Additional data sets are created if you indicate that you want to collect exception SQL activity, SQL error activity, or both. This information can be helpful when you are deciding space allocations for the datastore.

When you create the datastore, VSAM linear data sets are generated with the following names:

`highlevel.datastore_name.DTRnnn.ssid`

**Note:** `highlevel`, `datastore_name`, and `ssid` are taken from information you specify on the Create Datastore Display.

`DTRnnnn` is an identifier for the data set. `nnnn` is a four-byte identifier:

- **0001**
  - Indicates an activity collection data set.
- **0002**
  - Indicates an exception SQL collection data set.
- **0003**
  - Indicates an SQL error collection data set.

If you create a datastore to contain exception SQL collection data, a datastore work file data set is also created. The work file is a VSAM ESDS data set with the following name:

`highlevel.datastore_name.DTRW002.ssid`
Data Set Records

The activity collection data set uses variable-length records. However, the majority of application related records are 800 bytes long. At the end of a collection interval, an 800-byte record is written for the following:

- Each unique plan name executed during the interval.
- Each program executed by each unique plan during the interval.
- Each SQL statement executed by each unique plan and program combination during the interval.
- Each unique program executed during the collection interval.
- Each plan that has executed the unique program during the collection interval.
- Each SQL statement executed by the unique program during the collection interval.
- If you have enabled dynamic SQL text-based statistics, each dynamic SQL statement executed by the unique program during the collection interval.

If you are using CA Detector with CA Subsystem Analyzer for the integrated panels, add 250 bytes for the following:

- Each table accessed by the unique program during the collection interval.
- Each index accessed by the unique program during the collection interval.

If you have enabled dynamic SQL text-based statistics, add sufficient space for all the dynamic SQL text executed during the collection interval.

Exception SQL Collection Space Requirements

When you create a datastore for exception SQL collection, two data sets are actually created: a collection datastore data set and a work file data set.

Exception SQL Data Set

Accurately estimating the amount of space required to retain exception SQL collection data depends on a number of factors. When estimating space requirements, consider the following:

- The number of collection intervals to be retained in the data set. You supply this value when the data set is created.
- The interval time; that is, how often collections are externalized to the data set. You supply this value when collection starts.
The collection profile definition. If you define your collection profile to capture the majority of exception SQL requests, you need more space than if you specify a few exception SQL requests to be captured.

The maximum number of exception SQL requests to be retained per collection interval. You specify this value in the global defaults.

It can be difficult to predict space requirements due to the previously stated factors and to the variable length of a record written for exception SQL activity. The length of a record written for an SQL request varies according to the type of activity generated by the SQL request and the length of the SQL text being generated. However, you can use an average length for an exception SQL request record of 4,000 bytes to help you in estimating your space requirements. You can then increase or decrease the allocation values to optimize the data set allocation size.

At the end of a collection interval, one record is written for each exception SQL request captured during the collection interval.

**Exception SQL Work File Data Set**

Space allocations specified for the work file data set should be adequate to contain the exception SQL data collected in a single collection interval. Specify a space allocation sufficient to hold the number of average size exception records that you have specified as the maximum exceptions per interval on the collection profile global defaults. Provide secondary space allocations to allow data set extents if required.

**SQL Error Collection Space Requirements**

Accurately estimating the amount of space required to retain SQL error collection data depends on a number of factors. When estimating space requirements, consider the following:

- The number of collection intervals to be retained in the data set. This value is supplied when you create the data set.
- The interval time; that is, how often collections are externalized to the data set. This value is supplied when you start collection.
- The collection profile definition. If you define your collection profile to exclude most SQL error codes, you need less space than if you capture most SQL error codes.
- The typical amount of SQL error activity that occurs on the subsystem.

The SQL error collection data set uses fixed length records of approximately 400 bytes. At the end of a collection interval, one 400-byte record is written for each SQL error collected during the collection interval.

In addition if you have specified SQL error text as a collection start option an additional record may be written to the SQL exception collection datasets to record the SQL text and host variables associated with the error. In this case, you should increase the amount allocated to these datasets by an additional five to 10 percent.
Exception SQL Text Table

The exception SQL text table contains the SQL text for those SQL statements identified as exception SQL. The SQL text column is limited to 254 characters, and anything beyond the 254-character limit is contained in another row, with the same START_TIME value and an incremented SEQNO value, up to 254 characters. In this way, the entire SQL statement can be stored in the table, allowing you to review the entire SQL query.

The PDTDYNT member of CDBASRC maps the unload data record that contains the data loaded into this table.

These records are only written to the SYSREC data set for REPORT=EXCP and HOSTVARS=Y reports.

**Note:** The SSID and START_TIME columns can be used to join the exception SQL table with the exception SQL text table.

Most of the DB2 columns have the same names and meanings as the columns shown on the product panels.

The following data columns differ in name from the online displays.

**INTERVAL_START**
- Specifies the date and time the collection interval started. This column is a standard DB2 timestamp format.

**INTERVAL_END**
- Specifies the date and time the collection interval ended. This column is a standard DB2 timestamp format.

**SEQNO**
- Specifies the sequence number of the SQL text segment. You can use the ORDER BY SEQNO clause to select all text segments for a SQL statement in sequence.

**SQL_TEXT**
- Specifies a 254-character portion of the SQL statement. If the SEQNO value is 1, it is the first 254 characters. If the SEQNO value is 2, it is the second 254 characters, and so on.

View Collection Status

You can view a list of active DB2 subsystems and data sharing groups and the collection status of each collection and group member. This includes the options in effect when collection was started, such as collection interval times, the type of data being collected, and whether the data is being externalized.
To view collection status, do either of the following from the CA Detector Main Menu:

- Type 3 in the Option line and press Enter.
  
  The DETECTOR Collection Status Display appears. You can select an active collection to view the options that were in effect when the collection was started.

- Type 11 in the Option line and press Enter.
  
  The Data Sharing Groups Display panel appears. You can select a group to set it as the default or view information about the members in the data sharing group. This includes the member status (if the member was selected for data sharing statistics integration). For members with active collections, you can view the options that were in effect when collection was started.

---

### Collection Control Using MVS Modify Commands

You can use the following MVS modify commands to perform the following tasks:

- START(DTR) to start collection
- STOP(DTR) to stop collection
- PROF(DTR) to load a collection profile
- PSTAT(DTR) to monitor exception cache use

This provides flexibility to start and stop collection through MVS console commands, an automated operation tool, or any other technique with which MVS modify commands can be issued.

**Note:** Use of the MVS modify commands to control collections is limited to DB2 subsystems that are executing on the same LPAR as the one from which the modify command is issued. Related error messages are issued in the form of WTO (write-to-operator) messages. Check the JES joblog or the MVS syslog for any WTO messages that indicate a successful or failed result.

---

### START(DTR) Command—Start Collection

The START(DTR) command initiates CA Detector collection using an MVS modify command to the Xmanager started task on a specific DB2 subsystem and appropriate keywords.

Only one CA Detector collection can be active for a selected DB2 subsystem. However, many CA Detector collectors can be active concurrently for multiple DB2 subsystems.

**Note:** You can specify all start collection options available to you from the Start Collection Display panels.
This command has the following format:

`F PTXMAN,START(DTR),DB2(ssid),AUTO(Y|A|N),CHV(Y|N),DST(datastore-name),
DUR(hhmm),DYN(Y|N),EXT(Y|N),HLVL(high-level),INT(hhmm),KEYS(Y|N),
PLEXINTV(Y|N),PLNX(Y|N),POOL(nnmm),PRF(profile-name),RND(Y|N),
SDE(Y|N),SQE(Y|N|O),SQLT(Y|N),SQT(Y|N),TRG(Y|N)`

The following keywords are required:

**PTXMAN**

(Required) Specifies the name of the Xmanager started task. PTXMAN is usually specified.

**START(DTR)**

Specifies the start keyword and related operand. Specify START(DTR) as the first keyword.

**DB2(ssid)**

(Required) Specifies the DB2 subsystem on which processing occurs.

The DB2 subsystem must reside on the same z/OS system as the one from which the modify command is issued.

The remaining keywords are optional and can be entered in any order:

**Note:** If you do not specify a keyword, it is the same as specifying the default value.

**AUTO(Y|A|N)**

Specifies whether to start or restart collection automatically if DB2 is not available when the command is issued or if DB2 terminates during collection.

Specify one of the following operands:

**Y**

Starts collection automatically when DB2 is available. If DB2 terminates during collection, the collection is suspended. Collection is restarted when DB2 is available. If collection is terminated manually, this setting is canceled.

**A**

Starts collection automatically even if the DB2 subsystem is not defined to the current LPAR. Typically, the collection start request would fail with an error message.

This option is useful when you want collection to start automatically when a subsystem that is executing on one LPAR moves to the current LPAR.
N
Indicates that collection is not started automatically. If DB2 is not available when the command is issued, the modify command terminates. Messages are written to the Xmanager joblog and SYSLOG indicating that DB2 is not available.

Default: N. No entry is the same as specifying N.

CHV(Y|N)
Specifies whether to start host variable collection with exception SQL data collection.
Specify one of the following operands:
Y
Collects host variable data. If Y is specified, also specify the PRF keyword and either DYN(Y) or SDE(Y) in the command syntax.
N
Indicates that host variable data is not collected.
Default: N. No entry is the same as specifying N.

DST(datastore-name)
Specifies the name of an existing data store to use for collection.
This keyword is required if EXT(YES) is specified.

Note: Use the same data store name for every member of a data sharing group that you want included in the group integrated data sharing statistics.

DUR(hhmm)
Specifies the amount of time in hours and minutes that the collection remains active.
No time limit is in effect when you specify 0000 or omit this keyword.
Limits: 0000 to 9999
DYN(Y|N)
Specifies whether data from dynamic SQL is collected when the SQL surpasses a threshold specified in a profile and becomes exception SQL.
Specify one of the following operands to indicate whether to start dynamic SQL request collection:

Y
Collects dynamic SQL data. If Y is specified, also specify the PRF keyword in the command syntax.

N
Indicates that dynamic SQL data is not collected.
Default: N. No entry is the same as specifying N.

EXT(Y|N)
Specifies whether to externalize (write) the collection data to a data store at the end of the collection interval. The data is collected on an interval basis.
Specify one of the following operands:

Y
Externalizes data at the end of the collection interval. When the collection interval ends, the collection interval data is reset to null and collection starts for the new interval. The data from the prior collection is written to a data store and is then available for viewing.
If Y is specified, also specify DST.

N
Indicates that collected data is not externalized. The data from the prior collection interval is discarded and only the current interval data is available for viewing.
Default: N. No entry is the same as specifying N.

HLVL(high-level)
Specifies the high-level qualifier of the data store. HLVL is also referred to as VCT.
This keyword is required if EXT(Y) is specified.
Note: Use the same high-level qualifier for every member of a data sharing group that you want included in the group integrated data sharing statistics.

INT(hhmm)
Specifies the collection interval to use in hours and minutes.
Limits: 0005 to 9999
Default: 0030
KEYS(Y|N)

Collects additional thread identification information to organize standard activity data by additional views, such as by userid or correlation id, in addition to the standard plan hierarchy view.

Specify one of the following operands:

Y

Indicates to enable additional monitoring for additional View By keys.

N

Indicates that additional monitoring for View By keys is not enabled.

Default: N. No entry is the same as specifying N.

PLEXINTV(Y|N)

Specifies whether to use the common Xmanager sysplex time interval as the collection time interval. Use of the sysplex interval facilitates coordination of collection activity between products across all LPARS in a sysplex.

Specify one of the following operands:

Y

Uses the common sysplex interval for the collection interval. Specify PLEXINTV(Y) for data sharing group members that you want to be included in the group integrated data sharing statistics.

N

Indicates to use the value of the INT keyword for the collection interval.

Default: N. No entry is the same as specifying N.

PLNX(Y|N)

Specifies whether to load the plan Exclude/Include list from the specified collection profile. The plan Exclude/Include list limits CA Detector monitoring to a specific set of plans that you identify.

Specify one of the following operands:

Y

Loads the plan Exclude/Include list. Depending on how you define the list, CA Detector collection is limited to the plans on the list or to all plans except the plans on the list.

N

Indicates that the plan exclusion list is not loaded.

Default: N. No entry is the same as specifying N.
POOL(nnnn)

Specifies the size of the exception collection storage cache pool. The size that is specified is the number of 1 MB blocks allocated to the pool. The storage is allocated in the Xmanager region private area.

Allocating cache storage can help reduce the application overhead that is associated with exception collection.

Limits: You can specify any positive integer up to 20 MB (0020) for the number of pool blocks. No exception cache storage is allocated when you specify a value of 0 or the POOL keyword is omitted.

PRF(profile-name)

Specifies the name of the collection profile to use. The operand is the profile ID.

Note: This keyword is required if you specify Y for DYN, SDE, CHV, SQE, or PLNX.

RND(Y|N)

Specifies whether to end the collection interval on a time boundary that is an integral multiple of the interval time.

If PLEXINTV(Y) is specified, the interval is always rounded.

Specify one of the following operands:

Y

Rounds the end of the interval to an integral boundary. You must also specify INT(Y).

N

Indicates that the collection interval time is not rounded.

Default: Y. No entry is the same as specifying Y.

SCA(Y|N|O)

Specifies whether to start SQL error collection.

Note: This option is synonymous with SQE. SCA is retained for compatibility with prior releases but can be removed in a future release. We recommend that you replace use of SCA with SQE.

Default: N. No entry is the same as specifying N.
SDE(Y|N)

Specifies whether data from static SQL is collected when the SQL surpasses a threshold specified in a profile and becomes exception SQL.

Specify one of the following operands to indicate whether to start static SQL request collection:

Y

Collects static SQL data. If Y is specified, also specify the PRF and DYN(Y) keywords in the command syntax.

N

Indicates that static SQL data is not collected.

**Default:** N. No entry is the same as specifying N.

SQE(Y|N|O)

Specifies whether to start SQL error collection.

**Note:** Specifying SQE is the same as specifying SCA.

Specify one of the following operands:

Y

Collects data on SQL error activity. If Y is specified, also specify the PRF keyword in the command syntax.

N

Indicates that SQL error data is not collected.

O

Standard and exception collections are not initiated. SQL text that is associated with negative SQL return codes is collected even if CA Detector does not record a performance exception for the SQL statement.

If SQE(O) is specified, also specify SQT(Y).

**Default:** N. No entry is the same as specifying N.
**SQLT(Y|N)**

Specifies whether to enable additional SQL text collection to collect the resource statistics that are associated with dynamic SQL statements.

Specify *one* of the following operands:

**Y**

Indicates to enable additional monitoring for dynamic SQL statements statistics collection.

**N**

Indicates that additional monitoring for dynamic SQL statements statistics is not enabled.

**Default:** N. No entry is the same as specifying N.

**SQT(Y|N)**

Specifies whether to enable additional SQL text collection for SQL error data.

Specify *one* of the following operands:

**Y**

Collects additional SQL text for SQL error data. If Y is specified, also specify SQE(Y) or SQE(O).

**N**

Indicates that additional SQL text collection for SQL error data is not enabled.

**Default:** N. No entry is the same as specifying N.

**TRG(Y|N)**

Specifies whether to collect triggered SQL statistics or SQL originating from native SQL stored procedures.

Specify *one* of the following operands:

**Y**

Collects triggered SQL statistics and SQL originating from native SQL stored procedures. Collecting triggered SQL statistics introduces a slight overhead. These figures do not include table and index data.

**N**

Indicates triggered SQL statistics and SQL originating from native SQL stored procedures are not collected.

If you do not use table triggers or you do not require separate statistics for triggered SQL, set this keyword to N.

**Default:** Y. No entry is the same as specifying Y.
**Example: Start Collection**

The following syntax illustrates some of the options you can specify to start collection using an MVS modify command:

```plaintext
F PTXMAN,START(DTR),DYN(Y),PRF(COLLDASN1),DB2(DSN1),INT(0015),
   EXT(Y),DST(DETECTOR),HLVL(PDT)
```

In this example, the command syntax performs the following tasks:

- Starts activity and exception SQL collection on DB2 subsystem DSN1 for the COLLDASN1 collection profile.
- Defines a collection interval of 15 minutes.
- Saves the collected data to a datastore named DETECTOR with the high-level qualifier PDT.

**STOP(DTR) Command—Stop Collection**

The STOP command terminates collection with MVS modify commands. You must specify the appropriate keywords to the Xmanager started task on the appropriate MVS system.

This command has the following format:

```plaintext
F PTXMAN,STOP(DTR),DB2(ssid)
```

- **PTXMAN** *(Required)* Specifies the name of the Xmanager started task. PTXMAN is usually specified.
- **STOP(DTR)** Indicates the stop keyword and related operand. Specify this as the first keyword, exactly as shown.
- **DB2(ssid)** Indicates the DB2 subsystem identifier keyword. The operand specifies the local DB2 subsystem identifier for which collection should be terminated.

**PROF(DTR) Command—Load a Collection Profile**

The PROF command refreshes the contents of the CA Detector collection profile or loads a new collection profile with an MVS modify command. This lets you customize profile controlled collection activity for varying workload requirements without having to stop and restart your current collection.
This command has the following syntax:

\[ \text{F PTXMAN, PROF(DTR), PRF(profile-name), DB2(ssid)} \]

**PTXMAN**

(Required) Specifies the name of the Xmanager started task. PTXMAN is usually specified.

**PROF(DTR)**

Indicates the profile keyword and related operand. Specify this as the first keyword, exactly as shown in the provided syntax.

**PRF(profile-name)**

Specifies the collection profile to be used.

This profile replaces the currently loaded collection profile.

**DB2(ssid)**

(Required) Specifies the DB2 subsystem on which processing occurs.

The DB2 subsystem must reside on the same z/OS system as the one from which the modify command is issued.

**Example: Replace the Current Collection Profile**

Specify the following syntax to replace the current collection profile contents with the collection profile PROFILE1 for the collection on DB2 subsystem DSN1:

\[ \text{F PTXMAN, PROF(DTR), PRF(PROFILE1), DB2(DSN1)} \]

**PSTAT(DTR) Command—Monitor the Exception Cache Pool**

The PSTAT command monitors the effectiveness of the exception storage cache with an MVS modify command. Periodically monitoring cache statistics helps you to size the cache pool for maximum effectiveness. You can monitor cache statistics by issuing the command to the Xmanager started task on the appropriate MVS system.

When you are first using the exception storage cache for a DB2 subsystem you should periodically monitor the cache statistics to determine if you have allocated sufficient storage. If the average percent of cache hits in the PDT0525 message is consistently below 95% you may want to increase the initial cache allocation.

This command has the following format:

\[ \text{F PTXMAN, PSTAT(DTR), DB2(ssid)} \]

**Note:** This command produces cache statistics messages PDT0525 for the collection on the specified DB2 subsystem.
PSTAT(DTR)

Indicates the pool statistics keyword and related operand. Specify this as the first keyword, exactly as shown in the previous illustration.

DB2(ssid)

Indicates the DB2 subsystem identifier keyword. The operand specifies the local DB2 subsystem identifier for which cache statistics should be displayed.

More information:

Collection Profiles (see page 137)

Modify RUN Command

The Xmanager modify commands used to start and stop collection can be executed by placing the commands in an Xmanager parmlib member and then issuing the RUN modify command.

This method is useful when you want to automate the use of modify commands. It is also useful when the parameters for the modify command exceed the maximum allowable length for an MVS console command, JCL, or other technique.

The modify RUN command syntax is as follows:

RUN(member)

member

Specifies the name of the Xmanager parmlib member containing the modify commands you want to issue to Xmanager.

How to Create the Xmanager Parmlib Member

Create a new member in the Xmanager CDBAPXMP PDS using ISPF Edit or another appropriate method. The member name should not begin with D because the members that CA Detector creates in this PDS begin with D.

Edit the member so that it contains the appropriate modify commands to be issued to Xmanager. The commands must be in the format previously described, such as START(DTR), and so on. Do not use the full MVS command structure. For example, assuming that PTXMAN is the name of the Xmanager started task, the F PTXMAN portion of the command is not required.

All commands must be preceded with a hyphen (-) followed immediately by the command. The command can be split across lines if necessary. The hyphen indicates the start of the next command.
CA Detector uses columns 1 - 72 for values but ignores columns 73 - 80. If your values exceed 72 columns, move on to the next line and continue there. For example, you can enter the following into a member:

-START(DTR), DYN(Y), PRF(RQPAP310), DB2(DB2F), DST(RQPAP310), HLVL(OLS), SCA(Y), EXT(Y), INT(0010), DUR(0010)

When this member is executed using a RUN command it starts the collection on subsystem DB2F.

**Issue the RUN Command**

To issue the modify command you have entered in the Xmanager parmlib member, use the RUN command, as shown in the following example:

```
F PTXMAN, RUN(MODIFY01)
```

This example assumes the parmlib member is MODIFY01, and that PTXMAN is the name of the Xmanager started task. The command causes Xmanager to find each command in MODIFY01 that starts with a hyphen and execute it as if it were issued by a modify command. For example, if MODIFY01 contained the -START(DTR) command as previously illustrated, collection starts.

**How To Synchronize Collection**

CA Detector and CA Subsystem Analyzer work together to provide you with detailed information you can use to effectively tune your DB2 subsystems and applications. By synchronizing collection, you can easily access related information between the two products. For example, you can use CA Subsystem Analyzer to identify a heavily accessed table. If you synchronize collection, you can select an SQL option on the CA Subsystem Analyzer panel and jump to CA Detector to view all the SQL statements that have accessed the table.

To relate the collection data between the products, the intervals must be aligned. The collection that starts first sets the interval values that are used. When collection for the other product starts on the same subsystem, it uses the values that are already in use, regardless of what is specified. For example, if you start CA Subsystem Analyzer collection on subsystem D81A with an interval time of one hour and then start CA Detector collection on subsystem D81A with an interval time of 30 minutes, the CA Detector collection interval is set to one hour.
You should start collection for both products in approximately the same time frame. Otherwise, the data for the second product may not be complete. For example, if you start collection for CA Detector and specify a two-hour interval and then one hour later you start collection for CA Subsystem Analyzer, the first CA Subsystem Analyzer collection data only contains one hour of data compared to two hours of CA Detector data. When the CA Detector interval is complete, the data collected for both products is reset and a new collection interval begins for both products. If collection stops for one product, when it resumes it synchronized with the product that remained running.

To synchronize collection when you do not externalize collection data, start collection in both products and the interval values set when the first collection starts will be used by both collections. To synchronize collection when you externalize data, specify the same datastore names in both products when you start collection. The interval criteria set in the first collection is automatically used. If you use different datastore names, the collections will not synchronize and you can use independent interval times. If you do not synchronize collection, you may not be able to access related data between the products.

**Note:** The Time Limit value can be set independently in each product regardless of synchronization.

**Relating Collection Data**

To relate collection data between CA Detector and CA Subsystem Analyzer, the collection intervals must be aligned. When you synchronize, the collection that starts first sets the interval values that will be used. When collection for the other product starts on the same subsystem, it will automatically use the values already in use, regardless of what is specified.

For example, if you start CA Subsystem Analyzer collection on subsystem DB2A with an interval time of one hour and then start CA Detector collection on subsystem DB2A with an interval time of 30 minutes, the CA Detector collection interval is automatically set to one hour.

**Synchronization**

Whether synchronization occurs is determined by the specification of certain criteria when starting collection in both products.

When you do not externalize collection data, synchronization is automatic; start collection in both products and the interval values set when the first collection starts will be used by both collections.
When you externalize collection data, synchronization is optional and is controlled by the datastore names used. If you use the same names in both products, the collections will synchronize; if you use different names, the collections will not synchronize and you can use independent interval times.

Note, however, that if you do not synchronize collection, you may not be able to access related data between the products. To synchronize, prior to starting collection, use the respective Create Datastore Display in each product to create datastores with identical high-level qualifiers and datastore names. When starting collection in the first product, specify the interval criteria and the datastore name that will be used by both collections. When starting the second collection, make sure to specify the same datastore name used in the first product.

**Note:** The Time Limit value can be set independently in each product regardless of synchronization.

**Considerations for Synchronizing Data**

If you are synchronizing data, you should start collection for the two products in approximately the same time frame. Otherwise, the data for the second product may not be complete.

For example, you can start collection for CA Detector and specify a two-hour interval. Then one hour later you start collection for CA Subsystem Analyzer. Collection for CA Subsystem Analyzer will automatically be set to two hours; however, for the first interval, only one hour of data will be collected compared to two hours of CA Detector data. When the CA Detector two-hour interval is up, the data collected for both products is reset; then a new collection interval begins for both products.

If collection is stopped for one of the products, when collection resumes it will synchronize with the product that remained running.
Chapter 4: Viewing Collection Activity

This section contains the following topics:

- **Collection Activity** (see page 63)
- **Current and Historical Activity** (see page 64)
- **View Standard Collection Activity** (see page 64)
- **View Exception SQL Collection Activity** (see page 93)
- **View SQL Errors Collection Data** (see page 106)
- **View Dynamic SQL Errors** (see page 115)
- **View SQL Call Text for Static SQL Statements** (see page 115)
- **View DB2 Data Sharing Group Activity** (see page 116)
- **CA Subsystem Analyzer Panel Integration** (see page 122)
- **View Historical Collection Data** (see page 131)
- **View Historical Data Sharing Collection Data** (see page 132)
- **View Audit Records** (see page 134)

**Collection Activity**

After you start collection to capture application information over a period of time, you can identify heavily used components within applications, understand overall DB2 workload characteristics, and pinpoint and resolve performance problems deeply embedded within your application activity. Resource intensive DB2 performance traces are not required. DB2 accounting data is used as the source for collection. Because of this approach, you can better understand your DB2 application workload characteristics.

You can view information from both real time and historical perspectives and for data sharing groups for the following types of collection activity:

- DB2 subsystem activity (standard)
- Exception SQL activity
- SQL error activity
- Additional keys collection data
- Historical collection activity
- DB2 data sharing group activity
- CA Subsystem Analyzer synchronized collection activity
Current and Historical Activity

You can view application workload and application performance information with various levels of granularity from both real-time and historical perspectives.

CA Detector provides a unique, informative, and cost effective approach to understanding DB2 application use and resource consumption. It gives you the ability to view accounting data values from application, plan name, package, DBRM, and SQL statement perspectives. You can capture application information for long periods of time, then identify heavily used components within applications, understand overall DB2 workload characteristics, and pinpoint and resolve performance bottlenecks deeply embedded within your application activity.

Collection and display logic does not require you to start resource-intensive DB2 performance traces. It uses DB2 accounting data as the source of its collection; because of this unprecedented approach you discover and understand your DB2 application workload characteristics like never before.

You can also view similar current and historical displays for exception SQL activity. All major accounting trace performance information is collected at the SQL call level. In addition, application SQL error activity can be evaluated using current and historical displays. Application errors incurred as a result of abnormal SQL call return codes can be viewed and evaluated.

View Standard Collection Activity

The CA Detector standard collection displays provide a unique, informative, and cost effective approach to understanding DB2 application use and resource consumption.

You can view accounting data values from application, plan name, package/DBRM, and SQL statement perspectives without resource-intensive DB2 performance traces. Instead, we use DB2 accounting data as a source of collection data.

You can view collection data from a real time (current) and historical perspective as follows:

- By plan name
- By DBRM/package name
- By SQL call
- By dynamic SQL text
By reporting profile

By additional optional groups (keys)

Note: The information gathered is the same for current and historical activity, except for the fields defining the collection interval. For current activity, the interval collection fields are Interval Time and Interval Elapsed. For historical activity, the collection interval fields are Interval Date, Interval Time, and Elapsed Time.

Follow these steps:

1. Do either of the following on the CA Detector Main Menu:
   - Type 1 (SSID current interval data) in the Option field and press Enter.
     The Planname Summary Display panel appears.
     Note: This panel also appears when you type A in the View Type field on any current display panel.
   - Type 9 in the Option field and press Enter.
     The DETECTOR DS Group Planname Summary Display panel appears. Data sharing group application activity is provided.
     Note: Use the View By field to sort the collection activity by plan, program, SQL call, dynamic SQL text, profile, and other groupings (keys) as applicable.

2. Specify one of the following line commands next to a plan:
   - S
     Shows the next level of granularity for the selected line.
   - D (Detail)
     Shows a detail report of the column data for the selected line. The detail report shows the same column data but in a report format, which lets you view all column data with less scrolling.
   - Q (Dynamic SQL)
     Shows the executed dynamic SQL statements for the selected line. Not applicable for data sharing groups.
   - H (Plan History)
     Shows a history report of the column data for the selected line. Not applicable for data sharing groups.
View Standard Collection Activity

T (Active Threads)
Shows active thread collection data for the selected line. Not applicable for data sharing groups.

M (Member Activity)
Lists interval data for data sharing group members that were used to formulate the data on the selected line.

Press Enter.
The collection activity panel appears for the specified command.

More information:
View Exception SQL Collection Activity (see page 93)
View Historical Collection Data (see page 131)

View Collection Activity by Plan Name

You can view application activity from a plan name perspective. Application activity and resource use for the interval being viewed are summarized by plan name. You can use this information to identify which plans are most frequently used within the DB2 subsystem and analyze resource use by plan name. You can see which plans are consuming a high-percentage of all DB2 resources during the collection interval and why it is occurring. You can also view exception SQL and SQL error collection data, and view related keys information. Plans with currently active threads are highlighted.

You can select a plan and perform the following tasks:

■ View collection statistics by DBRM/package name
■ View plan information
■ View dynamic SQL activity
■ View related keys
■ View application activity history
■ View active thread collection data
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and S (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection. Use the Total/Avg field to show totals or averages in the data columns.

   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Specify one of the following line commands next to a plan:

   **S (Programs)**
   
   Shows program activity originating from a plan on the Plan Program Display panel. From this panel, you can view and evaluate activity and resource use from a DBRM/package point of view.

   **D (Detail)**
   
   Shows the currently displayed plan collection data in a report format on the Planname Summary Detail Display panel. The fields are grouped by plan information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   **Q (Dynamic SQL)**
   
   Shows dynamic SQL statements executed by a plan on the Plan Dynamic SQL Display panel. From this panel, you can select a dynamic SQL statement and view the plan related statistics and the complete dynamic SQL statement text. You can also evaluate access path information if you have a license for CA Plan Analyzer.

   **K (Keys)**
   
   Shows plan related keys data on the Planname Key Display panel. The keys are identified by type, such as DB2 connection user ID (AUTHID), correlation ID (such as batch job name or CICS transaction name), and so on.

   **H (Plan History)**
   
   Shows the plan application history activity on the Plan History Display panel. From this panel, you can identify the activity for a plan in the DB2 subsystem being viewed and examine resource use by the plan. Application activity and resource use for the plan are summarized for time intervals.

   **T (Active Threads)**
   
   Shows active thread collection data for the selected plan.

   Press Enter.

   The plan-related collection activity panel appears for the specified command.
View Plan Collection Activity by Program

You can view program (DBRM or package) activity originating from a plan. Plan resource use and SQL activity can be evaluated from a DBRM/package point of view. This helps you identify which programs account for most plan activity and resource consumption within the plan. You can view all major accounting class data values at the program level.

You can select a program and perform the following tasks:

- View the SQL activity originating from the program
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View program information

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.

2. Type S next to a plan and press Enter.
   The Plan Program Display panel appears. Use the Total/Avg field to show totals or averages in the data columns. Plan resource use and SQL activity is summarized by program name.

3. Specify one of the following line commands next to a program:
   
   **S (SQL stmts)**
   Shows SQL call activity and resource information on the Plan SQL Display panel.
   From this panel, you can
   
   **Q (Dynamic SQL)**
   Shows dynamic SQL activity on the Pgm Dynamic SQL Display panel.
E (Explain)
Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.

D (Detail)
Shows collection data for the program/plan in a report format on the Plan Program Detail Display panel. This lets you view the column data with less scrolling. The fields are grouped by program information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, and program identification information.

Press Enter.
The program/plan related collection activity panel appears for the specified command.

View Plan Dynamic SQL Activity
You can view the dynamic SQL activity for plans to easily identify which dynamic SQL statements have been executed.

You can select a dynamic SQL statement and perform the following tasks:
- View statistics relative to a plan.
- View the complete dynamic SQL statement text.
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View the referenced tables and indexes (a valid CA Subsystem Analyzer license is required).

Follow these steps:
1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.
2. Type Q (Dynamic SQL) next to a plan and press Enter.
   The Plan Dynamic SQL Display panel appears.
3. Specify one of the following line commands next to a dynamic SQL statement text:

**D (Detail)**

Shows dynamic SQL statement text information in a report format on the resulting display. This lets you view the column data with less scrolling. The fields are grouped by SQL statement information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

**E (Explain)**

Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.

**Q (SQL text)**

Shows the SQL call text for the dynamic SQL statement on the Dynamic SQL Call Text Display panel. You can pass the SQL call text to CA Plan Analyzer for Quick Explain or to interactive SQL (ISQL) for edit and execution. (Valid CA Plan Analyzer licensing is assumed for Quick Explain.)

**T (Tables/indexes)**

Shows tables and indexes referenced by the dynamic SQL statement on the Dyn SQL Table Activity Display panel. Valid CA Subsystem Analyzer licensing is required.

Press Enter.

The dynamic SQL statement related plan activity panel appears for the specified command.

**View Plan Related Keys**

You can view the keys related to a plan. The keys are identified by type such as user ID and correlation ID, number of commits, thread aborts, SQL calls, and the percentage of total interval or program INDB2 time.

You can select a key and view the collected data in a report format by key information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing locking activity, parallelism accounting data, wait times and counts, and additional time values.

**Note:** You must have enabled Additional View By Keys on CA Detector collection start and specified a collection profile to view keys data.
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.

2. Type K next to a plan and press Enter.
   The Planname Key Display panel appears. Use the Total/Avg field to show totals or averages in the data columns. Plan key related data is summarized by key.

3. Type D next to a key to display the collection data in a report format and press Enter.
   The Planname Key Detail Display panel appears.

4. Scroll the data as needed using F7 and F8 and press F3 to exit.

View Plan History

You can view the history of application activity for a plan. Application activity and resource use for plans are summarized by time intervals. You can identify the activity for a plan in the DB2 subsystem and examine resource use by the plan.

You can select a time interval and perform the following tasks:

- View program activity originating from a plan
- View interval information

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.

2. Type H next to a plan and press Enter.
   The Plan History Display panel appears.
View Standard Collection Activity

3. Specify one of the following line commands next to a plan:

   **S (View Programs)**
   Shows collection data for the program on the Plan Program Display panel.

   **D (View Detail)**
   Shows collection data for the time interval in a report format on the Plan History Detail Display panel. This lets you view the column data from the Plan History Display panel with less scrolling. The fields are grouped by interval information, statement execution data, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   Press Enter.
   The plan related history panel appears for the specified command.

View Plan Active Thread Collection Activity

You can view active thread collection data for DB2 subsystems.

**Follow these steps:**

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears.

2. Type **T** next to a plan and press Enter.

   The Thread Terminator Active Threads Display panel appears.

3. Specify one of the following line commands next to a plan:

   **S (View Thread Detail)**
   Shows detailed time related information about a currently active thread to help you analyze thread elapsed time, INDB2 time, and CPU use. Wait analysis information is also provided.

   All Class 2 times are only available if Accounting Class 2 is active on the subsystem being viewed. All Class 3 times are only available if Accounting Class 3 is active on the subsystem being viewed.

   **T (Terminate Thread)**
   Selects a thread for termination.

   Press Enter.
   The thread related panel appears for the selected plan.
View Collection Activity by Program

You can evaluate application performance from an application program (DBRM or package name). All major accounting data values can be viewed for programs.

You can select a program and perform the following tasks:

- View SQL statements
- View all DB2 plans that have used the DBRM or package
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View program information
- View SQL calls originating from the program
- View program history

**Note:** For data sharing groups, you can view SQL statements and additional program history, and invoke CA Plan Analyzer to evaluate access path information. You can also view interval data for data sharing group members.

**Follow these steps:**

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   
   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection.
   
   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Type G in the View By field and press Enter.
   
   The Package/DBRM Summary Display panel appears. Program information is summarized by application DBRM/package name. Use the Total/Avg field to show totals or averages in the data columns.

3. Specify one of the following line commands next to a program:
   
   **S (SQL)**
   
   Shows SQL statements for the selected DBRM or package on the Package/DBRM SQL Display panel.

   **P (Plans)**
   
   Shows collection data for all plans that have executed the selected DBRM or package on the Package/DBRM Plan Display panel.

   **E (Explain)**
   
   Invokes CA Plan Analyzer Quick Explain to evaluate access path information for the selected program. Valid CA Plan Analyzer licensing is required.
D (Detail)
Shows collection data for the program in a report format on the Package/DBRM Summary Detail Display panel. This lets you view the column data with less scrolling. The fields are grouped by program information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, and program identification information.

Q (Dynamic SQL)
Shows dynamic SQL statements executed by the selected program on the Pgm Dynamic SQL Display panel.

K (Keys)
Shows program related keys data on the Package/DBRM Key Display panel. The keys are identified by type, such as DB2 connection user ID (AUTHID), correlation ID (such as batch job name or CICS transaction name), and so on.

Note: You must have enabled Additional View By Keys on CA Detector collection start and specified a collection profile to view keys data.

H (Program History)
Shows the history of application activity for the selected program on the Package/DBRM History Display panel.

Press Enter.
The program related collection activity panel appears for the specified command.

View Program SQL Statements
You can view SQL statements for DBRMs or packages. Output is summarized by SQL statement. All major accounting class data values are available.

You can select an SQL call and perform the following tasks:
- View static and dynamic SQL call text.
- View the referenced tables and indexes (a valid CA Subsystem Analyzer license is required)
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View SQL information in a report format
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection.

   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Type G in the View By field and press Enter.

   The Package/DBRM Summary Display panel appears.

3. Type S (SQL) next to a program and press Enter.

   The Package/DBRM SQL Display panel appears. From this panel, you can determine program SQL call use and resource consumption at the SQL call level. Information is provided for all plan users of the program.

4. Specify one of the following line commands next to an SQL call:

   **Q (View SQL Text)**

   Shows the SQL call text for the selected SQL call on the SQL Call Dynamic SQL Display panel. Use this option to view the dynamic SQL activity for the selected prepare or execute immediate SQL statement.

   **T (View Tables/Indexes)**

   Shows tables and indexes that the SQL call referenced on the SQL Table Display panel. Valid CA Subsystem Analyzer licensing is required. Use this option to view all tables that have been referenced by the statement and the associated activity.

   **E (Explain)**

   Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.

   **D (Detail)**

   Shows SQL calls issued during the collection interval in a report format on the Package/DBRM SQL Detail panel. The fields are grouped by SQL information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, and plan/program ID values.

   Press Enter.

   The SQL call related panel appears for the specified command.
You can view all plans that have executed a specific DBRM or package. You can also determine which plans are the largest users of a selected program.

You can select a plan and perform the following tasks:

- View plan information
- View the SQL activity originating from the plan

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Pllannename Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection.

   Note: This panel also appears when you type A in the View Type field on any current display panel.

2. Type G in the View By field and press Enter.

   The Package/DBRM Summary Display panel appears.

3. Type P (Plans) next to a program and press Enter.

   The Package/DBRM Plan Display panel appears.

   Note: This panel also appears when you type P on the Package/DBRM History Display panel and type P next to an SQL call on the SQL Statement Summary Display panel.

4. Specify one of the following line commands next to a plan:

   **D (Detail)**

   Shows collection data for the plan in a report format on the Package/DBRM Plan Detail Display panel. This lets you view the column data from the Package/DBRM Plan Display panel with less scrolling. The fields are grouped by plan information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, and program identification information.

   **Q (Dynamic SQL)**

   Shows dynamic SQL statements executed by the selected program on the Plan Dynamic SQL Display panel.

Press Enter.

The plan related collection activity panel for the program appears for the specified command.
Evaluate Access Path Information

You can invoke CA Plan Analyzer Quick Explain to evaluate access path information. Valid CA Plan Analyzer licensing is assumed.

Quick Explain is only available for DB2 subsystems on the local z/OS or that are members of a data sharing group with an active group member on the local z/OS.

To evaluate access path information, type E next to a program or SQL call on any of the following panels:

- Plan Program Display
- Package/DBRM Summary Display
- SQL Statement Summary Display
- Dynamic SQL Summary Display
- Key Plan/Program Display
- Key Package/DBRM Display

Press Enter.

The PPA Quick Explain panel appears. From this panel, you can explain the following SQL source types:

- Plan
- Plan/DBRM combination
- Collection ID
- Collection ID/package combination

View Program Dynamic SQL Activity

You can view the dynamic SQL activity for a package or DBRM to easily identify which dynamic SQL statements have been executed.

You can select a dynamic SQL statement and perform the following tasks:

- View statistics relative to a program.
- View the complete dynamic SQL statement text.
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View the referenced tables and indexes (a valid CA Subsystem Analyzer license is required).
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

The Planname Summary Display panel appears.

2. Type G (program) in the View By field and press Enter.

The Package/DBRM Summary Display panel appears.

3. Type Q (Dynamic SQL) next to a program and press Enter.

The Pgm Dynamic SQL Display panel appears. The dynamic SQL statements that have been executed from the selected program appear. You can evaluate program activity and resource use from the dynamic SQL point of view.

   **Note:** This panel also appears when you type G next to an SQL statement on the Dynamic SQL Summary Display panel and press Enter.

4. Specify one of the following line commands next to a dynamic SQL statement text:

   - **D (Detail)**
     
     Shows dynamic SQL statement text information in a report format on the DETECTOR Pgm Dynamic SQL Detail Display panel. This lets you view the column data with less scrolling. The fields are grouped by SQL statement information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   - **E (Explain)**
     
     Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.

   - **Q (Dynamic SQL)**
     
     Shows the SQL call text for dynamic SQL activity on the Pgm Dynamic SQL Display panel.

   - **T (Tables/indexes)**
     
     Shows tables and indexes referenced by the dynamic SQL statement on the Dyn SQL Table Activity Display panel. Valid CA Subsystem Analyzer licensing is required.

   Press Enter.

The dynamic SQL statement related program activity panel appears for the specified command.
View Program Related Keys

You can view the keys related to a program. The keys are identified by type such as user ID and correlation ID, number of commits, thread aborts, SQL calls, and the percentage of total interval or program INDB2 time.

You can select a key and view the collected data in a report format by key information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing locking activity, parallelism accounting data, wait times and counts, and additional time values.

Note: You must have enabled Additional View By Keys on CA Detector collection start and specified a collection profile to view keys data.

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.
2. Type G (program) in the View By field and press Enter.
   The Package/DBRM Summary Display panel appears.
3. Type K (Keys) next to a program and press Enter.
   The Package/DBRM Key Display panel appears.
4. Type D next to a key to display the collection data in a report format and press Enter.
   The Package/DBRM Key Detail Display panel appears. The information is grouped by the type of data collected (key information, buffer manager activity, and so on).
5. Scroll the data as needed using F7 and F8 and press F3 to exit.

View Program History

You can view the history of application activity for a program. This lets you identify the activity for a program in a DB2 subsystem and examine resource use by the program during the collection interval.

You can select a time interval and perform the following tasks:

- Determine program SQL call use and resource consumption at the SQL call level
- View all plans that have executed a selected DBRM or package
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View interval information
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.

2. Type G in the View By field and press Enter.
   The Package/DBRM Summary Display panel appears.

3. Type H (Program History) next to a program and press Enter.
   The Package/DBRM History Display panel appears. Application activity and resource use for the program are summarized for time intervals.

4. Specify one of the following line commands next to a time interval:
   - **S (View SQL stmts)**
     Shows SQL statements for the selected DBRM or package on the Package/DBRM SQL Display panel.
   - **P (View Plans)**
     Shows collection data for all plans that have executed the DBRM or package on the Package/DBRM Plan Display panel.
   - **E (Explain)**
     Invokes CA Plan Analyzer Quick Explain to evaluate access path information for the selected time interval. Valid CA Plan Analyzer licensing is required.
   - **D (View Detail)**
     Shows collection data for the time interval in a report format on the Package/DBRM History Detail panel. The fields are grouped by interval information, statement execution data, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.

The program history related collection activity panel appears for the specified command.

View SQL Calls

You can view SQL calls issued during the collection interval.

**Note:** You must set the SQL collection and exception options when you start CA Detector collection.
You can select an SQL call and perform the following tasks:

- View SQL call text (for dynamic and static calls)
- View collection data for all plans that have executed the selected SQL call
- View tables and indexes that the SQL call referenced (valid CA Subsystem Analyzer licensing is required)
- Invoke Quick Explain (valid CA Plan Analyzer licensing is required) and ISQL
- View SQL information in a report format
- View SQL call history

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection. Use the Total/Avg field to show totals or averages in the data columns.

   Note: This panel also appears when you type A in the View Type field on any current display panel.

2. Type S (SQL) in the View By field and press Enter.

   The SQL Statement Summary Display panel appears. Use the Total/Avg field to show totals or averages in the data columns. Use the INDB2_CPU and INDB2_Time fields to show only those SQL requests that exceed the specified time.

3. Specify one of the following line commands next to an SQL call:

   **Q (SQL Text)**
   
   Shows the SQL call text for the call on the SQL Call Text Display panel for static SQL calls and on the SQL Call Dynamic SQL Display panel for dynamic SQL calls.

   **P (Plans)**
   
   Shows collection data for all plans that have executed the selected SQL call on the Package/DBRM Plan Display panel.

   **T (Tables)**
   
   Shows tables and indexes that the SQL call referenced on the SQL Table Display panel. Valid CA Subsystem Analyzer licensing is required.

   **E (Explain)**
   
   Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.
View Standard Collection Activity

D (Detail)
Shows SQL calls issued during the collection interval in a report format on the SQL Statement Detail Display panel. This lets you view the column data from the SQL Statement Summary Display panel with less scrolling. The fields are grouped by SQL information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, and plan/program ID values.

H (SQL History)
Shows the history of activity for an SQL call on the SQL Statement History Display panel.

Press Enter.
The SQL call related panel appears for the specified command.

View Static SQL Call Text
You can view SQL call text originating from a static SQL application or that has exceeded at least one of your thresholds and become exception SQL for dynamic SQL. For example, if you are viewing an SQL OPEN CURSOR statement, you can view the SQL call text of the related SQL DECLARE CURSOR. You can then pass the SQL call text to CA Plan Analyzer for access path analysis or to ISQL to interactively edit and execute the SQL call text. DB2 catalog access is not required so catalog lock contention is not a concern.

You can view the SQL text for static SQL and perform the following tasks:

- Pass the SQL call catalog information to Quick Explain (valid CA Plan Analyzer licensing is required)
- Pass the SQL call text to Quick Explain for editing (valid CA Plan Analyzer licensing is required)
- Pass the SQL call text to ISQL and interactively edit and execute the SQL text

Follow these steps:

1. Type Q next to an SQL call on any of the following panels:
   - Plan SQL Display
   - Package/DBRM SQL Display
   - SQL Statement Summary Display
   - Key Pkg/DBRM SQL Display
   - SQL error panels
Press Enter.
The SQL Call Text Display panel appears.
2. Specify one of the following line commands next to an SQL call:

**E - Explain SQL**

Passes the SQL call catalog information to the Quick Explain component of CA Plan Analyzer. This is the default explain action for static SQL statements and should be used if the SQL call text contains host variable markers. Valid CA Plan Analyzer licensing is required.

**T - Explain Text**

Passes the SQL call text to the Quick Explain component of CA Plan Analyzer and lets you edit the call text within CA Plan Analyzer. Valid CA Plan Analyzer licensing is required.

**I - ISQL**

Passes the SQL call text to ISQL to interactively edit and execute the SQL text.

*Note:* For more information about ISQL, see the *CA Database Management Solutions for DB2 for z/OS Value Pack Reference Guide*.

Press Enter.

*Note:* These options are only available for the DB2 subsystems on the local z/OS system or that are members of a data sharing group with an active group member on the local z/OS system.

The related SQL call text panel appears.

**View Dynamic SQL Call Text**

You can view the SQL text for a dynamic SQL statement and invoke:

- CA Plan Analyzer Quick Explain (a valid CA Plan Analyzer license is required) to evaluate access path information
- Interactive SQL (ISQL) to execute SQL statements contained in a data set

These options are only available on the local z/OS subsystem or that are members of a data sharing group with an active group member on the local z/OS subsystem. ISQL is a value pack component and is installed automatically with any CA Database Management Solutions for DB2 for z/OS product installation.

*Note:* For more information about ISQL, see the *CA Database Management Solutions for DB2 for z/OS Value Pack Reference Guide*. 
View Standard Collection Activity

Follow these steps:

1. Type Q next to an SQL call on any of the following panels:
   - Plan Dynamic SQL Display
   - Pgm Dynamic SQL Display
   - Key Dynamic SQL Display

   Press Enter.

   The Dynamic SQL Call Text Display panel appears.

2. Specify one of the following line commands next to an SQL call:

   **E - Explain SQL**
   
   Passes the SQL call text information to the Quick Explain component of CA Plan Analyzer. Valid CA Plan Analyzer licensing is required.

   **I - ISQL**
   
   Passes the SQL call text to ISQL to interactively edit and execute the SQL text.

   **Note:** For more information about ISQL, see the *CA Database Management Solutions for DB2 for z/OS Value Pack Reference Guide*.

   Press Enter.

   **Note:** These options are only available for the DB2 subsystems on the local z/OS system or that are members of a data sharing group with an active group member on the local z/OS system.

   The related SQL call text panel appears.

View SQL Call Related Plans

You can view collection data for all the plans that have executed SQL calls.

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection. Use the Total/Avg field to show totals or averages in the data columns.

   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Type S (SQL) in the View By field and press Enter.

   The SQL Statement Summary Display panel appears.
3. Type P next to an SQL call and press Enter.
   The Package/DBRM Plan Display panel appears.

4. Specify one of the following line commands next to a plan name:
   
   **D (Detail)**
   Shows all plans that have executed a DBRM or package in a report format. The fields are grouped by plan information, buffer manager activity, and so on.

   **Q (Dynamic SQL)**
   Shows dynamic SQL statements executed by the selected plan.
   Press Enter.
   The plan related SQL call panel appears for the specified command.

**View SQL Call Related Tables and Indexes**

You can view tables and indexes referenced by SQL calls.

**Note:** Valid CA Subsystem Analyzer licensing is assumed.

**Follow these steps:**

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection. Use the Total/Avg field to show totals or averages in the data columns.
   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Type S (SQL) in the View By field and press Enter.
   The SQL Statement Summary Display panel appears.

3. Type T next to an SQL call and press Enter.
   The SQL Table Display panel appears. This panel shows all tables that have been referenced by an SQL statement and what kind of activity is associated with those tables.
View SQL History

You can view the history of activity for an SQL call. Activity and resource use for the SQL call are summarized for time intervals. You can easily identify the activity for an SQL call in the DB2 subsystem being viewed and examine resource use by the SQL call.

You can select a collection interval and perform the following tasks:

- View SQL call text for the call
- View collection data for all plans that have executed the selected SQL call
- View tables and indexes referenced by the SQL call
- Evaluate access path information (valid CA Plan Analyzer licensing is required)
- View information in a report format

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   The Planname Summary Display panel appears.

2. Type S (SQL) in the View By field and press Enter.
   The SQL Statement Summary Display panel appears.

3. Type H (SQL History) next to an SQL call and press Enter.
   The SQL Statement History Display panel appears.

4. Specify one of the following line commands next to a time interval:

   **Q - View SQL Text**
   Shows SQL call text on the SQL Call Text Display panel for static SQL calls and on the SQL Call Dynamic SQL Summary Display panel for dynamic SQL calls.

   **P - View Plans**
   Shows collection data for all plans that have executed the SQL call on the Package/DBRM Plan Display panel.

   **T - View Tables/Indexes**
   Shows the tables and indexes referenced by the SQL call on the SQL Table Display panel. Valid CA Subsystem Analyzer licensing is required.
E - Explain

Invokes CA Plan Analyzer Quick Explain to evaluate access path information for the selected time interval. Valid CA Plan Analyzer licensing is required.

D - View Detail

Shows collection data for the SQL call on the SQL Statement History Detail panel. This lets you view the column data from the SQL Statement History Display panel with less scrolling. The fields are grouped by interval information, statement execution data, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.

The history related collection activity panel appears for the specified command.

View Dynamic SQL Activity

You can view dynamic SQL activity to identify which dynamic SQL statements within the DB2 subsystem being monitored are the most frequently used or most resource intensive.

You can view dynamic SQL call activity and resource information originating from a plan or program (package or DBRM) entry summarized by dynamic SQL text string. You can evaluate major accounting class data values from a dynamic SQL call perspective. You can identify the dynamic SQL statements and view resource consumption at the dynamic SQL statement level. You can also view the dynamic SQL statement text.

Note: You must have enabled Dynamic SQL Stats and Dynamic SQL Text Statistics at CA Detector collection start to receive dynamic SQL collection activity. These statistics do not include host variables. To include this information, enable SQL exceptions.

You can select a dynamic SQL statement and perform the following tasks:

- View SQL statement information
- Evaluate access path information (a valid CA Plan Analyzer license is required)
- View collection statistics for all packages, DBRMs, and plans that have executed the dynamic SQL statement
- View related keys
- Evaluate table activity (a valid CA Subsystem Analyzer license is required)
Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears.

2. Type Q in the View By field and press Enter.

   The Dynamic SQL Summary Display panel appears.

3. Specify one of the following line commands next to a dynamic SQL statement text:

   **D (Detail)**

   Shows dynamic SQL statement text information in a report format on the Dynamic SQL Detail Display, Plan Dynamic SQL Detail Display, or Pgm Dynamic SQL Detail Display panel. This lets you view the column data from the Dynamic SQL Summary Display, Plan Dynamic SQL Display, or Pgm Dynamic SQL Display panel with less scrolling. The fields are grouped by SQL statement information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   **E (Explain)**

   Invokes the Quick Explain component of CA Plan Analyzer to evaluate access path information. Valid CA Plan Analyzer licensing is required.

   **G (Programs)**

   Shows collection data for all the packages and DBRMs that executed the selected dynamic SQL statement on the Dynamic SQL Pgm Display panel. This panel lets you view the application activity for the packages and DBRMs that executed a specific dynamic SQL statement. From this panel, you can:

   - Type D to view the information in a report format by program on the Dynamic SQL Pgm Detail Display panel.
   - Type S to view SQL statements for the selected DBRM or package information on the Dynamic Package/DBRM SQL Display panel. Type D on this panel to view the data in a report format on the Dyn Pkge/DBRM Detail SQL Display panel. You can also invoke CA Plan Analyzer to evaluate access path information and view the tables and indexes that the dynamic SQL statement referenced. Valid CA Plan Analyzer or CA Subsystem Analyzer are required.

   **P (Plans)**

   Shows collection data for all plans that have been executed the selected dynamic SQL statement on the Dynamic SQL Plan Display panel. Type D on this panel to view the data in a report format on the Dynamic SQL Plan Detail Display panel.
Q (SQL)

Shows the SQL text for a dynamic SQL statement on the Dynamic SQL Call Text Display panel. Type E to pass the SQL call text to CA Plan Analyzer (valid licensing is assumed), or type I to pass the SQL call text to ISQL to interactively edit and execute the SQL text.

K (Keys)

Shows SQL call text related keys data. The keys are identified by type, such as DB2 connection user ID (AUTHID), correlation ID (such as batch job name or CICS transaction name), and so on.

**Note:** You must have enabled Additional View By Keys on CA Detector collection start and specified a collection profile to view keys data.

T (Tables/indexes)

Shows tables and indexes referenced by the dynamic SQL statement on the Dyn SQL Table Activity Display panel. Valid CA Subsystem Analyzer licensing is required.

Press Enter.

The dynamic SQL statement related plan activity panel appears for the specified command.

**Evaluate Access Path Information**

You can invoke CA Plan Analyzer Quick Explain to evaluate access path information. Valid CA Plan Analyzer licensing is assumed.

Quick Explain is only available for DB2 subsystems on the local z/OS or that are members of a data sharing group with an active group member on the local z/OS.

To evaluate access path information, type E next to a program or SQL call on any of the following panels:

- Plan Program Display
- Package/DBRM Summary Display
- SQL Statement Summary Display
- Dynamic SQL Summary Display
- Key Plan/Program Display
- Key Package/DBRM Display

Press Enter.
The PPA Quick Explain panel appears. From this panel, you can explain the following SQL source types:

- Plan
- Plan/DBRM combination
- Collection ID
- Collection ID/package combination

**View Collection Activity by Reporting Profile**

You can view application activity and resource use from an application perspective using a reporting profile. A reporting profile consists of a list of the plans or packages that make up the application.

A reporting profile for standard activity collection can be a plan profile or a package profile. If it is a plan profile, then all reporting profile IDs are defined to include the plan names contained in the application. If it is a package profile, then all reporting profile IDs are defined to include the packages and DBRMs contained in the application.

**Follow these steps:**

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and **1** (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears. Use the View History field to view intervals from a datastore that is currently in use by collection.

   **Note:** This panel also appears when you type **A** in the View Type field on any current display panel.

2. Type **F** in the View By field and press Enter.

   The Application Profile Display panel appears. The data is grouped according to the definitions in the current profile. The NO PROFILE entry represents all applications that are not specifically defined in the profile.

   **Note:** If a profile has not been assigned to the DB2 subsystem being viewed, the Reporting Profiles Display panel appears. You can then create or choose a reporting profile.
3. Specify one of the following line commands next to an application:

**S (View activity)**

Shows all plans or packages that have been executed by the application on the Application Planname Display or Application Package/DBRM Display panel. This lets you understand which DB2 plans or packages account for the majority of an application's workload. You can then select a plan or package and view the application activity in greater detail. For plans, you can view program, detail, and active threads. For programs, you can view plans, SQL statements, access path information, and detail.

**D (View Detail)**

Shows application activity in a report format on the Application Profile Detail Display panel. This lets you view the column data from the Application Profile Display panel with less scrolling. The fields are grouped by application information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

**H (Application History)**

Shows the application history on the Application Profile History Display panel.

Press Enter.

The application activity related panel appears for the specified command.

---

**View Application Profile History**

You can view the history of activity for an application. Data is summarized by activity and resource use for the application according to time intervals. This lets you easily identify the activity for an application in the DB2 subsystem being viewed and examine resource use by the application.

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   
   The Planname Summary Display panel appears.

2. Type F in the View By field and press Enter.
   
   The Application Profile Display panel appears.

3. Type H (Application History) next to an application and press Enter.
   
   The Application Profile History Display panel appears.
View Keys Collection Activity

You can view application activity from the perspective of optional additional view by keys as follows:

■ The DB2 connection user ID (AUTHID)
■ The correlation ID, such as batch job name or CICS transaction name
■ The DB2 connection type, such as TSO or CICS.
■ The connection name, such as the CICS region name.
■ The remote location name or IP address.
■ The end user ID that may be specified for distributed and RRSAF connections.
■ The end user transaction and workstation ID that may be specified for distributed and RRSAF connections.

Note: You must have enabled Additional View By Keys on CA Detector collection start and specified a collection profile.

You can identify DB2 subsystem activity by optional view by keys such as user or correlation ID. Additionally, you can view resource use by the optional keys.

You can select a key and perform the following tasks:

■ View all plans and programs that have been executed by the key during the collection interval
■ Access exception SQL and SQL error collection data

Follow these steps:

1. Specify the DB2 subsystem ID that you want to retrieve collection data for in the DB2 SSID field and 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.

   The Planname Summary Display panel appears.

2. Type K (Keys) in the View By field and press Enter.

   The Key Summary Display panel appears. Use the Key field to specify the standard activity data is grouped by (DB2 connection ID, correlation ID, connection type, connection name, remote location name, end user ID, end user transaction, or end user workstation ID).

3. (Optional) Change the Key value to view the standard activity data in a different order and press Enter.

   Note: Only the values you selected for collection are included in this list.
4. Specify one of the following line commands next to a time interval:

   **P (Plans)**
   Shows key activity from a plan name perspective on the Key Planname Display panel.

   **G (Programs)**
   Shows key activity from a program perspective on the Key Package/DBRM Display panel. From this panel, you can select a program and view SQL statements, plans, dynamic SQL, access path information, and detail.

   **Q (Dynamic SQL)**
   Shows the dynamic SQL activity related to a selected view by key on the Dynamic SQL Summary Display panel.

   **D (Detail)**
   Shows key activity in a report format on the Key Summary Detail Display panel. This lets you view the column data from the Key Summary Display panel with less scrolling. The fields are grouped by key information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   Press Enter.

   Additional information about the selected key appears.

---

**View Exception SQL Collection Activity**

The exception SQL collection facilities let you evaluate and control SQL from any application origin including SQL originating from client server database gateways, fourth generation tools, or typical adhoc queries.

Exception SQL collection uses collection profile information that lets you:

- Specify when a request should be considered for data capture. Exception SQL data capture provides all major accounting information for the execution of the initial request, as well as for the SQL calls executed as a result of the request.
- Define SQL call termination thresholds that, when met, cause the SQL request to be terminated.
Preempt the execution of an SQL request. For example, you can specify a resource threshold related to SQL execution cost. If the PREPARE of the SQL statement indicates the cost is higher than you specified in the collection profile, the execution of the SQL request can be preempted.

**Note:** You must define a collection profile to capture performance information that contains application definitions and resource-related information in effect for those applications.

You can view current and historical exception SQL activity and easily identify which users are executing exception SQL within the DB2 subsystem you are viewing as follows:

- By user
- By DBRM/package name
- By plan name
- By reporting profile
- By correlation ID
- By collection ID

**Note:** The information gathered is the same for current and historical activity, except for the fields defining the collection interval. For current activity, the interval collection fields are Interval Time and Interval Elapsed. For historical activity, the collection interval fields are Interval Date, Interval Time, and Elapsed Time.

**Follow these steps:**

1. Type 1 (SSID current interval data) in the Option field on the CA Detector Main Menu and press Enter.
   
   The Planname Summary Display panel appears.
   
   **Note:** This panel also appears when you type A in the View Type field on any current display panel.

2. Type X in the View Type field and press Enter.
   
   The Exception SQL User Summary Display panel appears.
   
   **Note:** Use the View By field to sort the collection activity by user, program, plan, profile, correlation ID, or collection ID. A unique user can be defined by combining the original authorization ID, plan name, and connection ID. Exception SQL activity and resource use for the interval you are viewing are grouped by original authorization ID.
3. Specify one of the following line commands:

   **S**
   
   Shows exception SQL requests by plan for the specified user during the collection interval on the Exception SQL Request Summary Display panel. This panel lets you identify which plans are executing exception SQL within the DB2 subsystem being viewed.

   **D**
   
   Shows the same data as the Exception SQL User Summary panel, but in a report format on the Exception SQL User Detail panel. The detail report lets you view all column data with less scrolling. The fields are grouped by the type of collection data.

   Press Enter.

   The related exception SQL collection activity panel appears.

**More information:**

- [Collection Profiles](#) (see page 137)
- [View Standard Collection Activity](#) (see page 64)

**View Exception SQL by User**

You can view exception SQL activity from a user perspective. Exception SQL activity and resource use are grouped by original authorization ID. This lets you easily identify which users are executing exception SQL within the DB2 subsystem being viewed. You can also identify resource use by user.

You can select a user and view all plans, DBRMs, and packages that have been executed by the plan during the collection interval, as well as SQL call statements and statistics.

You can also access standard activity and SQL error collection activity.

**Follow these steps:**

1. Type **X** in the View Type field on any panel that has this field and press Enter.
   
   The Exception SQL User Summary Display panel appears. Use the Total/Avg field to show totals or averages in the data columns.

   **Note:** This panel also appears when you type **U** in the View Type field on any current display panel.
2. Specify one of the following line commands next to a plan:

S (View Exception Reqs)
Shows the exception SQL requests during the collection interval on the Exception SQL Request Summary panel.

D (Detail)
Shows exception SQL collection activity by user in a report format on the Exception SQL User Summary Detail Display panel. This lets you view the data columns from the Exception SQL User Summary Display panel with less scrolling. The fields are grouped by OPID information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.
The user exception SQL collection activity panel appears for the specified command.

View Exception SQL Requests By Plan For A User

You can view exception SQL requests by plan for a user during the collection interval and identify which plans are executing exception SQL within the DB2 subsystem being viewed.

You can select an exception SQL request and perform the following tasks:

■ View the SQL call statement types that make up the exception SQL request
■ View the SQL call text for the selected exception SQL request
■ Evaluate access path information (valid CA Plan Analyzer licensing is required)
■ View exception SQL requests for the user

Follow these steps:

1. Type S (View Exception Req) next to an OPID on any of the following panels:
   ■ Exception SQL User Summary Display
   ■ Exception SQL Program/User Summary
   ■ Exception SQL Plan/User Summary
Press Enter.
The Exception SQL Request Summary panel appears.
2. Specify one of the following line commands next to a plan:

   **S (View SQL stats)**
   
   Shows the SQL call statement types that make up the exception SQL request during the collection interval on the Exception SQL Request Detail panel.

   **Q (View SQL text)**
   
   Shows the SQL call text for the exception SQL request on the Exception SQL Call Text panel.

   **E (Explain)**
   
   Invokes CA Plan Analyzer Quick Explain to evaluate access path information for the exception SQL request. Valid CA Plan Analyzer licensing is required.

   **D (View Detail)**
   
   Shows exception SQL collection activity by plan for a user in a report format on the Exception SQL Request Summary Detail Display panel. This lets you view the data columns from the Exception SQL Request Summary Display panel with less scrolling. The fields are grouped by SQL request information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, exception SQL collection information, program information, and end user data.

   Press Enter.

   The exception SQL collection activity panel appears for the specified command.

**View Exception SQL Call Statement Types**

You can view the SQL call statement types that make up an exception SQL request.

You can select an executed SQL call type and view a detailed description.

**Follow these steps:**

1. Type **X** in the View Type field and **U** in the View By field on any panel that has these fields and press Enter.

   The Exception SQL User Summary Display panel appears.

2. Type **S** (View Exception Req) next to an OPID and press Enter.

   The Exception SQL Request Summary panel appears.
3. Type $ next to an exception SQL request and press Enter.
   The SQL call statement types that make up the exception SQL requests appear on
   the Exception SQL Request Detail panel.

4. Type D next to an execute SQL call type and press Enter.
   The information provided on the Exception SQL Request Detail panel appears in a
   report format. This lets you view the data columns with less scrolling. The fields are
   grouped by SQL information, buffer manager activity, prepare statement cache
   activity, RID list processing, locking (IRLM) activity, data sharing lock activity,
   parallelism accounting data, wait times and counts, and additional time values.

View Exception SQL Call Text

You can view the SQL call text for exception SQL requests for the SQL call that is
currently being executed by the active thread. SQL call text is only available for threads
that are currently executing within DB2.

The actual values of the host variables is also provided with the SQL call request. The
parameter markers are listed in consecutive order as they appear in the SQL call
request. These values only appear for SQL calls that contain host variable data.

You can view the exception SQL call text and perform the following tasks:

- Pass the SQL call catalog information to Quick Explain (valid CA Plan Analyzer
  licensing is required)
- Pass the SQL call text to Quick Explain for editing (valid CA Plan Analyzer licensing is
  required)
- Pass the SQL call text to ISQL and interactively edit and execute the SQL text

Follow these steps:

1. Type X in the View Type field and U in the View By field on any panel that has these
   fields and press Enter.
   The Exception SQL User Summary Display panel appears.

2. Type S (View Exception Req) next to an OPID and press Enter.
   The Exception SQL Request Summary panel appears.
3. Type Q (View SQL text) next to an exception SQL request and press Enter.
   The SQL call text for the exception SQL call request appears on the Exception SQL Call Text panel.

4. Specify one of the following line commands next to an SQL call:

   **E -Explain SQL**
   Passes the SQL call catalog information to the Quick Explain component of CA Plan Analyzer. This is the default explain action for static SQL statements and should be used if the SQL call text contains host variable markers. Valid CA Plan Analyzer licensing is required.

   **T -Explain Text**
   Passes the SQL call text to the Quick Explain component of CA Plan Analyzer and lets you edit the call text within CA Plan Analyzer. Valid CA Plan Analyzer licensing is required.

   **I -ISQL**
   Passes the SQL call text to ISQL to interactively edit and execute the SQL text.

   **Note:** For more information about ISQL, see the *CA Database Management Solutions for DB2 for z/OS Value Pack Reference Guide*.

   Press Enter.
   The related SQL call text panel appears for the specified line command.

**View Exception SQL By Program**

You can view exception SQL requests by program for the collection interval and identify which users are executing exception SQL within the DB2 subsystem being viewed.

You can select a program and perform the following tasks:

- View program users
- View program information

**Follow these steps:**

1. Type X in the View Type field and G in the View By field on any panel that has these fields and press Enter.
   The Exception SQL Program Summary panel appears.
2. Specify one of the following line commands next to a program:

   **S (View Program Users)**
   
   Shows the users that have executed exception SQL requests during the collection interval in the selected program on the Exception SQL Program/User Summary panel.

   **D (View Detail)**
   
   Shows exception SQL collection activity by program in a report format on the Exception SQL Program Summary Detail panel. This lets you view the data columns from the Exception SQL Program Summary panel with less scrolling. The fields are grouped by program information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, additional time values, exception SQL collection information, and program information.

   Press Enter.
   
   The exception SQL collection activity panel appears for the specified command.

---

**View Exception SQL User Program Information**

You can view all the users who have executed exception SQL requests in a program during the collection interval.

You can select a user and view all plans, DBRMs, and packages that have been executed by the plan during the collection interval, as well as SQL call statements and statistics.

You can also access standard activity and SQL error collection activity.

**Follow these steps:**

1. Type **X** in the View Type field and **G** in the View By field on any panel that has these fields and press Enter.
   
   The Exception SQL Program Summary panel appears.

2. Type **S** next to a program and press Enter.
   
   The Exception SQL Program/User Summary panel appears.
3. Specify one of the following line commands next to a user:

**S (View Exception Reqs)**

Shows the exception SQL requests during the collection interval on the Exception SQL Request Summary panel.

**D (Detail)**

Shows exception SQL collection activity by user in a report format on the Exception SQL User Summary Detail Display panel. This lets you view the data columns from the Exception SQL User Summary Display panel with less scrolling. The fields are grouped by OPID information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.

The exception SQL collection activity panel appears for the specified command.

**View Exception SQL by Plan**

You can view all plan names that have executed exception SQL requests during the collection interval.

You can select a plan and perform the following tasks:

- View plan users
- View plan information

**Follow these steps:**

1. Type `X` in the View Type field and `P` in the View By field on any panel that has these fields and press Enter.

   The Exception SQL Planname Summary panel appears.
2. Specify one of the following line commands next to a plan:

S (View Plan Users)
Shows the users that have executed exception SQL requests during the collection interval in the selected plan on the Exception SQL Plan/User Summary panel.

D (View Detail)
Shows exception SQL collection activity by user in a report format on the Exception SQL Plan/User Summary panel. This lets you view the data columns from the Exception SQL Planname Summary panel with less scrolling. The fields are grouped by OPID, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.
The exception SQL collection activity panel appears for the specified command.

View Exception SQL User Plan Information

You can view all the users who have executed exception SQL requests in a plan during the collection interval.

You can select a user and view all plans, DBRMs, and packages that have been executed by the plan during the collection interval, as well as SQL call statements and statistics.

You can also access standard activity and SQL error collection activity.

Follow these steps:

1. Type X in the View Type field and P in the View By field on any panel that has these fields and press Enter.
The Exception SQL Planname Summary panel appears.

2. Type S next to a plan and press Enter.
The Exception SQL Plan/User Summary panel appears.
3. Specify one of the following line commands next to a user:

**S (View Exception Reqs)**

Shows the exception SQL requests during the collection interval on the Exception SQL Request Summary panel.

**D (Detail)**

Shows exception SQL collection activity by user in a report format on the Exception SQL Planname Summary panel. This lets you view the data columns from the Exception SQL Plan/User Summary panel with less scrolling. The fields are grouped by plan information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

Press Enter.

The exception SQL collection activity panel appears for the specified command.

---

**View Exception SQL by Reporting Profile**

You can view exception SQL request activity and resource use from an application perspective using a reporting profile. The reporting profile consists of a list of the plans, packages, or users that make up the application.

An exception SQL reporting profile can be a plan profile, a package profile, a user profile, a correlation ID profile, or a collection ID profile:

- If it is a plan profile, then the profile ID is defined to include the plan names contained in the application.
- If it is a package profile, then the ID is defined to include the packages and DBRMs contained in the application.
- If it is a user profile, then the profile ID is defined to include the user IDs contained in the application.
- If it is a correlation ID profile, then the profile ID is defined to include the correlation IDs in the application.
- If it is a collection ID profile, then the profile ID is defined to include the collection IDs in the application.

Users can share reporting profiles.

CA Detector aggregates (totals) display output by profile ID. This lets you analyze DB2 subsystem use by application. You can easily identify which applications account for the majority of DB2 resource use.

You can view application activity, detail, and history.
Follow these steps:

1. Type X in the View Type field and F in the View By field on any panel that has these fields and press Enter.

   The Exception SQL Profile Display panel appears. The data is grouped according to the definitions in the current profile. The NO PROFILE entry represents all applications that are not specifically defined in the profile.

   **Note:** If a profile has not been assigned to the DB2 subsystem being viewed, the Reporting Profiles Display panel appears. You can then create or choose a reporting profile.

2. Specify one of the following line commands next to an application:

   **S (View Exception Activity)**

   Shows all plans or packages that have been executed by the application on the Application Plannname Display or Application Package/DBRM Display panel. This lets you understand which DB2 plans or packages account for the majority of an application’s workload. You can then select a plan or package and view the application activity in greater detail. For plans, you can view program, detail, and active threads. For programs, you can view plans, SQL statements, access path information, and detail.

   **D (View Detail)**

   Shows application activity in a report format on the Exception SQL Profile Detail panel. This lets you view the column data from the Exception SQL Profile Display panel with less scrolling. The fields are grouped by application information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   Press Enter.

   The application activity related panel appears for the specified command.

---

**View Exception SQL By Correlation ID**

You can view exception SQL requests by correlation ID for a collection interval.

You can select a correlation ID and perform the following tasks:

- View correlation ID information
- View users that have executed exception SQL requests by correlation ID
- View plans that have executed exception SQL requests by correlation ID
- View programs that have executed exception SQL requests by correlation ID
Follow these steps:

1. Type X in the View Type field and C in the View By field on any panel that has these fields and press Enter.
   The Exception SQL Corrid Summary panel appears.

2. Specify one of the following line commands next to a correlation ID:

   **D (View Detail)**
   Shows exception SQL collection activity by correlation ID in a report format on the Exception SQL Corrid Summary panel. This lets you view the data columns from the Exception SQL Corrid Summary panel with less scrolling. The fields are grouped by correlation ID information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   **S (View Users)**
   Shows all users that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Corrid/User Summary Display panel.

   **P (View Plans)**
   Shows the plans that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Corrid/Plan Summary panel.

   **G (View Programs)**
   Shows the programs that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Corrid/Pgm Summary panel.

Press Enter.
The exception SQL collection activity panel appears for the specified command.

**View Exception SQL By Collection ID**

You can view exception SQL requests by collection ID for a collection interval.

You can select a collection ID and perform the following tasks:

- View collection ID information
- View users that have executed exception SQL requests by collection ID
- View plans that have executed exception SQL requests by collection ID
- View programs that have executed exception SQL requests by collection ID
Follow these steps:

1. Type **X** in the View Type field and **C** in the View By field on any panel that has these fields and press Enter.

   The Exception SQL Collid Summary panel appears.

2. Specify one of the following line commands next to a collection ID:
   
   **D (View Detail)**
   
   Shows exception SQL collection activity by collection ID in a report format on the Exception SQL Collid Summary panel. This lets you view the data columns from the Exception SQL Collid Summary panel with less scrolling. The fields are grouped by collection ID information, buffer manager activity, prepare statement cache activity, RID list processing, locking (IRLM) activity, data sharing lock activity, parallelism accounting data, wait times and counts, and additional time values.

   **S (View Users)**
   
   Shows all users that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Collid/User Summary panel.

   **P (View Plans)**
   
   Shows the plans that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Collid/Plan Summary panel.

   **G (View Programs)**
   
   Shows the programs that have executed exception SQL requests for the selected correlation ID during the collection interval on the Exception SQL Collid/Pgm Summary panel.

   Press Enter.

   The exception SQL collection activity panel appears for the specified command.

---

**View SQL Errors Collection Data**

CA Detector contains collection and reporting facilities to assist in detecting and understanding application SQL error activity.

CA Detector’s SQL error collection lets you view data on application errors incurred as a result of abnormal SQL call return codes.

**Note:** No performance trace activity is required for SQL error collection.

You can create collection profiles to tailor SQL error collection by excluding SQL error conditions which are of no interest.
Additional thresholds are also provided to limit the amount of SQL error information to be retained.

You can view SQL error data that has been captured for the current interval or for historical data saved in a datastore for an individual subsystem or data sharing group.

This lets you quickly determine which SQL errors are occurring most frequently on a DB2 subsystem by setting the SQL error collection option when you initialize collection (set the SQL Error Text and Collection Profile options when you start collection).

Use collection and reporting profiles to detect and understand application SQL error activity. Collection profiles are used to tailor SQL error collection by excluding SQL error conditions that are of no interest. Additional thresholds area also provided to limit the amount of SQL error information to be retained.

**Note:** SQL error collection does not capture SQL errors incurred during DB2 CONNECT, CREATE THREAD, SIGNON, or other DB2 activity not related to SQL statement execution, including deadlocks or timeouts that occur during the execution of these statements.

To view SQL error collection data, type E in the View Type field of any CA Detector panel with that field and press Enter.

The DETECTOR SQL ERROR Summary Display or DETECTOR DS Group Error Interval Display panel appears. You can use these panels to view data about application errors resulting in abnormal SQL return codes.

From the SQL error panels, you can use the View By field to view the following types of activity from the collected data:

**C (SQL code)**

Shows data by SQL error code.

**G (program)**

Shows collection data by DBRM/package name.

Use line commands to view SQL statements, collection data for all executed plans, additional program detail, dynamic SQL, and program history. If licensed for CA Plan Analyzer you can invoke CA Plan Analyzer to evaluate access path information. For data sharing groups, you can view SQL statements and additional program history, and invoke CA Plan Analyzer to evaluate access path information. You can also view interval data for data sharing group members.

**U (user)**

Shows collection data by user. A unique user can be defined by combining the original authorization ID, plan name, and connection ID.
**L (List)**  
Shows a list of errors for the interval.

**A (All)**  
Shows all the SQL statements that generated an SQL error code for all intervals in a datastore.

You can also use the following line commands to view additional information:

**U (View Users)**  
Shows SQL error data by user for the SQL error code.

**G (View Programs)**  
Shows all programs (DBRMS or packages) that generated a specified SQL code.

**L (View List)**  
Shows all SQL statements that generated an SQL error code during the interval.

**M (Member Activity)**  
Lists interval data for the members that were used to formulate the data on the selected line.

**More information:**

[Collection Profiles](see page 137)
View SQL Error Codes

You can view all SQL errors captured for the current interval. You can quickly determine which SQL errors are occurring most frequently. You can select an SQL code and view all users or programs that encountered the error.

Follow these steps:

1. Type **E** in the View Type field of any CA Detector panel with that field and press Enter.
   The SQL Error Summary Display panel appears. The View By field defaults to SQLcode.

2. Specify one of the following line commands next to an SQL error code:
   **U (View Users)**
   Formats the SQL error data on the SQL ERROR User/SQLCODE Summary panel. Output is summarized by user and listed chronologically with the most recent occurrence first. After you select a user, the SQL Error Detail Display panel appears. All statements that generated a particular SQL code for the user appear. The SQL statements are listed chronologically with the most recent occurrence first.

   **G (View Programs)**
   Shows all the programs (DBRMs or packages) that generated a specific SQL code on the SQL Error Program/SQLCODE Summary panel. The detail lines are listed chronologically with the most recent occurrence first. After you select a program, the SQL Error Detail Display panel appears. All statements that generated a particular SQL code for the program appear. The SQL statements are listed chronologically with the most recent occurrence first.

   **L (View List)**
   Shows the SQL statements that generated a particular SQL code during the interval. From 1 to 999 SQL statement detail lines can appear. The statements are listed chronologically, with the most recent occurrence first.

Press Enter.

The SQL error code collection activity panel appears for the specified command.
3. Specify one of the following line commands:

**S (SQLCA Message Display)**

Shows the SQLCA area for the SQL call that issued the return code on the SQLCA Data Area Display panel. You can toggle between showing the SQLCA area message or showing the SQLCA area in hexadecimal.

**Q (SQL Call Text)**

Shows the SQL call text on the SQL Call Text Display panel for static SQL calls and on the SQL Call Dynamic SQL Summary Display panel for dynamic SQL calls.

**V (View Detail)**

Shows all SQL statements that generated a particular SQL code for the user being viewed on the SQL Error Detail Display panel. This lets you view the data columns with less scrolling. The fields are grouped by SQL error environment information, SQL error data from the SQLCA, and warning information from the SQLCA.

Press Enter.

The SQL error code panel appears.

---

**View SQL Errors by Program**

You can view all the programs (DBRMs or packages) that generated a specific SQL code. You can determine what applications account for the SQL error activity for that code. You can then select a program and view a list of the SQL codes generated by the program.

**Follow these steps:**

1. Type **E** in the View Type field and **G** in the View By field of any CA Detector panel with that field and press Enter.

   The SQL Error Program Display panel appears. From this panel, you can determine which SQL errors are occurring in the program. You can select an SQL code to view the SQL calls that generated errors.

2. Select a program and press Enter.

   The SQL statements that generated a particular SQL code for the program being view appear on the SQL Error Detail Display panel. The SQL statements are listed chronologically, with the most recent occurrence first.
3. Specify one of the following line commands:

**S (SQLCA Message Display)**

Shows the SQLCA area for the SQL call that issued the return code on the SQLCA Data Area Display panel. You can toggle between showing the SQLCA area message or showing the SQLCA area in hexadecimal.

**Q (SQL Call Text)**

Shows the static and dynamic SQL call text.

Press Enter.

The SQL error code collection activity panel appears for the specified command.

---

**View SQL Errors by User**

You can view all the users that generated an SQL error. This helps determine which users and plans account for the SQL error activity. You can then select a user to view the SQL codes generated by that user.

You can access the SQL Error User Summary Display by entering U in the View By field on the SQL Error Summary Display or on any SQL Error display that contains the View By field.

**Follow these steps:**

1. Type **E** in the View Type field and **U** in the View By field of any CA Detector panel with that field and press Enter.

   The SQL Error User Summary Display panel appears.

2. Select a user and press Enter.

   The SQL Error User Display panel appears. This panel lists all the SQL errors generated by a user for the current interval.

3. Select an error and press Enter.

   The SQL Error Detail Display panel appears. This panel shows all the SQL statements that generated a particular SQL code for the user being viewed. The SQL statements are listed chronologically, with the most recent occurrence first.

4. Specify one of the following line commands:

   **S (SQLCA Message Display)**

   Shows the SQLCA area for the SQL call that issued the return code on the SQLCA Data Area Display panel. You can toggle between showing the SQLCA area message or showing the SQLCA area in hexadecimal.

   **Q (SQL Call Text)**

   Shows the static and dynamic SQL call text.
V (View Detail)

Shows all SQL statements that generated a particular SQL code for the user being viewed on the SQL Error Detail Display panel. This lets you view the data columns with less scrolling. The fields are grouped by SQL error environment information, SQL error data from the SQLCA, and warning information from the SQLCA.

Press Enter.

The SQL error panel appears.

View a List of SQL Errors

You can view a list of all SQL statements that generated an SQL error code during the collection interval. The SQL statements are listed chronologically, with the most recent occurrence first. From 1 to 999 SQL statement detail lines can be viewed at one time.

Follow these steps:

1. Type E in the View Type field and L in the View By field of any CA Detector panel with that field and press Enter.

   The SQL Error List Display panel appears. From this panel, you can view the SQLCA Message Display or the SQL call text for the statement.

2. Specify one of the following line commands:

   S (SQLCA Message Display)

   Shows the SQLCA area for the SQL call that issued the return code on the SQLCA Data Area Display panel. You can toggle between showing the SQLCA area message or showing the SQLCA area in hexadecimal.

   Q (SQL Call Text)

   Shows the static and dynamic SQL call text.

   V (View Detail)

   Shows all SQL statements that generated a particular SQL code for the user being viewed on the SQL Error Detail Display panel. This lets you view the data columns with less scrolling. The fields are grouped by SQL error environment information, SQL error data from the SQLCA, and warning information from the SQLCA.

   Press Enter.

   The SQL error panel appears.
List All SQL Error Codes

You can list all the SQL statements that generated an SQL error code for all intervals in a datastore. From 1 to 999 SQL statement detail lines can be viewed at one time.

Note: You can only access this information when viewing historical collection data.

Follow these steps:

1. Type E in the View Type field and A in the View By field of any CA Detector panel with that field and press Enter.

   The SQL Error List-All Display panel appears. The SQL statements are listed chronologically with the most recent occurrence first. The display can show from 1 to 999 SQL statement detail lines. From this display, you can view the SQLCA Message Display or SQL call text for a statement.

2. Specify one of the following line commands:

   S (SQLCA Message Display)
   Shows the SQLCA area for the SQL call that issued the return code on the SQLCA Data Area Display panel. You can toggle between showing the SQLCA area message or showing the SQLCA area in hexadecimal.

   Q (SQL Call Text)
   Shows the static and dynamic SQL call text.

   V (View Detail)
   Shows all SQL statements that generated a particular SQL code for the user being viewed on the SQL Error Detail Display panel. This lets you view the data columns with less scrolling. The fields are grouped by SQL error environment information, SQL error data from the SQLCA, and warning information from the SQLCA.

   Press Enter.

   The SQL error panel appears.
View SQLCA Data Area

The SQLCA Data Area Display shows the SQLCA area for the SQL call that issued the return code. You can toggle between displaying the SQLCA area messages and displaying the SQLCA area in hexadecimal.

You can access the SQLCA Data Area Display by selecting an SQL statement from one of the SQL Error List displays or SQL Error Detail displays.

This area shows the contents of the SQLCA in message format or hexadecimal.

**Note:** For information about the data contained in this area, see the current IBM DB2 SQL Reference Manual.

The following shows the SQLCA in message format:

```
DSNT408I SQLCODE = -805, ERROR: DBRM OR PACKAGE NAME D41ADB2..PTADRVAC.0F0F-1F0F04040404 NOT FOUND IN PLAN RDTP025S. REASON 01
DSNT418I SQLSTATE   = 51002 SQLSTATE RETURN CODE
DSNT415I SQLERRP    = DSNXEPM SQL PROCEDURE DETECTING ERROR
DSNT416I SQLERRD    = -250 0 0 -1 0 0 SQL DIAGNOSTIC INFORMATION
DSNT416I SQLERRD    = X'FFFFFF06' X'00000000' X'00000000' X'FFFFFFFF'
                        X'00000000' X'00000000' SQL DIAGNOSTIC INFORMATION
*****************************************************************************
```

The following shows the SQLCA in hexadecimal format:

```
HEX OFFSET  HEX DATA                             CHARACTER DATA
---   -------------------------                    -------------------------
+ 0000  E2D8D3C3 C1A40404 09000000 FFFFFCDB  *SQLCA   ........*
+ 0010  002EC4F4 F1C1C4C2 F24B4BD7 E3C1C4D9  *D41ADB2..PTADR*
+ 0020  E5C1C34B F0C6F0C6 F1C6F0C6 F0F0F4F0  *VAC.0F0F0F0F0*
+ 0030  F4F0F4F0 FFD9C4E3 D7F0F2F5 E2FFFFF1  *E4040.RDTP025S.01*
+ 0040  40404040 40404040 40404040 40404040  *DSNXEPM*
+ 0050  40404040 40404040 40404040 40404040  *DSNXEPM*
+ 0060  FFFFFFFF 00000000 00000000 FFFFFFFF  *.............*
+ 0070  00000000 00000000 40404040 40404040  *51002      *
+ 0080  404040F5 F1F0F0F2  *51002      *
```
View Dynamic SQL Errors

You can view dynamic SQL errors.

**Note:** If you have not enabled SQL error text collection when the collection was started, you can only view the SQL call text for dynamic SQL statements if the SQL error was also captured as an SQL exception; otherwise, the request fails with a DT160I error message.

To view dynamic SQL errors, type Q in the S column for a row on the SQL Error Detail Display panel.

The SQL Error Call Text Display appears.

View SQL Call Text for Static SQL Statements

You can view SQL call text for dynamic SQL statements, exceptions, and errors.

Static SQL statements can be viewed from the Standard Activity, Exception SQL, and SQL error panels.

**Note:** CA Detector might be unable to display static SQL text. Some SQL statements, such as CONNECT and SET, do not allow CA Detector to collect enough information to correctly classify the issuing program type as a package or DBRM. Typically, CA Detector is able to resolve this ambiguity when the program executes additional SQL statements that CA Detector can classify. However, if a program executes only a single unclassifiable statement, the program type remains unknown, and an attempt to view SQL call text fails with a DT165E message. In other cases, the attempt to view static SQL text might fail with a DT161E message, due to a bind or authorization error that prevents the Xmanager address space from reading the DB2 catalog. If you receive the DT161E message, examine the related Xmanager job log for PDT0220 messages that describe the failure.

To view static SQL call text for static SQL statements, type Q in the S column next to the static SQL call from any standard activity SQL statement display, the Exception SQL Request Summary Display, or the SQL Error Detail Display and press Enter.
The SQL Call Text Display panel appears. On this panel, you can view the entire SQL text and select from the following text processing options:

- **Type E** and press Enter to explain static SQL statements with CA Plan Analyzer Quick Explain.
  
  The related SQL statement and section number is passed to CA Plan Analyzer.
  
  **Note:** Choose this option for static SQL statement text containing host variables. A license for CA Plan Analyzer is required to use this feature.

- **Type T** to pass the SQL call text directly to CA Plan Analyzer Quick Explain.
  
  The SQL call text is available for edit from within CA Plan Analyzer.
  
  **Note:** The explain might fail with a -204 SQL error for static SQL statements containing host variables. If this problem occurs, select E to explain the SQL call statement. A license for CA Plan Analyzer is required to use this feature.

- **Type I** to pass the SQL call text directly to the Value Pack ISQL application. Within this application, you can interactively edit, save, and execute the SQL call text.

---

**View DB2 Data Sharing Group Activity**

With no additional overhead for data collections, you can view aggregated data sharing statistics for a data sharing group.

For the data to be aggregated, a collection should be started for each included member. The collection must use the sysplex interval and for historical data, must use the same datastore name. The data provides integrated and comparative views of your DB2 applications across the members of a data sharing group.

The DS Group input field on the CA Detector Main Menu lets you specify which group you want to work with. This field is independent of the DB2 SSID input field.

**Note:** The data sharing group options and panels are very similar to the panels for an individual DB2 subsystem. Aggregated data is available for the program activity and SQL error views.

**Follow these steps:**

1. **Type 9** in the Option field on the CA Detector Main Menu and press Enter.

   The DETECTOR DS Group Planname Summary Display panel appears. From this panel, you can view application activity from a plan name perspective for the data sharing group.

   You can use the View Type field to view standard (A) or error (E) activity and the View By field to view information by plan name (P), DBRM/package name (G), keys groups (K), and all SQL calls issued during the collection interval (S).
2. Specify one of the following line commands to view additional information:

   **S**
   
   Shows the next level of granularity for the selected line.

   **M (Member Activity)**
   
   Lists interval data for data sharing group members that were used to formulate the data on the selected line.

   **D (Detail)**
   
   Shows a detail report of the column data for the selected line. The detail report shows the same column data but in a report format, which lets you view all column data with less scrolling.

   Press Enter.

   The data sharing group panel for the specified command appears.

---

**Set Default Data Sharing Group**

Use the following procedure to set the default data sharing group.

To set a default data sharing group, type **11** (View DS Group Collection Status) on the CA Detector Main Menu and press Enter.

The Data Sharing Groups Display panel appears.

---

**View Status of Data Sharing Group Members**

Use the following procedure to view the DB2 status and CA Detector collection status of members of a data sharing group.

To view status of data sharing group members, enter **M** for a display line on the Data Sharing Groups Display and press Enter.

The Data Sharing Group Member Status Display appears for the selected data sharing group. You can also use this display to determine which group members will be included in the integrated statistics and evaluate conditions that may be excluding specific group members from statistics integration.
**View Data Sharing Group Activity By Plan**

You can use the DS Group Plansname Summary Display to view data sharing group application activity from a plan name perspective. This display summarizes the application activity and resource use by plan name.

You can use this display to identify the most-frequently-used plans in a data sharing group and to examine resource use by plan name. You can also easily choose to view the activity for a selected plan across all the group members that have executed the plan during the collection interval.

**Follow these steps:**

1. Type 9 (View DS Group Current Interval Data) on the CA Detector Main Menu and press Enter.
   The DS Group Plansname Summary Display panel appears.
2. Type D next to a plan and press Enter.
   The DS Group Plansname Detail Display panel appears. This display shows the same data as the DS Group Plansname Summary Display shows but in a report format for the selected display line. The fields are grouped by the type of collection data. Use the F7 (Up) and F8 (Down) keys to scroll through the report. Press F3 (End) to return to the DS Group Plansname Summary Display.

**View Plan Activity for Data Sharing Group Members**

The DS Group Plansname Member Summary Display summarizes application activity and resource use across the data sharing group members. You can use this display to identify how the activity for a plan is distributed across the member subsystems.

**Follow these steps:**

1. Type 9 (View DS Group Current Interval Data) on the CA Detector Main Menu and press Enter.
   The DS Group Plansname Summary Display panel appears.
2. Type D next to a plan and press Enter.
   The DS Group Plansname Detail Display panel appears.
3. Type M next to a plan and press Enter.
   The DS Group Plansname Member Summary panel appears.
4. Specify one of the following line commands:
   - S (Programs)
   - D (Detail)
   - Q (Dynamic SQL)
■ H (Plan History)
■ T (Active Threads)

Press Enter.

The data sharing group panel appears for the specified command.

**Note:** The line commands available on this display and other DS group member displays are the same as the line commands on the corresponding DB2 subsystem standard activity displays.

---

**Evaluate Program Activity for Data Sharing Group Members**

You can use the DS Group Plan/Program Display to view program (DBRM or package) activity originating from a selected plan. You can evaluate the plan activity and resource use from a DBRM or package point of view. You can view the program activity for the entire data sharing group or comparatively across the group members.

This display summarizes the plan resource use and SQL activity by program name. The display information makes it easy to identify the programs in the plan that account for most of the plan activity and resource consumption. You can view all major accounting class data values at the program level.

**Note:** If you have a license for CA Plan Analyzer, you can invoke it from this display to perform access path analysis for a selected program.

**Follow these steps:**

1. Select a plan from the DS Group Planname Summary Display.
   
   The DS Group Plan/Program Display appears.

2. Enter the D line command to select a display line.
   
   The DS Group Plan/Program Detail Display appears.

---

**View Plan Program Activity Across Data Sharing Group Members**

The DS Group Plan Program Member Summary Display summarizes plan program activity and resource use across the data sharing group members. Using this display, you can identify how the activity for a program is distributed across the member subsystems.

To view program activity across data sharing group members, enter the M line command to select a display line.

The DS Group Plan/Program Member Summary Display appears.
Review Data Sharing Group Plan SQL Call Types and Resource Consumption

The DS Group Plan SQL Display formats the SQL call activity and resource information originating from a selected plan/program entry. You can use the display to view the SQL call types and statement numbers for the SQL activity that have been executed from the program being viewed.

Use this display to evaluate major accounting class data values from an SQL call perspective. You can identify the SQL call types and view resource consumption at the SQL call level. You can also view the SQL text for static SQL calls.

Follow these steps:
1. Select a program from the DS Group Plan Program Display.
   The DS Group Plan SQL Display appears.
2. Enter the D line command to select a display line.
   The DS Group Plan SQL Detail Display appears.

View SQL Activity for Plans Across Data Sharing Group Members

The DS Group Plan SQL Member Display formats the SQL call activity and resource use across the data sharing group members. Using this display, you can identify how the SQL activity is distributed across the member subsystems.

To view SQL activity for plans across data sharing group members, enter the M line command to select a display line.

The DS Group Plan SQL Member Summary Display appears.

Evaluate Application Performance by Program for Data Sharing Groups

You can use the DS Group Package/DBRM Summary Display to evaluate application performance from an application program point of view. It summarizes display output by application DBRM/package name.

You can view all major accounting data values for the programs appearing in the display.

Follow these steps:
1. Select the View By option G from the DS Group Planname Summary Display.
   The DS Group Package/DBRM Summary Display appears.
2. Enter the D line command to select a display line.
   The DS Group Package/DBRM Detail Display appears.
View Program Activity Across Data Sharing Group Members

The DS Group Package/DBRM Member Summary Display formats application program activity and resource use across the data sharing group members. Using this display, you can identify how the program activity is distributed across the member subsystems.

To view program activity across data sharing group members, enter the M line command to select a display line.

The DS Group Package/DBRM Member Summary Display appears. Specify the S (SQL), P (plans), E (explain), D (detail), Q (dynamic SQL), and H (program history) commands as needed.

Review Data Sharing Group Program SQL Call Use and Resource Consumption

The DS Group Package/DBRM SQL Display formats SQL information for a program being viewed. Output is summarized by SQL statement.

All major accounting class data values are available for viewing from a SQL call perspective. From this display, you can determine program SQL call use and resource consumption. Information provided is for all plan users of the program. You can also view the text of static SQL calls.

Follow these steps:
1. Specify View By option G, and then use the S line command to select a program from the DS Group Package/DBRM Summary Display.
   The DS Group Package/DBRM SQL Display appears.
2. Enter the D line command to select a display line.
   The DS Group Package/DBRM SQL Detail display appears.

View Program SQL Activity Across Data Sharing Group Members

The DS Group Package/DBRM SQL Members Display formats application program SQL activity and resource use across the data sharing group members. Using this display, you can identify how the program SQL activity is distributed across the member subsystems.

To view program SQL activity across group members, enter the M line command to select a display line.

The DS Group Package/DBRM SQL Member Summary Display appears.
View Data Sharing Group SQL Call Summary Information

You can use the DS Group SQL Statement Summary Display to view all the SQL calls issued during the collection interval. You can then select a call and view SQL call text, view plan names associated with the call, invoke Quick Explain, invoke the SQL editor, or view a detail display for the SQL call.

Follow these steps:
1. Specify S in the View By field on an integrated data sharing collection display.
   The DS Group SQL Statement Summary Display appears.
2. Enter the D line command to select a display line.
   The DS Group SQL Statement Detail Display appears.

View SQL Activity Across Data Sharing Group Members

The DS Group SQL Statement Member Display formats the SQL call activity and resource use across the data sharing group members. Using this display, you can identify how the SQL activity is distributed across the member subsystems.

To view SQL activity distribution across data sharing members, enter the M line command to select a display line.

The DS Group SQL Statement Member Summary Display appears.

CA Subsystem Analyzer Panel Integration

With both CA Detector and CA Subsystem Analyzer in use, we provide panel integration to maximize product functionality and usefulness when evaluating your DB2 subsystem performance and subsystem application workload activity.

Selected panels from within CA Subsystem Analyzer let you invoke panels in CA Detector. For example, if you are using CA Subsystem Analyzer to view table activity for a selected database, you can invoke a related CA Detector panel to view all SQL activity that has referenced the table. Additionally, you can view DB2 tables and indexes referenced by an SQL statement whenever you view SQL statement activity within CA Detector.

Datastore Considerations and Naming Conventions

If you want to externalize data to be viewed on the integrated panels, you must create a datastore in both products with identical names; that is, they must have the same high-level qualifier and datastore name.
More information:

How To Synchronize Collection (see page 59)

Panels Accessible from CA Subsystem Analyzer

There are two panels in CA Subsystem Analyzer from which you can navigate into CA Detector. The CA Detector panels then access numerous subordinate panels to provide additional information.

The CA Subsystem Analyzer panels that let you invoke CA Detector are the Table Activity Display and Index Activity Display. To access CA Detector information from CA Subsystem Analyzer, specify one of the following on the SS Analyzer Table Activity Display panel:

- Type A (View SQL Activity) next to the table you want to view and press Enter.
  The DETECTOR Table SQL Activity Display panel appears. This panel lets you view all application SQL activity that has referenced the table. From this panel, other CA Detector panels are available for additional analysis of the SQL activity that has referenced the table.

- Type S (View Index Activity) next to the table you want to view index activity for and press Enter.
  The SS Analyzer Indexspace Activity Display panel appears.
  Type A (View SQL Activity) next to an indexspace name and press Enter.
  The Detector Index SQL Activity Display panel appears. This panel lets you view all SQL that has referenced the index during the collection interval. From this panel, other CA Detector panels are available for additional analysis of the SQL activity that has referenced the index.

Table and Index Getpage Activity

When both products are in use, you can view table and index getpage activity at the SQL statement level. This lets you invoke the SQL Table Activity Display panel from all points within CA Detector where SQL activity displays. For example, if you select a plan, then select a program, and then view its SQL statements, you can select the SQL statement and view all the tables and indexes that the SQL statement has referenced.

This provides an understanding of getpage activity incurred by the application. You can determine the tables and indexes referenced by the SQL statement. You can also observe what type of access path was used in deriving the result set requested by the SQL statement.
Displays Accessible From CA Subsystem Analyzer

CA Detector and CA Subsystem Analyzer output has been integrated on selected displays to maximize product functionality and usefulness when evaluating your DB2 subsystem performance and subsystem application workload activity. This chapter discusses the integrated displays and details the information they provide.

Most integrated displays have a line command area to the left of the display lines. The following describes possible line commands for the integrated displays. Not all line commands are valid on all displays; valid line commands are listed on each display.

**A (View All Tables)**
Enter A to view all tables and indexes referenced by the SQL call.

**E (Explain)**
Enter E to pass the SQL call text to CA Plan Analyzer Quick Explain. You must have a license for CA Plan Analyzer to use this feature. See the CA Plan Analyzer for DB2 for z/OS User Guide for information about Quick Explain.

**P (View Plans)**
Enter P to view the plans that have executed the SQL call referencing the table.

**Q (View SQL Text)**
Enter Q to view the SQL call text.

**S (View Indexes)**
Enter S to view the indexes on the table that referenced the SQL call.

**S (View Tables/Indexes)**
Enter S to view all tables and indexes that referenced the SQL call.

**Table SQL Activity Display**

The Table SQL Activity Display lets you view all the SQL that has referenced a table and its related indexes during the collection interval. The output displays SQL getpage activity referencing the table as well as getpage activity referencing indexes that exist on the table.

The output lets you determine which application SQL most frequently references the tables and its indexes. You can also determine whether the SQL activity displayed is referencing the table using sequential access or index access by evaluating other output columns.
To access the CA Detector Table SQL Activity Display from CA Subsystem Analyzer, enter option A next to the tablename on the Analyzer Table Activity Display and press Enter. The Table SQL Activity Display appears.

The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

- `COLLID`
- `CONTOKEN`
- `IS_GETPAGE`
- `ISCT`
- `PCTTGP`
- `PLCNT`
- `PROGRAM`
- `SECT#

See the “Display Columns” chapter for descriptions of the data columns.

**Table SQL/Multiple Tables Accessed Display**

The Table SQL/Multiple Tables Accessed Display shows all the SQL that has referenced the table selected, as well as any additional tables, during the collection interval. For example, SQL that has referenced the viewed table during join processing would be displayed because the SQL referenced more than a single table when it executed. Other activity, such as embedded dynamic SQL that references multiple tables in different executions, may also appear in the displayed output. Dynamic SQL PREPARE statements reference many DB2 catalog tables during statement execution, so they also may appear in the output.
To access the Table SQL/Multiple Tables Accessed display, enter option B in the View Type field and press Enter.

```
> ---- DETECTOR Table SQL/Multiple Tables Accessed ----
COMMAND ===> Scroll ==> PAGE
LINE 1 OF 2
DB2 SSID ===> ssid  Database ===> DSNDB07  Tablespace ===> DSN4K02
Table ===> TEMP_TABLE  Tablecreator ===> SYSIBM
View Type ===> B A -Table SQL, * -Table SQL/Multi-Table, C -Table SQL/Workfile
Total/Avg ===> T  Exclude Prepare ===> N Y/N
Interval Time ===> 02:00  Interval Elapsed ===> 01:58:57

SQLCALL  PROGRAM  STMT# TBCT ISCT SQL       TOT_TBGETP TOT_ISGETP TB_GETPAGE
--------  --------  ------- ------- ------- ------- ------- ------- -------
  OPEN     POA#IAS  02305  4  4  14  191464  149836  58982
  OPEN     RUA$RI1  00168  3  1  1  834  8  42
******************************************************************************
S -View Tables/Indexes, P -View Plans, Q -View SQL Text, E -Explain
```

The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

- COLLID
- CONTOKEN
- IS_GETPAGE
- ISCT
- PROGRAM
- SECT# SQL
- SQLCALL
- STMT# TB_GETPAGE TBCT
- TOT_ISGETP TYPE VERSION

See the “Display Columns” chapter for descriptions of the data columns.
Table SQL/Workfile Accessed Display

The Table SQL/Workfile Accessed display lets you view all the SQL that has referenced your selected table and the workfile during the collection interval. Using this display, you can identify excessive workfile use by SQL referencing the table.

To access the Table SQL/Workfile Accessed display, enter option C in the View Type field and press Enter.

```
> ------ DETECTOR Table SQL/Workfile Accessed ------
COMMAND ===> Scroll ==> PAGE
DB2 SSID ===> ssid        Database ==> DSND0606        Tablespace ==> SYSDBASE
Table      ===> SYSFOREIGNKEYS Tablecreator ==> SYSIBM
View Type ==> C  A -Table SQL, B -Table SQL/Multi-Table, * -Table SQL/Workfile
Total/Avg ==> T
Interval Time ==> 02:00                           Interval Elapsed ==> 04:41.81
-------------------------------------------------------------------------------
S -View Tables/Indexes, P -View Plans, Q -View SQL Text, E -Explain

<table>
<thead>
<tr>
<th>SQLCALL</th>
<th>PROGRAM</th>
<th>STMT#</th>
<th>TBCT</th>
<th>ISCT</th>
<th>SQL</th>
<th>SQLCALL</th>
<th>SECT#</th>
<th>TB_GETPAGE</th>
<th>TBCT</th>
<th>TOT_TBGETP</th>
<th>TOT_ISGETP</th>
<th>WF_GETPAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPEN</td>
<td>RXA@FKEY 00160</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>7321</td>
<td>0</td>
<td>114</td>
<td>7321</td>
<td>0</td>
<td>114</td>
<td>7321</td>
<td>0</td>
</tr>
</tbody>
</table>
```

The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

- COLLID
- CONTOKEN
- IS_GETPAGE
- ISCT
- PROGRAM
- SECT#
- SQL
- SQLCALL
- STMT#
- TB_GETPAGE
- TBCT
- TOT_TBGETP
- TOT_ISGETP
- WF_GETPAGE

See the “Display Columns” chapter for descriptions of the data columns.

Table/SQL Index Display

When you use the S line command to select a display line on the Table SQL Activity Display, the Table/SQL Index Display appears.

The Table/SQL Index Display shows the indexes for the table referenced by the selected SQL statement. If multiple index access was used, multiple output lines display. You can determine the number of index getpage requests issued and the number of table getpage requests issued using the index from the display output. The table getpage count on the display could be zero, which may indicate that the SQL statement used index only access.
The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

<table>
<thead>
<tr>
<th>INDEX</th>
<th>IS_GETPAGE</th>
<th>PCTIGP</th>
<th>PCTTGP</th>
</tr>
</thead>
<tbody>
<tr>
<td>TB_IS_RATIO</td>
<td>TB_GETPAGE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See the “Display Columns” chapter for descriptions of the data columns.

**SQL Table Activity Display**

To access the SQL Table Activity Display, do one of the following:

- Type the A line command to select a display line on the Table SQL Activity Display and press Enter.
- Type the S line command to select a display line on the Table SQL/Multiple Tables Accessed display or the Table SQL/Workfile Accessed display and press Enter.

The SQL Table Activity Display shows all tables and indexes that have been referenced by the SQL statement you want to view. The output is sorted by table name. Every table and index accessed display on separate lines. A line of output for each index used in referencing the table follows any table appearing in the output that was referenced by an index. The output columns enable you to understand the quantity of getpage activity incurred and how table data is being accessed.

The first line of output indicates it is a table entry, because the index column is empty. This output indicates that table SYSTABLES was read by index access because the TB_IDX_GP column is populated. This column reflects the number of table getpage requests issued using index access.

The second line of output is for an index on SYSTABLES. The index name is DSNDTX02. The output column IS_GETP indicates that 821 getpages were issued for the index. The IS_TBGETP count indicates that 7481 table getpage requests were issued using the index. This count matches the table display line of the preceding display.

This display provides a powerful method for understanding SQL statement getpage activity and the SQL access path used in obtaining the result set required by the SQL statement.

Press F3 (End) to return to the Table SQL Activity Display.
The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

<table>
<thead>
<tr>
<th>CREATOR</th>
<th>INDEX</th>
<th>IS_GETP</th>
<th>IS_TBGETP</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE</td>
<td>TB_IDX_GP</td>
<td>TB_LNK_GP</td>
<td>TB_SEQ_GP</td>
</tr>
</tbody>
</table>

See the “Display Columns” chapter for descriptions of the data columns.

**Table/SQL Plan Display**

When you use the P line command to select a display line, the Table/SQL Plan Display appears.

The Table/SQL Plan Display formats the plans that have executed the SQL statement being viewed. The output display columns help you determine which plans account for most SQL activity that references the table and the amount of getpage activity incurred on the table and its related indexes. Press F3 (End) to return to the Table SQL Activity Display.

The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

| IS_GETPAGE | PCTIGP | PCTTGP | PLANNAME |
| SQL        | TB_GETPAGE |         |

See the “Display Columns” chapter for descriptions of the data columns.

**SQL Call Text Display**

When you use the Q line command to select a display line, the SQL Call Text Display appears. See SQL Call Text Display.
Index SQL Activity Display

The Index SQL Activity Display lets you view all SQL that has referenced the index during the collection interval. You can determine what SQL accounts for the majority of index use. You can also determine table getpage activity incurred by the SQL that is referencing the index. Index only access SQL also appears in the output.

To access the Index SQL Activity Display from CA Subsystem Analyzer, enter option A next to the tablename on the Analyzer Indexspace Activity Display and press Enter. The Index SQL Activity Display appears:

```
> -------- DETECTOR Index SQL Activity Display --------
COMMAND ==> Scroll ==> PAGE
LINE 1 OF 7
DB2 SSID ==> ssid        Database ==> DSNDB06        Tablespace ==> SYSDBASE
Table  ==> SYSTABLES                               Tablecreator ==> SYSIBM
Index   ==> DSNDTX01                                Total/Avg    ==> T T/A
Interval Time ==> 02:00                           Interval Elapsed ==> 32:52.77
-------------------------------------------------------------------------------
S -View All Tables/Indexes, Q -View SQL Text, E -Explain

SQLCALL               PLCNT               PROGRAM  STMT#  SECT#  PCTIGP  SQL
OPEN                   1 RUASLT3    00378 00007  66.2%          4         90          0
FETCH                  1 RXA@TBV5 00210 00001  22.1%         44         30         6
FETCH                  1 RXA@ALT1 03714 00003   2.9%          4          4          2
PREPARE                1 RXASSQL3 00510 00001   2.9%          4          4          2
PREPARE                1 RUASDBF2 01901 00001   2.9%          4          4          2
SELECT                 1 UTAGLCEX 00902 00002   1.5%          2          2          1
OPEN                   1 RUASLT3    00362 00005   1.5%          2          2          0
*******************************************************************************
```

The following data columns display collection data for the current collection interval only. Use F10 (Left) or F11 (Right) to view the columns. These columns are for display only:

- COLLID
- CONTOKEN
- IS_GETPAGE
- PCTIGP
- PLCNT
- PROGRAM
- SECT#
- SQL
- SQLCALL
- STMT#
- TB_GETPAGE
- TYPE
- VERSION

See the “Display Columns” chapter for descriptions of the data columns.
View Historical Collection Data

You can view CA Detector collections that have been externalized (written) to a datastore for a specific subsystem or data sharing group. You can view interval summaries for standard, exception SQL, or SQL error collection.

You can view collection data and SQL error activity data for a specific subsystem or data sharing group that has been externalized (saved) to a datastore.

**Note:** This activity is optional and must be specified when collection is started. Externalizing activity and SQL error activity data is optional and must be specified when the collection is started. To collect exception SQL data, you must externalize the data.

**Follow these steps:**
1. Do any of the following:
   - Type 2 in the Option line on the CA Detector Main Menu and press Enter.
     The Detector Datastore Display panel appears. This panel lists the datastores that have been created on the specified subsystem. From this panel, you can select a datastore and perform the following tasks:
     - View interval data
     - See when the datastore was used during the collection interval
     - Delete a datastore
   - Type 10 in the Option line on the CA Detector Main Menu and press Enter.
     The Detector Data Sharing Datastore Display panel appears. Datastores are only shown for members of a data sharing group where at least one of the member’s had data that was collected using the sysplex time interval.
   - Type 5 in the View History field on any current activity panel where it is available and press Enter.
     A panel appears where you can select the datastore containing the collection data you want to view.
2. Select a datastore and press Enter.

The Datastore Interval Display or DS Group Datastore Interval Display panel appears. You can view interval summaries for standard, exception SQL, SQL errors, and collection, and objects (not applicable for data sharing groups). Interval data appears for each activity collection interval in the datastore or each sysplex collection interval that contains data from the data sharing group members. This includes the start date, time, and length, the number of commits and aborts, number of SQL calls and getpages, total amount of INDB2 CPU time consumed, total number of unique plans, programs, and SQL statements that were executed in the interval. The Members column specifies how many member datastores contain data for the collection interval for a data sharing group.

**Note:** You can also delete an interval. All data for the selected interval is purged and the space occupied by the deleted data is reused by subsequent collections. Use the View Type field to view intervals on the Exception SQL or SQL Error Interval Display panels.

3. Select an interval or orange of intervals and press Enter.

The Planname Summary Display panel appears.

**Note:** Use the View Type field to view standard, exception, error, or object collection activity. The panels that are available for viewing current interval data are also available for viewing historical data.

**More information:**

- [View Standard Collection Activity](#) (see page 64)
- [View Exception SQL Collection Activity](#) (see page 93)
- [View SQL Errors Collection Data](#) (see page 106)

---

**View Historical Data Sharing Collection Data**

You can view CA Detector collections that have been externalized (written) to a datastore for a specific subsystem or data sharing group. You can view collection data and SQL error activity data for a specific subsystem or data sharing group that has been externalized (saved) to a datastore.

**Note:** This activity is optional and must be specified when collection is started. Externalizing activity and SQL error activity data is optional and must be specified when the collection is started. To collect exception SQL data, externalize the data.
Follow these steps:

1. Do either of the following:
   - Type 10 in the Option line on the CA Detector Main Menu and press Enter.
     The Detector Data Sharing Datastore Display panel appears. Datastores are only shown for members of a data sharing group where at least one of the member's had data that was collected using the sysplex time interval.
   - Type S in the View History field on any current activity DS Group panel where it is available and press Enter.
     A panel appears where you can select the datastore containing the collection data you want to view.

2. Select a datastore and press Enter.

   The DS Group Datastore Interval Display panel appears. You can view interval summaries for standard and SQL errors collection. Interval data appears for each sysplex collection interval that contains data from the data sharing group members for the selected datastore. The Members column specifies how many member datastores contain data for the collection interval. This includes the following information and helps you evaluate application workload at various times of the day for the data sharing group you are viewing:
   - Start date, time, and length
   - Number of commits, aborts, SQL calls, and getpages
   - Total amount of INDB2 CPU time consumed by these data sharing group members within the interval

   Note: You can also delete an interval. All data for the selected interval is purged and subsequent collections reuse the space occupied by the deleted data. Use the View Type field to view intervals on the Exception SQL or SQL Error Interval Display panels.

3. Specify one of the following commands:
   - S
     Shows interval data for the selected datastore on the DS Group Planname Summary Display panel to view the data sharing group statistics for the selected interval. Use the View Type field to view standard and error collection activity. Use the View By field to sort the data by plan, program, keys, and SQL.
   - M
     Shows a list of member intervals that have been included for the selected sysplex interval on the DS Group Datastore Interval Members panel. You can view the collection intervals for the individual member subsystems that are aggregated together for the data sharing group panels.

   Press Enter.

   The data sharing group collection activity panel appears for the specified command.
View Audit Records

Every time CA Detector starts or stops collection, an audit record is written to the control file. The audit record contains a history of the collection activity, such as collection start and stop times and dates, and how collection was terminated.

You can view CA Detector collections that have been externalized (written) to a datastore for a specific subsystem or data sharing group. You can view interval summaries for standard, exception SQL, or SQL error collection.

You can view collection data and SQL error activity data for a specific subsystem or data sharing group that has been externalized (saved) to a datastore.

**Note:** This activity is optional and must be specified when collection is started. Externalizing activity and SQL error activity data is optional and must be specified when the collection is started. To collect exception SQL data, you must externalize the data.

**Follow these steps:**

1. Do any of the following:

   - Type 2 in the Option line on the CA Detector Main Menu and press Enter.
     
     The Detector Datastore Display panel appears.

   - Type 10 in the Option line on the CA Detector Main Menu and press Enter.
     
     The Detector Data Sharing Datastore Display panel appears. Datastores are only shown for members of a data sharing group where at least one of the member's has data that was collected using the sysplex time interval.

   - Type S in the View History field on any current activity panel where it is available and press Enter.
     
     A panel appears where you can select the datastore containing the collection data you want to view.

2. Type A next to a datastore and press Enter.

   The Collection Audit Display panel appears. A row of data is shown for each time collection that was started and stopped. For example, if you specify a time limit of 10 minutes when you start collection, and collection ends normally, there will be one row for the 10-minute collection session. If you stop collection using the Terminate SSID Collection option from the CA Detector Main Menu, on termination, an audit record will be written to the datastore and a row for the session appears on this panel.
3. Type E next to an audit record and press Enter.

   If collection terminated normally, the following message appears:
   
   DT139E: NO ERROR INFORMATION LOCATED FOR SELECTED ENTRY
   
   If collection terminated abnormally, the Collection Audit Error panel appears. This panel shows the original message issued.

4. Press F3 when you have finished reviewing the message.

   The Collection Audit Display panel reappears.
Chapter 5: Using Collection Profiles

This chapter describes how to create, view, modify, and delete collection profiles and view and modify global defaults, application groups, resource groups, exclusion groups, and additional keys groups.

This section contains the following topics:

- **Collection Profiles** (see page 137)
- **Create Collection Profiles** (see page 138)
- **Global Defaults** (see page 139)
- **View/Modify Global Defaults** (see page 139)
- **Application Groups** (see page 141)
- **Resource Groups** (see page 154)
- **Exclusion Groups** (see page 159)
- **Additional Keys Group** (see page 164)
- **Customize Additional Keys Group Collection** (see page 165)
- **Delete a Collection Profile** (see page 167)

### Collection Profiles

You can limit the amount of activity captured on your subsystems by creating collection profiles. Collection profiles let you tailor SQL and SQL error collection by specifying threshold values or excluding conditions that are of no interest. For example, you can set CPU time, getpage, or rows returned thresholds that limit the SQL activity captured.

Using collection profiles, you can perform the following tasks:

- Control captured SQL activity
- Use capture thresholds to specify when a request should be considered for data capture
- Define SQL call termination thresholds
- Preempt the execution of an SQL request
- Tailor SQL error collection
- Tailor SQL exception collection for specific applications
- Control the type of exception data collected for specific applications or exclude certain applications from exception collection using application group options.
Create Collection Profiles

Before collection limits and thresholds can be set, the collection profile must be created.

Each collection profile resides in its own VSAM KSDS variable record length data set. Various record types are used within the collection profile data set. The collection data set contains control information and global defaults, as well as application group, resource group, and collection exclusion group collection profile information. The default data set size provided when you create a collection profile is 10 primary and 10 secondary cylinders. The defaults provided should be adequate for most sites.

Note: You can change your collection profile without stopping collection or waiting for collection to complete. You can also load new active collection profile definitions using the PROF(DTR) modify command from the MVS console. The collection profile needs to be created before you can define collection limits and thresholds.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 (View Collection Profiles) and press Enter.
   The Collection Profiles Display panel appears. This panel shows one line for each collection profile created for the DB2 SSID. The data columns provide information about the collection profiles. Each row represents a different profile. Use F10 (right) and F11 (left) to view this information.
   Note: You can use the L line command to load the current profile with values from an existing collection profile.
3. Type S in the Create Profile field and press Enter.
   The Create Collection Profile Display panel appears.
4. Specify the DB2 SSID for which the collection profile will be used, the profile ID and optional profile description, and the profile data set criteria, and press Enter.
   The collection profile is created.
5. Press F3 (End).
   The Collection Profiles Display panel appears.
   The collection profile data set is created with the following name based on values specified during the creation process:

   high-level.PDTCPR.profile-id.profile-ssid

   CA Detector adds several sample application and resource groups to the profile. These groups contain typical settings and values for your review.

Global Defaults

Global defaults let you specify capture thresholds for exception SQL collection without having to define application groups. CA Detector uses the global defaults to limit the amount of exception SQL activity captured. They consist of the same SQL collection capture thresholds that you can specify using resource groups, plus an additional field that limits the number of exception requests captured per collection interval and the maximum amount of host variable data and SQL text captured per exception request.

If an SQL request does not fall within any application group plan or program entry, CA Detector applies the global defaults to the request.

View/Modify Global Defaults

Use of global defaults is optional. You may choose to enable or disable global default collection of dynamic SQL exceptions, static SQL exceptions, and Host Variable data. These options let you limit SQL exception monitoring to only those plans and programs associated with collection profile application groups.

When you create a collection profile, CA Detector generates global defaults for you. You can accept these defaults or modify them for your installation.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type $ in the Set Global Defs field and press Enter.
   The View Collection Profile Global Defaults panel appears. This panel provides the current global defaults for the collection profile.

5. Press Enter to view additional global default options.
   The second View Collection Profile Global Defaults panel appears.

6. Type $ in the Update Global Defs field to modify the current values and press Enter.
   The Update Collection Profile Global Defaults Display panel appears.

7. Specify values in any or all of the default dynamic and static SQL collection capture threshold fields (CPU time, getpage requests, fetch count) and press Enter.
   **Important!** If you specify 0 (zero) in all three default SQL collection capture threshold fields, all SQL activity is captured, but you will also incur an excessive application overhead and CA Detector storage use. We do not recommend you do this.

   The next Update Collection Profile Global Defaults Display panel appears.

8. Complete the following fields:

   **Max Exception Requests Captured Per Collection Interval**
   Limits the total number of SQL exceptions you want collected during a single collection interval. When this limit is reached, SQL exception monitoring stops for the remainder of the collection interval. If you enter 0 (zero), no limit is applied. This is not recommended and can result in excessive application overhead and CA Detector datastore storage usage.

   **Max Host Variable Data and SQL Text Captured per Exception Request**
   Controls the maximum amount of SQL text or Host Variable data retained for each SQL exception. If the length of SQL text associated with an SQL exception exceeds the amount specified, the SQL text is truncated. If the total amount of storage required to retain the Host Variables associated with an SQL exception exceeds the amount specified, the Host Variable data is not retained.

   The supplied default values should be sufficient for most applications. However, you might need to increase these amounts if you are collecting exceptions for SQL with a large number of host variables or inputting CA Detector exception SQL text into CA Plan Analyzer and have applications that generate large dynamic SQL statements.

   These fields let you trade off the importance of retaining the SQL data with the storage costs associated with externalizing SQL exceptions. You can retain any amount of dynamic SQL text up to the maximum of 32767 bytes, however, limitations in the DB2 execution time environment limit the maximum amount of static SQL text retained to 4000 bytes even if a higher capture limit is specified.

   If you enter 0 (zero) in either of these fields, then none of the associated SQL data is retained when creating an SQL exception.
Enable Global Dynamic SQL, Static SQL, and Host Variable Collection

Enables or disables global default collection. Set these fields to N to disable associated global default collection and limit exception collection activity to only SQL associated with application groups. This is useful if you want to target SQL performance monitoring to specific applications. In this situation, disabling the global collection defaults reduces application overhead and external storage use associated with monitoring and collecting non-critical SQL exceptions.

You must still have the associated exception collection options enabled when starting collection for these collection profile options to have effect.

Press Enter.

The Update Collection Profile Global Defaults panel appears.

9. Press Enter to confirm your updates.

The View Collection Profile Defaults Display panel appears. A message on this panel indicates that the global defaults have been successfully updated for the collection profile.

Application Groups

Application groups are used to define logical groupings of application plans or programs, user IDs, or DB2 locations. Once you have defined application groups, you can manage collection for the applications by group, rather than individually.

Application groups and resource groups work together to control exception SQL collection and to provide resource limiting of SQL activity. When you create an application group, you must specify a resource group to use with the application group. Within the application group, you can specify plan entries, program entries, or both to define the group. You can use plan name, program name, location, correlation ID, collection ID, primary authorization ID, or a combination of these to define the application group entries.

You do not have to create an application group to start exception SQL collection. When you create a collection profile, global default settings are created for you. These defaults are used to control collection of exception SQL requests.

While you do not have to define application groups, you may find it beneficial to do so. Application groups let you tailor SQL exception collections to your applications different performance requirements. You can also use application group options in combination with global default options to limit the overhead of exception collection to only those applications that require performance monitoring.

You might want to begin working with collection profiles by first defining a resource group. You can then use the same resource group for multiple application groups.
More information:

View/Modify Resource Groups (see page 154)

How Application Groups Are Applied

It is important to understand how application group plan and program entries are applied when CA Detector encounters an SQL request. You should review this information before attempting to define your own application groups for the first time.

When CA Detector encounters an SQL request, it analyzes all program and plan entries in the collection profile for criteria matching the SQL request. One of the following happens:

- If CA Detector finds a matching plan or program entry, then it applies the entry’s resource group thresholds to the SQL request.
- If CA Detector does not find a matching plan or program entry, then it applies the global defaults to the SQL request.

The general sequence in which the entries are examined is as follows:

- If there are any program entries in the collection profile, CA Detector examines them for a match first. If it locates a match, CA Detector uses the program entry’s resource group thresholds for the SQL request.
- If there are no program entries or none of the program entries match, then CA Detector searches the plan entries for a match. If CA Detector finds a matching plan entry, CA Detector uses the plan entry’s resource group thresholds for the SQL request.
- If there are no matching plan entries or if there are no application groups defined for the collection profile, CA Detector applies the global defaults to the request.
- When CA Detector loads a collection profile, it merges and orders the application group entries so that more specific program and plan entries are searched before the more generic entries.
**Program Entries**

When CA Detector encounters an SQL request, it searches the collection profile for a matching program. All program entries in the collection profile are examined. The fields compared for a match are listed in the following table:

<table>
<thead>
<tr>
<th>Order and precedence</th>
<th>Fields analyzed for match</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Location, correlation ID, collection ID, auth ID, and program</td>
</tr>
<tr>
<td>2</td>
<td>Location, correlation ID, collection ID, and auth ID</td>
</tr>
<tr>
<td>3</td>
<td>Location, correlation ID, program, and auth ID</td>
</tr>
<tr>
<td>4</td>
<td>Location, correlation ID, collection ID, and program</td>
</tr>
<tr>
<td>5</td>
<td>Location, correlation ID, and collection ID</td>
</tr>
<tr>
<td>6</td>
<td>Location, correlation ID, and auth ID</td>
</tr>
<tr>
<td>7</td>
<td>Location, correlation ID, and program</td>
</tr>
</tbody>
</table>

After CA Detector has examined all program entries, it uses the program entry that matches with the highest precedence according to the table for the SQL request.

For example, suppose a dynamic SQL request search results in two matching program entries. The first match is a program entry that matches the location, collection ID, and authid, and the second match is a program entry that matches the location, collection ID, and the program name. CA Detector would use the program entry with the matching location, collection ID, and auth ID for the dynamic SQL request because that program entry has the highest precedence according to the table.

**Note:** Except for the Location field, the match can be an exact character-for-character match or can be using masking characters.

Masking characters are not allowed in the Location field. The Location field is matched as follows:

- The SQL request originates from a local location, and the Location field in the program entry is blank (which implies the local location).
- The SQL request's location matches the Location field character for character in the program entry.
- The Location field in the program entry is a single % (which matches any location including the local location).

If a matching program entry is found, then the program entry's resource group thresholds are applied to the SQL request. If no program entry matches are found, CA Detector searches plan entries as described next in Plan Entries.
Plan Entries

After searching the program entries and finding no matches, the collection profile is searched for a matching plan entry. All plan entries in the collection profile are examined. The fields compared for a match are listed in the following table:

<table>
<thead>
<tr>
<th>Order and Precedence</th>
<th>Fields analyzed for match</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Location, correlation ID, plan name, and auth ID</td>
</tr>
<tr>
<td>2</td>
<td>Location, correlation ID, and auth ID</td>
</tr>
<tr>
<td>3</td>
<td>Location, correlation ID, and plan name</td>
</tr>
<tr>
<td>4</td>
<td>Location and correlation ID</td>
</tr>
</tbody>
</table>

When CA Detector has examined all plan entries, it uses the plan entry that matches with the highest precedence according to the table for the SQL request.

For example, suppose a dynamic SQL request search results in two matching plan entries. The first match was a plan entry that matched the location, plan name, and auth ID, and the second match was a program entry that matched the location and the plan name. CA Detector would use the plan entry with the matching location, plan name, and auth ID for the dynamic SQL request because that plan entry has the highest precedence according to the table.

**Note:** Except for the Location field, the match can be an exact character-for-character match or can be using masking characters.

Masking characters are not allowed in the Location field. The Location field is matched as follows:
- The SQL request originates from a local location, and the Location field in the plan entry is blank (which implies the local location).
- The SQL request’s location matches the Location field character for character in the plan entry.
- The Location field in the plan entry is a single % (which matches any location including the local location).

If a matching plan entry is found, then the plan entry’s resource group thresholds are applied to the SQL request. If no plan entry matches are found, then the global defaults thresholds are applied to the SQL request. See Global Defaults for more information about global defaults.
View/Modify Application Groups

Application groups let you define logical groupings of application plans or programs, user IDs, or DB2 locations. Once you have defined application groups, you can manage collection for the applications by group rather than individually.

You can view, modify, and delete application groups and view information about resource groups and SQL error collection profiles.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.
   The data columns provide information about the collection profile. Each row is a separate application group within the profile. Use F10 (Right) and F11 (Left) to scroll through the fields.
4. Do one of the following:
   - Type S in the Set Global Defs field to view or modify global defaults.
   - Type S in the Add Application Group field to add an application group to the collection profile.
   - Type S in the line command field next to an application group to view or modify the application group plan or program entries.
   - Type D in the line command field next to a resource group to delete it from the collection profile.
   - Type R, X, or K in the View Group Type field to access another group type.
   Press Enter.
   Additional panels appear for you to view or modify the application group or to set global defaults.
View/Modify Application Group Plan Entries

You can add, view, modify, and delete plan entries contained in an application group.

Follow these steps:
1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
   **Note:** You can view another group type by changing the specified value in the View Group Type field.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears. The data columns provide information about the plan entries within the application group. Each row is a separate plan entry within the selected application group. Use F10 (Right) and F11 (Left) to scroll through the fields.
5. Do one of the following:
   - Type S in the ADD Plan Entry field to add new plan entries to the application group.
   - Type S in the line command field next to an application group plan entry to view a plan entry in report format.
   - Type U in the line command field next to an application group plan entry to change the plan entry definition.
   - Type D in the line command field next to an application group plan entry to delete it from the application group.
   Press Enter.

Additional panels appear for you to view or modify plan entries for the selected application group.
Add Application Group Plan Entries

You can define new plan entries for the application groups.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.
5. Type S in the ADD Plan Entry field and press Enter.
   The Add Application Group Plan Entry panel appears.
6. Complete the fields for the new plan entry and press Enter.
   A verification message appears.
7. Confirm the addition of the new plan entry to the application group and press Enter.
   The new plan entry appears on the Add Application Group Plan Entries panel. A message confirms that the plan entry has been added for the application group.

View Application Group Plan Detail

You can view application group plan detail in a report format.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.

5. Type S in the line command field next to an application group plan entry and press Enter.
   The View Application Group Plan Detail panel appears.

**Update Application Group Plan Entries**

You can update application group plan entries to change the thread selection criteria, the name of the associated resource group, and enable or disable SQL exception monitoring.

**Follow these steps:**

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.

3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.

4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.

5. Type U in the line command field next to an application group plan entry and press Enter.
   The Update Application Group Plan Entry panel appears.

6. Complete the fields as follows:

   - Specify the thread selection criteria for the application group entry in the Location Name, Correlation ID, Planname, and Authid fields. The non-blank fields in the plan entry will be matched against the corresponding fields of the DB2 thread to determine if the associated resource group thresholds should be applied to the DB2 thread. You can specify values in some or all of these fields. You must specify at least one non-blank field.

   - Specify the name of the resource group associated with this plan entry in the Resource Group field. You must specify a resource group name.

   - To enable or disable the corresponding SQL exception monitoring activity for threads matching the plan entry selection criteria, enter data in the Dynamic Exceptions, Static Exceptions, and Host Variables fields. Using these fields you can control the type of exception data collected for specific applications. If you set all of these fields to N, exception monitoring for matching threads is bypassed.
Press Enter.
A verification message appears.

7. Press Enter to confirm the update.

The View/Modify Application Group Plan Entries panel appears. A message appears to confirm the update to the plan entry.

**View/Modify Application Group Program Entries**

You can add, view, modify, and delete program entries contained in an application group.

**Follow these steps:**

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
   **Note:** You can view resource, SQL, and keys groups by changing the value of the View Group Type field.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.
5. Type G in the View Entry Type field to view, modify, or delete program entries for the selected application group and press Enter.
   The View/Modify Application Grp Program Entries panel appears. The data columns provide information about the program entries within the application group. Each row is a separate program entry within the selected application group. Use F10 (Right) and F11 (Left) to scroll through the fields.
6. Do one of the following:
   - Type S in the ADD Pgm Entry field to add new program entries to the application group.
   - Type S in the line command field next to an application group program entry to view the program entry in report format.
- Type **U** in the line command field next to an application group program entry to change the program entry definition.

- Type **D** in the line command field next to an application group program entry to delete it from the application group.

Press Enter.

Additional panels appear for you to view or modify program entries for the selected application group.

**Add Application Group Program Entries**

You can add new program entries for an application group.

**Follow these steps:**

1. Type **4** on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type **1** and press Enter.
   The Collection Profiles Display panel appears.

3. Type **S** next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.

4. Type **S** next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.

5. Type **G** in the View Entry Type field and press Enter.
   The View/Modify Application Grp Program Entries panel appears.

6. Type **S** in the ADD Pgm Entry field and press Enter.
   The Add Application Group Program Entry panel appears.

7. Complete the fields for the new program entry and press Enter.
   A verification message appears.

8. Press Enter to confirm the new program entry.
   The Add Application Group Program Entries panel appears. A message indicates that the new program entry has been added to the application group.
View Application Group Program Detail

You can view application group program detail in a report format.

Follow these steps:
1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.
5. Type G in the View Entry Type field and press Enter.
   The View/Modify Application Grp Program Entries panel appears.
6. Type S in the line command field next to an application group program entry and press Enter.
   The View Application Group Program Detail panel appears. You cannot update the fields on this panel. They are for information only.

Update Application Group Program Entries

You can update application group program entries to change the thread selection criteria, the name of the associated resource group, and enable or disable SQL exception monitoring.

Follow these steps:
1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.
4. Type S next to an application group and press Enter.
   The View/Modify Application Group Plan Entries panel appears.
5. Type G in the View Entry Type field and press Enter.
   The View/Modify Application Grp Program Entries panel appears.
6. Type U in the line command field next to an application group program entry and press Enter.

The Update Application Group Program Entry panel appears.

7. Complete the fields as follows:
   - Specify the thread selection criteria for the application group entry in the Location Name, Correlation ID, Planname, and Authid fields. The non-blank fields in the program entry will be matched against the corresponding fields of the DB2 thread to determine if the associated resource group thresholds should be applied to the DB2 thread. You can specify values in some or all of these fields. You must specify at least one non-blank field.
   - Specify the name of the resource group associated with this program entry in the Resource Group field.
   - Specify values in the Dynamic Exceptions, Static Exceptions, and Host Variables fields to enable or disable the corresponding SQL exception monitoring activity for threads matching the program entry selection criteria. Using these fields you can control the type of exception data collected for specific applications. If you set all of these fields to N, exception monitoring for matching threads is bypassed.

Press Enter.

A verification message appears.

8. Press Enter to confirm the update.

The View/Modify Application Grp Program Entries panel appears. A message confirms that the program entry has been updated for the application group.

Create an Application Group

Your application groups should be efficient and easy to administer. When you create application groups and their corresponding plan and program entries, note the following administration considerations:

- Create an application group using program entries or plan entries, but not both. While you can use both types of entries in an application group, it is easier to administer if the entries are all the same type.
- Place related plans or programs for an application in one application group. This helps to eliminate or minimize conflicting entries across application groups.
- Limit the number of entries in your application groups using masking. For example, if you have several plans that you want to include that start with PR, create one plan entry specifying PR% in the Planname field, instead of creating a plan entry for each plan.
Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.

3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears.

4. Type S in the Add Appl Group field and press Enter.
   The Add Collection Profile Application Group Display panel appears.

5. Complete the following fields:
   
   **Application Group Name**
   (Required) Specifies a name for the application group. The name must be unique in the collection profile.

   **Limits:** 1 to 18 alphanumeric characters.

   **Application Group Description**
   (Optional) Describes the application group.

   **Limits:** 1 to 40 alphanumeric characters.

   Press Enter.
   The Add Application Group Plan/Program Entries panel appears.

6. Complete the fields as follows:

   ■ To add plan entries, specify values in the Planname or Primary Authid field, and you must enter a resource group name in the Resource Group Name field.

   ■ To add program entries, specify values in the Collection Id, Program Name, or Primary Authid field, and you must enter a resource group name in the Resource Group Name field.

   You can add plan or program entries, or both. You are not required to enter both.

   **Note:** When you initially create plan or program entries, the collect dynamic SQL, static SQL, and host variable options are all defaulted to Y. To change these values, you must update the entries after they have been created.

7. Press Enter to process your selections.
   A verification message appears.
8. Press Enter to confirm the new additions.
   The Add Application Group Plan/Program Entries panel appears. A message indicates that the entries have been added to the application group. To make additional modifications to the new plan or program entry, return to the Add Application Group Plan/Program Entries panel.

Delete an Application Group

You can delete an application group from a collection profile.

Follow these steps:
1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type D next to an application group and press Enter.
   The Delete Application Group Verification panel appears.
5. Press Enter to confirm the delete request.
   The application group is deleted from the collection profile and removed from the list of application groups on the View/Modify Application Groups panel. A message confirms the deletion.

Resource Groups

Resource groups let you define the SQL thresholds and limits you want to use in your collection profiles.

View/Modify Resource Groups

You can define resource groups to perform the following tasks:

- Limit SQL collection activity using termination thresholds
- Control SQL execution
- Preempt SQL execution of statements that are too resource-intensive
Resource groups are not stand-alone entities; they work only with application groups. One resource group can be linked to many application groups. Therefore, you can define a few resource groups for use with all your application groups. For example, to use the same set of thresholds for all your application groups, define one resource group that contains those thresholds, and then specify that resource group in your application groups.

When you create a collection profile, sample resource groups are generated for you. You can examine these groups to understand how resource groups are defined, modify them for your own use, or delete them.

You can add, view, update, and delete resource groups for use in collection profiles.

**Follow these steps:**

1. Type 4 on the CA Detector Main Menu and press Enter.
   
   The Collection/Reporting Profiles Menu panel appears.

2. Type 1 and press Enter.

   The Collection Profiles Display panel appears.

3. Type S next to a profile name and press Enter.

   The View/Modify Application Groups panel appears showing application group information for the profile.

4. Type R in the View Group Type field and press Enter.

   The sample groups for the collection profile appear on the View/Modify Resource Groups panel. You can add, view, update, and delete entries.

   **Note:** This panel also appears when you type S in the View Res Groups field on the View/Modify Application Group Program Entries and View/Modify Application Grp Program Entries panels.

   The data columns provide information about the resource groups in the selected collection profile. Each row is a separate resource group within the profile. Use the right and left scroll (F10 and F11) keys to view the data.

5. **Do one of the following:**
   - Type S in the Add Res Group field to add a new resource group to the collection profile.
   - Type S in the line command field next to a resource group to view the resource group definition in a report format.
   - Type U in the line command field next to a resource group to update the resource group.
Resource Groups

- Type D in the line command field next to a resource group to delete it from the collection profile.
- Type A, X, or K in the View Group Type field to access another group type.

Press Enter.

Additional panels appear for you to view or modify resource groups for the selected collection profile.

More information:

Application Groups (see page 141)

Add Collection Profile Resource Groups

You can add a new resource group for use in the specified collection profile.

Capture thresholds define the level of resource use by a request that warrant collection data capture for the request. Exception SQL data capture provides all major accounting information for the execution of the initial request, as well as for the SQL calls executed as a result of the request.

Termination thresholds let you specify resource use limits that, when met, cause the SQL request to be terminated.

Different applications typically have different performance requirements. Application groups let you specify thread criteria identifying specific plans and programs, or related groups of plans and programs. You can associate different exception thresholds with each of these groups, letting you collect only those SQL statements as exceptions whose performance is outside the expected range for the specific applications.

Resource thresholds let you can specify limits relating to SQL execution cost. You can preempt the execution of an SQL request if the PREPARE of that SQL statement indicates the cost is higher than you specified in the collection profile.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type $ next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type R in the View Group Type field and press Enter. The View/Modify Resources Groups panel appears.

5. Type S in the Add Res Group field and press Enter. The Add Collection Profile Resource Group panel appears.

6. Complete the following fields:

   **Resource Group Name**
   Specifies a unique name for the resource group for the specified collection profile. This field is required.
   **Limits:** 1 to 18 alphanumeric characters.

   **Resource Group Desc**
   Specifies a description for the resource group.
   **Limits:** 1 to 40 alphanumeric characters.

   **SQL Collection Capture Thresholds**
   Specifies the thresholds that must be met before data on SQL requests is collected. For example, if you specify a CPU time of 10 seconds, an SQL request has to consume more than 10 seconds of CPU time before CA Detector collects data on the request.

   You can specify values for one, two, or all of these fields. If any of these thresholds is met, the data is collected. If you set these fields to zeros, CA Detector ignores these fields and uses the global default collection capture thresholds instead.

   **SQL Termination Thresholds**
   Specifies the thresholds that must be met before an SQL request is terminated. For example, if you set Getpage Reqs to 500000, an SQL request would have to perform more than 500,000 getpages before the SQL request would be terminated.

   You can specify values for one, two, or all of these fields. If any threshold is met, the SQL request is terminated with a -905 return code. Leave these fields set to zero if you do not want SQL requests to be terminated.

   **Note:** Whether the termination thresholds have been exceeded is determined at the end of the SQL request. CA Detector will not terminate SQL requests during their execution, but sets a -905 error code upon completion of the SQL request.

   Because CA Detector does not interrupt an SQL request during execution, the SQL termination thresholds might be useful in interrupting an application that is in a fetch loop, but are not useful as a means of interrupting long running SQL statements, such as an update or singleton select.
Preempt SQL Execution Thresholds

Preempts the execution of the SQL call based on the value specified in the Timeron Units field. This field is used to estimate the cost of the SQL statement. Specify the timeron exponent value that the SQL call must exceed. This value is compared to the exponent of the timeron value generated during the PREPARE of the SQL request. For example, if you specify 30 and an SQL request generates a timeron of 0.6043 E40, the exception SQL request is preempted with a -905 return code. Leave this field set to zeros if you do not want SQL requests to be preempted.

Note: Specifying a value lets you preempt only SQL requests originating as a result of a PREPARE. It does not affect execution of EXECUTE IMMEDIATE requests.

Press Enter.

A message appears confirming that the resource group has been added to the collection profile.

7. Press F3.

The new entries appear on the View/Modify Resource Groups panel. You can update, delete, and view entries as needed, or repeat the previous steps to add additional entries.

View Collection Profile Resource Groups

You can view a resource definition in a report format.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type R in the View Group Type field and press Enter.
   The View/Modify Resource Groups panel appears.
5. Type S in the line command field next to a resource group and press Enter.
   The View Collection Profile Resource Group Display panel appears. The fields are informational only and cannot be updated.
Update Collection Profile Resource Groups

You can change a resource group definition including the description, SQL collection capture thresholds, SQL collection termination thresholds, and preempt SQL execution thresholds.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter. The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter. The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter. The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type R in the View Group Type field and press Enter. The View/Modify Resource Groups panel appears.
5. Type U in the line command field next to a resource group and press Enter. The Update Collection Profile Resource Group panel appears.
6. Modify the fields as needed and press Enter. A verification message appears.
7. Verify the changes and press Enter. A message appears confirming the resource group has been updated.

Exclusion Groups

The collection exclusion groups limit the number of plan names collected, the amount of SQL error activity collected, and exclude selected negative SQL return codes.

Collection exclusion groups let you perform the following tasks:

- Limit collection activity by plan name, SQL error, and thread based keys (location name, correlation ID, plan name, and auth ID). For SQL requests meeting defined criteria, you can include or exclude standard activity, dynamic SQL, additional keys, and static or dynamic SQL table/index statistics collection activity.
- Modify SQL error collection profile information for the collection
Limit Collection Activity

You can limit collection activity by plan name, SQL error, and thread based keys (location name, correlation ID, and auth ID) in your collection profiles using collection exclusion groups. Additionally, for SQL requests meeting a defined criteria, the following statistics collection activity can be selected for inclusion or exclusion:

- CA Detector statistics collection
- Dynamic SQL statistics collection
- Additional keys statistics collection
- Standard activity table/index statistics collection
- Dynamic SQL table/index statistics collection

This processing must be activated by specifying Plan Excl/Incl List=Y on the Start Collection Display panel when collection is initialized.

Masking is supported when defining the plan names.

Follow these steps:

1. Type 4 on the CA Detector Main Menu and press Enter. The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter. The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter. The View/Modify Application Groups panel appears showing application group information for the profile.
4. Type X in the View Group Type field and press Enter. The Collection Exclusion Groups Menu panel appears.
5. Type 1 in the Command line and press Enter. The View/Modify Include/Exclude List Entry panel appears. You can add, view, update, and delete entries.
6. Type S in the ADD Entry field and press Enter. The Add Include/Exclude List Entry Display panel appears.
7. Complete the following fields as needed:

**Location**
Specifies the DB2 location from which the SQL requests originate. You can
include wildcards. If left blank, this value is replaced with % and matches any
location name. A non-DB2 location can be specified as <location>. The greater
than and less than symbols bracket the LUNAME of the originating location,
such as DB2/2 or DB2/6000.

**CORRID**
Specifies the correlation ID for the thread under which the SQL request is
made. You can include wildcards. If left blank, this value is replaced with % and
matches any correlation ID.

**PLANNAME**
Specifies the plan name under which the SQL request is made. You can include
wildcards. If left blank, this value is replaced with % and matches any plan
name.

**AUTHID**
Specifies the primary authorization ID for the thread under which the SQL
request is made. You can include wildcards. If left blank, this value is replaced
with % and matches any authorization ID.

**ACT/Enable Detector Statistics Collection**
Indicates whether the entry should be included (Y) or excluded (N) from the
collection. If N, all other filters are ignored.

**DYN/Enable dynamic SQL statistics collection**
Indicates whether dynamic SQL should be included (Y) or excluded (N) from the
collection for this entry.

**KEY/Enable Additional Keys Statistics collection**
Indicates whether user-defined key collection is enabled (Y) or disabled (N) for
this entry.

*Note:* If additional keys statistics collection is enabled when collection is
started, but disabled for one or more Include/Exclude List entries, the activity
for the excluded entries appears as "NULL" on the CA Detector key summary
panels.
**Exclusion Groups**

**TBL/Enable Standard Activity Table/Index Statistics Collection**

Indicates that table and index data is included (Y) or excluded (N) from the collection for this entry.

*Note:* CA Subsystem Analyzer must be active for this entry to be recognized.

**DTB/Enable Dynamic SQL table/Index Statistics collection**

Indicates that table and index data is included (Y) or excluded (N) from the collection for dynamic SQL statements for this entry.

*Note:* CA Subsystem Analyzer must be active for this entry to be recognized.

Press Enter.

A message appears confirming that the update was successful and that you can continue adding entries as needed.


The new entries appear on the Add Include/Exclude List Entry Display panel. You can view, update, and delete entries as needed, or repeat the previous steps to add additional entries.

**Define SQL Error Groups**

The SQL error group is used by CA Detector collection when SQL error collection is active. It lets you limit the amount of SQL error activity collected and exclude selected negative SQL return codes.

The SQL error group is automatically included when you create a collection profile. You do not create multiple SQL error groups as you do with application or resource groups. To customize SQL error collection, you modify a single SQL error group.

The SQL error group is distinct from application groups. If you do not create any application groups, the SQL error group still exists in your collection profile.

If SQL error collection is active, SQL error data is automatically collected on all negative SQL error codes. However, you can exclude certain SQL error codes by listing the codes in the SQL error exclude list.

**Follow these steps:**

1. Type 4 on the CA Detector Main Menu and press Enter.

The Collection/Reporting Profiles Menu panel appears.

2. Type 1 and press Enter.

The Collection Profiles Display panel appears.
Chapter 5: Using Collection Profiles

- Type S next to a profile name and press Enter.
  The View/Modify Application Groups panel appears showing application group information for the profile.

- Type X in the View Group Type field and press Enter.
  The Collection Exclusion Groups Menu panel appears.

- Type 2 and press Enter.
  The View/Modify SQL Error Group panel appears. From this panel, you can define the SQL error collection profile information.

- Complete the following fields:
  
  View SQLcode List
  Indicates whether you want to view or modify the SQL code exclude list for the collection profile. If you type S and press Enter, the View/Modify SQL Error Exclude List Display panel appears. You can modify the SQL code exclude list on this panel.

  Max SQL errors retained in a collection interval
  Specifies the maximum number of SQL errors you want to collect data on during a single collection interval. The default is 3000.

  Max SQL errors retained for a single SQLCODE
  Specifies the maximum number of SQL errors you want to retain data on for a single SQL error code. The default is 200. For example, if you specify the default, CA Detector collects data on -818 SQL error codes until 200 of these error codes have been collected.

  Press Enter.
  A message appears confirming that the SQL error collection profile information was successfully updated if changes are made.

**Exclude SQL Error Codes**

You can exclude SQL error conditions that are of no interest or limit the amount of SQL error information to be retained. This means they will not be captured during SQL error collection.

**Follow these steps:**

1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type S next to a profile name and press Enter.
   The View/Modify Application Groups panel appears showing application group information for the profile.

4. Type X in the View Group Type field and press Enter.
   The Collection Exclusion Groups Menu panel appears.

5. Type 2 and press Enter.
   The View/Modify SQL Error Group panel appears.

6. Type S in the View SQLCode List field to view or modify the SQL code exclude list for the collection profile and press Enter.
   The View/Modify SQL Error Exclude List panel appears.

7. Complete the following fields:
   
   **Specify SQLCODE to be ADDED to exclude list**
   - Specifies an SQL code (without the negative sign) that you want to exclude from CA Detector SQL error collection.

   **Specify SQLCODE to be DELETED from exclude list**
   - Specifies an SQL code (without the negative sign) that you want to delete from the exclude list.

   **Note:** Do not enter a leading minus sign in front of the SQL error code. All specified SQL error codes are assumed to be negative.

   Press Enter.
   The code is added to or deleted from the View/Modify SQL Error Exclude List panel.
   If no codes have been defined, the message No SQLCODES currently exist in the SQL error exclude list appears on the panel.

---

**Additional Keys Group**

The additional keys group lets you control what additional data should be collected and the types of DB2 connections on which the data is collected. You can view the plans, programs, and SQL executed by a selected key such as a specific user or connection type.
Customize Additional Keys Group Collection

The additional view by keys group is automatically included when you create a collection profile. It lets you specify additional thread identification criteria to organize standard activity collection data by additional View By groupings. It defines the additional view by keys to be collected and the connection types the keys are to be collected for if additional keys collection is enabled at collection start.

The additional keys group is distinct from application groups. If you do not create any application groups, the Additional Keys group still exists in your collection profile. If SQL error collection is active, SQL error data is automatically collected on all negative SQL error codes. However, you can exclude certain SQL error codes by listing the codes in the SQL error exclude list.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) in the Option line on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 (View collection profiles) in the Option line and press Enter.
   The Collection Profiles Display appears.
3. Select a profile from the list and press Enter.
   The View/Modify Application Groups panel appears.
4. Type K in the View Group Type field and press Enter.
   The View Collection Profile Additional Keys panel appears. The current settings of the additional standard activity View By keys being collected and the DB2 connection types the additional collection is enabled for appear.
5. Type S in the Update Additional Keys field and press Enter.
   The Update Collection Profile Additional Keys panel appears.
6. Complete the following fields:

   Enable collection for the following connection types
   Specifies the DB2 connection types for which additional View By keys are to be collected. Type Y (enable) or N (disable) for the following types:
   - TSO—DSN command processor, CAF, and RRSAF connections established from an online TSO address space.
   - Batch—DSN command processor, CAF, and RRSAF connections established from a non-TSO address space or from a batch TSO address space
   - CICS—CICS DB2 connections.
   - DL/I Batch—DL/I DB2 connections.
Customize Additional Keys Group Collection

- IMS—IMS DB2 connections including BMP, MPP, and control region.
- Distributed—Distributed DB2 connections including DRDA and DB2 private protocol.

**Note:** Activity for connection types excluded from Additional Keys Statistics collection appears as "NULL" on the CA Detector key summary panels.

**Enable collection of the following additional keys**

Specifies the additional standard activity View By keys to be collected. Type Y (enable) or N (disable) for the following keys:

- **Userid**—The DB2 connection user ID (AUTHID).
- **Correlation id**—The correlation ID, such as batch job name or CICS transaction name.
- **Location**—The remote location name or IP address.
- **Connection type**—The DB2 connection type, such as TSO or CICS.
- **Connection name**—The connection name, such as the CICS region name.
- **End user ID**—The end user ID that may be optionally specified for distributed and RRSAF connections.
- **End user TX**—The end user transaction ID that may be optionally specified for distributed and RRSAF connections.
- **End user WS**—The end user workstation ID that may be optionally specified for distributed and RRSAF connections.

Press Enter.

The updates to the additional keys group are saved.
Delete a Collection Profile

Collection profiles can be deleted when they are no longer needed.

Follow these steps:
1. Type 4 on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 1 and press Enter.
   The Collection Profiles Display panel appears.
3. Type D next to the profile you want to delete and press Enter.
   The Collection Profile Delete Verification panel appears.
4. Press Enter to confirm the delete request.
   A message appears to verify deletion of the profile. The collection profile is deleted.
Chapter 6: Using Reporting Profiles

This section contains the following topics:

Reporting Profiles (see page 169)
Space Requirements (see page 170)
Create Reporting Profiles (see page 170)
Set the Current Profile (see page 171)
View and Update Reporting Profiles (see page 172)
Delete Reporting Profiles (see page 176)

Reporting Profiles

Reporting profiles classify DB2 plans, packages, or users into composite application profiles. Once the reporting profiles are defined, you can view application use and resource consumption by application. The plan, package, and user data is grouped for you according to the profiles you define.

You have maximum flexibility in defining your reporting profiles. A reporting profile can be a collection of DB2 plans, a collection of DB2 packages, or a collection of DB2 users. You can define unique profiles for each DB2 subsystem, or define a profile datastore for several DB2 subsystems. You can create a profile datastores for your own use or share profiles with multiple users.

Reporting profiles let you view resource consumption according to the applications defined for your site, so that you can accurately determine which applications will most benefit from SQL tuning efforts. You can define a reporting profile to be a collection of DB2 plans or a collection of DB2 packages.

For example, the plans for shipping, receiving, and current stock could be assigned to the profile name INVENTORY for an inventory application profile. For exception SQL collection, you can also define a user profile, correlation ID profile, or a collection ID profile.

Detailed reporting capabilities let you quickly locate and identify the application programs and SQL statements that are the most resource-intensive. You are informed how frequently various application components are used as well as the related resource used. Consequently, you can easily identify application areas that will benefit most from application tuning efforts. Application performance analysis and tuning can therefore be concentrated in those areas in which you can achieve the largest positive impact.

You can create, view, update, and delete reporting profiles.
Space Requirements

Each reporting profile resides in its own data set. The reporting profile data set is a VSAM KSDS file. Each application entry defined in the profile data set is stored as a single VSAM record. The record length is fixed and is 827 bytes. The physical space required by a profile data set is relatively small.

To determine the space required, multiply the record length (827) times the number of application profile entries that are to be used. Add additional space to this computation to allow for insert, update, and delete activity, as well as the index space required for the data set. Adding an additional 20% to your total usually provides the necessary room for expansion and index component needs. Use a secondary allocation value to alleviate possible out-of-space conditions. The default data set size provided when you create a reporting profile is 10 primary and 5 secondary cylinders.

Create Reporting Profiles

Reporting profiles let you classify DB2 plans, packages, or users into composite application profiles.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 2 (View Reporting Profiles) and press Enter.
   The Reporting Profiles Display panel appears.
3. Type S in the Create Profile field and press Enter.
   The Create Reporting Profile Display panel appears.
4. Specify the DB2 SSID for which the profile will be used, the profile ID and optional description, profile type, and profile data set allocation criteria, and press Enter.
   The reporting profile is created.
5. Type data in the fields as needed and press Enter.
   Your selections are processed and a message appears to inform you that the profile has been defined and initialized.
6. Press F3 (End) to return to the Reporting Profiles Display panel.

The reporting profile data set is created with the name:

\[ \text{highlevel.PDTRPROF.profileid.profilessid} \]

**Note:** highlevel, profileid, and profilesid are the fields you entered on the Create Reporting Profile Display.

Once the reporting profiles are defined, application use and resource consumption can be viewed by application. CA Detector groups the plan, package, or user data for you.

---

**Set the Current Profile**

You can view collection data using a profile.

**Note:** If you type F in the View By field on any activity or exception SQL collection panel that contains the field, the current profile is used as set on the Reporting Profile Display panel. If the current profile type does not fit the type of collection data you are viewing, you are prompted to change it.

**Follow these steps:**

1. Do one of the following:
   - Type \textbf{S} in the Chg Profile field on the Application Profile or Exception SQL Profile panels.
   - Type \textbf{2} in the Option line on the Collection/Reporting Profiles Menu panel.
   
   Press Enter.
   
   The Reporting Profiles Display panel appears.

2. Type \textbf{S} next to an existing profile and press Enter.

The reporting profile is set as the current reporting profile for collection and the Current Profile field on the Reporting Profiles Display panel is updated with your selection.
View and Update Reporting Profiles

After you define the reporting profiles, you can view the application profile definitions to see what plans, packages, correlation IDs, collection IDs, and users are defined in the reporting profile. You can also add new profile entries, and view, update, and delete existing application profile entries.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   
The Collection/Reporting Profiles Menu panel appears.
2. Type 2 (View Reporting Profiles) and press Enter.
   
The Reporting Profiles Display panel appears.
3. Type V next to a profile and press Enter.
   
The View/Modify Reporting Profile panel appears. In the data area, each line represents one reporting profile ID. The single underscore to the left of each profile name is the selection area for that profile name. Use the Tab key to jump from one ID field to the next. The arrow keys can also be used to move around within the data area.
4. Do one of the following:
   ■ Type S in the Select to ADD Profile entries field to add new profile entries.
   ■ Type V next to an application profile ID to view the application profile definition.
   ■ Type D next to an application profile ID to delete it from the reporting profile.
   ■ Type U next to an application profile ID to add, change, and delete the names that define the application profile.
   
   Press Enter.
   
   Additional panels appear for you to complete reporting profile processing.

Add Profile Entries

You can add profile entries that define plan, package, corrid, collid, and user names that will make up the application profile definition for the selected profile ID.

For example, you can add an entry called BATCH PROCESSOR to the SAMPLE profile that analyzes all plans beginning with RBP.
Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type 2 (View Reporting Profiles) and press Enter.
   The Reporting Profiles Display panel appears.

3. Type V next to a reporting profile and press Enter.
   The View/Modify Reporting Profile panel appears.

4. Type S in the Select to ADD profile entries field and press Enter.
   The Add Application Profile Entry panel appears.

5. Complete the following fields:
   
   **Profile application ID**
   
   Specifies a meaningful and descriptive name of the application to be added to the profile.
   
   **Limits:** 1 to 18 characters. Imbedded blanks can be used.
   
   **Specify the application PACKAGE names in the application profile entry**
   
   Defines the plan, package, corrid, collid, or user name to be added to the application profile entry.
   
   Selection criteria (masks) are allowed.
   
   Up to 40 entries can be specified for inclusion in the application profile entry.
   
   **Note:** CA Detector processes profile entries in EBCDIC collating sequence. If you include a planname, or mask, in different profiles, the performance data for the planname is assigned to the profile that appears first in the collating sequence.

   Press Enter.
   
   The Add Profile Entry Confirmation panel appears.

6. Verify your entries and press Enter.
   
   A message appears stating that your entry has been successfully added.

7. Press F3 (End).
   
   The View/Modify Reporting Profile Display appears.

8. Press F3 (End).
   
   The Reporting Profiles Display appears again. Notice that the profile entry you created has been set as the current profile. You can view the activity collection data using the current profile.
More information:

Masking (see page 18)

View Application Profile Entries

You can view application profile entries.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   
   The Collection/Reporting Profiles Menu panel appears.

2. Type 2 (View Reporting Profiles) and press Enter.

   The Reporting Profiles Display panel appears.

3. Type V next to a profile and press Enter.

   The View/Modify Reporting Profile panel appears.

4. Type V next to an application profile ID and press Enter.

   Each plan, package, user, correlation ID, collection ID, and mask included in the application profile appears on the View Application Profile Entry panel. These fields cannot be updated on this panel.

Update Application Profile Entries

You can add, change, and delete the plan, package, user, correlation IDs, and collection IDs that define the application profile.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.

   The Collection/Reporting Profiles Menu panel appears.

2. Type 2 (View Reporting Profiles) and press Enter.

   The Reporting Profiles Display panel appears.

3. Type V next to a reporting profile and press Enter.

   The View/Modify Reporting Profile panel appears.
4. Type U next to an application profile ID and press Enter.
   The Update Application Profile Entry panel appears. This panel shows all the plan, package, usernames, correlation IDs, or collection IDs that make up the application profile definition for the selected profile ID.

5. Add, change, or delete the plan, package, user name, correlation ID, or collection ID that defines the application profile.
   You can use the tab key to jump from field to field within the data. You can type over an entry to change the name, delete single characters, or the entire name.
   Press Enter.
   The Update Profile Entry Confirmation panel appears.

6. Verify your entry before it is updated in the profile and press Enter.
   The entry is updated.

Delete Profile Entries

You can delete profile entries from your application definitions.

Follow these steps:

1. Type 4 (Maintain collect/report profiles) on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu panel appears.

2. Type 2 (View Reporting Profiles) and press Enter.
   The Reporting Profiles Display panel appears.

3. Type V next to a reporting profile and press Enter.
   The View/Modify Reporting Profile panel appears.

4. Type D next to an application profile ID and press Enter.
   The Delete Profile Confirmation panel appears.

5. Verify the delete information and press Enter.
   The View/Modify Reporting Profile panel appears. A message appears confirming that the profile entry has been deleted.
Delete Reporting Profiles

You can delete reporting profiles when they are no longer needed.

Follow these steps:
1. Type 4 (View/Modify Profiles) in the Option field and press Enter.
   The Collection/Reporting Profiles Menu panel appears.
2. Type 2 (View Reporting Profiles) in the Option line and press Enter.
   The Reporting Profiles Display panel appears.
3. Type D next to a profile name and press Enter.
   The Report Profile Delete Verification panel appears.
4. Verify the information shown and press Enter.
   A message appears confirming the delete request.
Chapter 7: Using SQL Traces

This section contains the following topics:

- **SQL Traces** (see page 177)
- **Start an SQL Trace** (see page 179)
- **View SQL Trace Data** (see page 181)
- **Stop an SQL Trace** (see page 192)
- **SQL Traces Control Using Modify Commands** (see page 193)

**SQL Traces**

You can use the SQL trace component to analyze and diagnose troublesome application activity. SQL trace collection and display facilities provide a granular analysis of application performance. This includes insight into application flow and resource use. Information is available about application threads, commits, aborts, programs, SQL calls, access path scan activity, and locking activity. It also lets you invoke CA Plan Analyzer at selected points within SQL trace panels.

Application developers can use SQL traces to evaluate the impact of planned SQL modifications on performance. This means that you can resolve potential performance problems during the development cycle before being moved into production.

You can also collect the DB2 trace records related to distributed SQL activity. This feature lets you see the flow of DB2 activity related to the execution of remote SQL.

Use the SQL trace request to select the application activity that you want to analyze. The trace request can be qualified or unqualified. To limit trace data collected, you can specify any combination of plan name, auth ID, original auth ID, connection ID, and correlation ID. You can set these fields to exact values or qualified with masks, giving maximum control and flexibility in selecting the application activity to be traced.

Once the trace has been started, you can view the collected trace data while the trace is active or after it has terminated. You can view application performance at various levels of granularity, including thread, commit or abort, application flow, program, and SQL call.


SQL Trace Prerequisites

Before you start a trace, you must allocate and initiate the data set where the trace data is to be collected.

The SQL trace component uses a VSAM ESDS file for the SQL trace data collected while an SQL trace is active. SQL trace data records are logically grouped into physical records prior to writing to the SQL trace data set. SQL trace records are variable, unspanned records with a maximum record length of 32,760 bytes.

The size of an SQL trace data set depends on the amount of application activity that is being traced. If you are not interested in evaluating application lock acquire activity or data manager scan activity, specify LOCK=NO and SCAN=NO when the trace is started. This minimizes the amount of space needed to contain the SQL trace data generated.

When an SQL trace data set becomes full, the related SQL trace data collection is terminated. SQL trace collection can generate a large amount of trace data in a small period of time. Consequently, you might need a trace data set of 50 cylinders or more to capture the SQL trace data necessary to evaluate application performance. The logical records that comprise a physical record are variable as well. The size of the records is dependent upon SQL call type, scan type, and so on. More importantly, volume is based on the application SQL activity that is occurring.

We suggest that you experiment with varying SQL trace data set space allocations. After familiarizing yourself with using the SQL trace component, you will have a feel for the SQL trace data set size required to trace application activity at your site.

If you are tracing application activity that is generating trace data at a rapid rate, make the primary space quantity large so that you will not have to obtain secondary extents during SQL trace collection. Data set extend activity can result in the loss of trace data when extensive trace data is being generated very rapidly.

More information:

Start an SQL Trace (see page 179)
Distributed SQL Traces Collection

The SQL Trace component can optionally collect the DB2 trace records related to distributed SQL activity. With this feature, you can see the flow of DB2 activity related to the execution of remote SQL.

We have integrated CA Detector and CA Subsystem Analyzer output to maximize product functionality when evaluating DB2 subsystem performance and subsystem application workload activity. You can review information provided about application workload on your DB2 objects. Getpage requests, physical I/O activity, and buffer pool hit ratios all help you evaluate which objects are most active and how efficiently they are performing.

CA Detector is also designed to interface seamlessly with CA Plan Analyzer to analyze DB2 access paths for packages, DBRMs, and SQL calls. CA Plan Analyzer works with CA Detector to improve DB2 application performance and provides important supplemental information about optimizing SQL access paths, physical designs, and plan enhancements.

SQL Trace and DB2 Performance Trace

The SQL trace component uses DB2 performance trace during its data collection activities. Various performance trace IFCIDs (Instrumentation Facility Component Identifiers) are activated automatically when a CA Detector start SQL trace request is issued. All performance trace activity is stopped when the SQL trace collection activity is terminated.

To minimize the performance impact of using the SQL trace component, specify a fully qualified plan name and authid when you start the SQL trace request. Also, collect lock information only to view application lock acquisition activity.

Start an SQL Trace

You can use CA Detector to trace application SQL activity and provide a granular analysis of application performance.

When you start an SQL trace, you can perform the following tasks:

- Specify thread related options.
- Specify SQL trace control file and data set related options. These data sets are used to store the data collected during an SQL trace.
- Schedule your trace for a future time.
Start an SQL Trace

- Specify the time limit for data collection.
- Indicate if data locking and scan data information should be collected.

You can start an SQL trace from the CA Detector panels or using MVS modify commands.

Before you start a trace, you must allocate and initiate the data set where the trace data is to be collected.

**Follow these steps:**

1. Type 8 (SSID, Active thread, SQL trace displays) from the CA Detector Main Menu and press Enter.
   
   The Additional Features panel appears.
   
   **Note:** Use the SQLTRACE global command to jump from another product to the SQL trace feature.

2. Type 4 (Initialize SQL trace) in the Option line and press Enter.
   
   The SQL Trace Request panel appears.

3. Specify thread-related options and a descriptive name for the trace and press Enter.
   
   The name helps you identify the trace when you view the collected information.
   
   The trace request can be qualified or unqualified and can optionally collected data related to application lock activity, access path scan information, and DB2 communication activity related to distributed SQL.
   
   **Note:** If you do not specify trace selection qualifications, all SQL trace data is collected until the trace time limit expires, the trace data set becomes full, or the trace is stopped by a terminate SQL trace request. Because the volume of data can grow quickly when all trace data is collected, you might want to limit the SQL trace to a specific plan name, authorization ID, original authorization ID, connection ID, or correlation ID. Masking characters can also be used to limit the selections.
   
   The next SQL Trace Request panel appears.
4. Specify the data set name and allocation parameters for the trace control file and the high-level data set name and allocation parameters for the trace data sets and press Enter.

   - The VSAM trace control file maintains historical and statistical information about the SQL trace request. The trace data set contains the data collected during the trace.
   
   - The trace request must include the name of the data set and amount of time in which to collect the data. This data set must be allocated and initiated before the trace starts and can be reused for subsequent trace data collections. You can also specify whether locking and scan data is collected.

Press Enter.

The next SQL Trace Request panel appears.

5. Specify the date and time for the trace to begin, the trace time limit, and the trace data collection options for locking, scan, and distributed data, and press Enter.

   **Note:** The default values let you start the trace immediately. Pending traces are canceled if the Xmanager task is stopped before the trace begins.

Your selections are processed. If you specified a deferred start time or date, a message informs you that the trace has been submitted to the pending request queue. Otherwise, a message informs you that the trace has started.

Once the trace has been started, you can view the collected data while the trace is active or after it has terminated.

Press Enter.

The trace starts. A message indicates that the request has started. A new trace data set is created using the specified high-level name and allocation parameters. If you specified a deferred start date or time, a message indicates that the trace has been submitted to the pending request queue.

**More information:**

- [SQL Trace Prerequisites](#) (see page 178)
- [START(SQLTRACE) Command—Start an SQL Trace](#) (see page 193)

---

**View SQL Trace Data**

To view SQL trace data, you must first start a trace. After a trace is started, the data can be viewed while the trace remains active or after the trace has terminated. The information collected lets you view application performance at various levels of granularity, including thread, commit/abort, application flow, program, and SQL call.

You can delete, restart, view, or terminate and SQL trace.
Follow these steps:

1. Type **8** (SSID, Active thread, SQL trace displays) from the CA Detector Main Menu and press Enter.
   
The Additional Features panel appears.

   **Note:** Use the SQLTRACE global command to jump from another product to the SQL trace feature.

2. Type **3** (View/term SQL trace) and press Enter.
   
The SQL Trace Status Display panel appears. The header fields determine which traces are included in the trace list at the bottom of the panel and show active, pending, and completed traces.

3. Update the header fields to determine which traces (active, pending, or completed) are included and press Enter.
   
The SQL Trace Status Display panel is refreshed.

4. Specify one of the following commands:

   **D**
   
   Deletes the trace entry in the control file and trace data set. The trace data set is deleted. This command is valid for completed traces only. Active and pending traces must be terminated (T) before they can be deleted.

   **R**
   
   Restarts the trace using the parameters of the trace you select. The SQL Trace Request panel appears, populated with the values from the trace you selected. You can modify these values, if necessary, before submitting the request.

   **S**
   
   Selects a trace to view the results. If you are viewing a trace that ended with an error status, the SQL Trace Status Error Display appears. If you select a trace that ended normally, the SQL Trace Thread Display panel appears showing the data collected during the trace. If the SQL Trace is for a DB2 subsystem that is not on the local LPAR, it must be stopped or have ended before you can use this line command.

   **T**
   
   Terminates (stops) an active trace or cancels a pending trace. If you terminate a pending trace it is removed from the control file and the panel. If you terminate an active trace, the status shows as TERM(USER) when the trace has stopped.

   Press Enter.

   The related SQL trace panel appears for the specified command.
5. Do one of the following:

Type S next to a trace.

- Complete the following steps:
  a. Type S in the View DSN field to view an SQL trace data set that still exists but is no longer associated with a trace control data set and press Enter. The SQL Trace View Request panel appears.
  b. Specify the SQL trace data set name.

Press Enter.

The SQL Trace Thread Display panel appears. A line of output for each unique thread plan name, connection ID, correlation ID, op ID, auth ID combination located in the SQL trace data set appears. Each line of output can contain multiple commits, aborts, or inflight units of work for the thread. It may take a few moments for the request to complete if a large data set (100 cylinders or more) is being formatted. If more data than you need is provided, you can qualify the SQL trace start request with additional thread selection criteria.

If the trace ended with an error, the SQL Trace Status Error Display panel appears instead and provides information about the error that occurred.

**Note:** If the SQL trace is for a DB2 subsystem that is not on the local LPAR, it must be stopped or have ended before you can select it for viewing.

6. Type S next to a thread to view detailed information and press Enter.

The SQL Trace Workunit Display panel appears. This panel shows a line of data for each commit, abort, or inflight unit of work that is active for the selected thread during the trace.

7. Specify one of the following commands to view various aspects of the thread’s activity and press Enter:

**P (Program)**

Evaluates application performance from a program perspective. When you press Enter, the SQL Trace Program Display panel appears.

You can also use the S line command in place of the P to achieve the same results.

**F (Flow)**

Provides unit of work flow information for a thread. When you press Enter, the SQL Trace Workunit Flow Display panel appears.

**C (Scan)**

Formats all scan activity for a unit of work. When you press Enter, the SQL Trace Object Scan Display panel appears.
L (Lock)

Provides application lock acquire activity. When you press Enter, the SQL Trace Locks Acquired Display panel appears.

V (Event)

Shows each DB2 event for a work unit and lets you view event application flow from the lowest level of granularity. When you press Enter, the SQL Trace Event Display panel appears.

D (Dist)

Shows each DB2 distributed flow for a work unit and lets you view application flow from the lowest level of granularity. When you press Enter, the SQL Distributed Flow Display appears.

Several SQL trace displays are provided for viewing program, unit of work flow, scan, locking, event, distributed trace facility, SQL, and detailed sort activity for the SQL call.

After you select an SQL trace thread on the SQL Trace Thread Display, there are several SQL trace displays available for viewing various aspects of the thread's activity. The following describes possible line commands for SQL trace displays. Not all line commands are valid on all displays; valid line commands are listed on each display.

P (Program)

Program information for the thread. When you press Enter, the SQL Trace Program Display appears.

You can also use the S line command in place of the P to achieve the same results.

F (Flow)

Unit of work flow information for the thread. When you press Enter, the SQL Trace Workunit Flow Display appears.

C (Scan)

Scan information for the thread. When you press Enter, the SQL Trace Object Scan Display appears.

L (Lock)

Locking information. When you press Enter, the SQL Trace Locks Acquired Display appears.

V (Event)

Event information. When you press Enter, the SQL Trace Event display appears.

D (Dist)

Distributed Trace Facility. When you press Enter, the SQL Distributed Flow Display appears.
**S (SQL)**

SQL information for the program. When you press Enter, the SQL Trace Program Statement Summary Display or the SQL Trace Call Detail Display appears.

**S (Select)**

Detailed sort activity for the SQL call. Applicable only from the SQL Trace Program Sort Summary Display and only for those calls that incurred multiple sorts.

When you press Enter, the SQL Trace Sort Detail Display appears.

**E (Explain)**

Access path analysis for the program.

This option leads to the PPA Quick Explain display in CA Plan Analyzer. It allows access path analysis for all SQL calls contained within the selected program.

See the *CA Plan Analyzer for DB2 for z/OS User Guide* for information about Quick Explain.

**R (Sort)**

Sort activity for the thread. When you press Enter, the SQL Trace Sort Detail display appears.

---

**View SQL Trace Program Data**

You can evaluate application performance from a program perspective and identify which programs have been executed and which programs account for the majority of application resource use after a trace is started.

**Note:** This procedure assumes that trace data has already been collected.

**Follow these steps:**

1. Type **8 (SSID, Active thread, SQL trace displays)** from the CA Detector Main Menu and press Enter.

   The Additional Features panel appears.

2. Type **3 (View/term SQL trace)** and press Enter.

   The SQL Trace Status Display panel appears.

3. Update the header fields to determine which traces (active, pending, or completed) are included and press Enter.

   The SQL Trace Status Display panel is refreshed.
4. Do **one** of the following:
   - Type **S** next to a trace.
   - Complete the following steps:
     a. Type **S** in the View DSN field to view an SQL trace data set that still exists but is no longer associated with a trace control data set and press Enter. The SQL Trace View Request panel appears.
     b. Specify the SQL trace data set name. Press Enter. The SQL Trace Thread Display panel appears.
5. Type **S** next to a thread to view detailed information and press Enter. The SQL Trace Workunit Display panel appears.
6. Type **P** next to a thread to view program information and press Enter. The SQL Trace Program Display panel appears. This panel lets you identify which programs have been executed and which programs account for the majority of application resource use. A line appears for each program that has executed an SQL call from within the thread unit of work being viewed. From this display you can evaluate application performance from a program perspective. You can easily identify which programs have been executed and which programs account for the majority of application resource use.
7. Specify one of the following commands:
   - **S** (SQL)
     Provides application resource use by SQL statement within a program. When you press Enter, the SQL Trace Program Statement Summary panel appears.
     The SQL Trace Program Statement Summary display formats SQL call activity originating from a selected program. A line of output is formatted for each unique SQL call type and statement number issued from the program. When viewing this display you can evaluate SQL activity for a selected program.
   - **E** (Explain)
     Access path analysis for the program.
     This option leads to the PPA Quick Explain display in CA Plan Analyzer. It allows access path analysis for all SQL calls contained within the selected program.
     See the *CA Plan Analyzer for DB2 for z/OS User Guide* for information about Quick Explain.
View SQL Trace Workunit Flow Data

To view SQL trace data, you must first start a trace. After a trace is started, the data can be viewed while the trace remains active or after the trace has terminated. The information collected lets you view application performance at various levels of granularity, including thread, commit/abort, application flow, program, and SQL call.

You can delete, restart, view, or terminate and SQL trace.

Follow these steps:

1. Type 8 (SSID, Active thread, SQL trace displays) from the CA Detector Main Menu and press Enter.

   The Additional Features panel appears.

   **Note:** Use the SQLTRACE global command to jump from another product to the SQL trace feature.

2. Type 3 (View/term SQL trace) and press Enter.

   The SQL Trace Status Display panel appears. The header fields determine which traces are included in the trace list at the bottom of the panel and show active, pending, and completed traces.

3. Update the header fields to determine which traces (active, pending, or completed) are included and press Enter.

   The SQL Trace Status Display panel is refreshed.
4. Do one of the following:
   ■ Type S next to a trace.
   ■ Complete the following steps:
     a. Type S in the View DSN field to view an SQL trace data set that still exists but is no longer associated with a trace control data set and press Enter. The SQL Trace View Request panel appears.
     b. Specify the SQL trace data set name.

Press Enter.

The SQL Trace Thread Display panel appears. A line of output for each unique thread plan name, connection ID, correlation ID, op ID, auth ID combination located in the SQL trace data set appears. Each line of output can contain multiple commits, aborts, or inflight units of work for the thread. It may take a few moments for the request to complete if a large data set (100 cylinders or more) is being formatted. If more data than you need is provided, you can qualify the SQL trace start request with additional thread selection criteria.

If the trace ended with an error, the SQL Trace Status Error Display panel appears instead and provides information about the error that occurred.

Note: If the SQL trace is for a DB2 subsystem that is not on the local LPAR, it must be stopped or have ended before you can select it for viewing.

5. Type S next to a thread to view detailed information and press Enter.

The SQL Trace Workunit Display panel appears. This panel shows a line of data for each commit, abort, or inflight unit of work that is active for the selected thread during the trace.

6. Type F (flow) and press Enter.

The SQL Trace Workunit Flow Display panel appears. This panel provides application flow information and helps you identify what area of application flow accounts for the majority of resource use. An output line indicates the program, SQL statement, the number of times the SQL statement was executed, and various resource use related fields. For example, if an application is executing in PGMA and executes an OPEN CURSOR once, a line of output displays describing the OPEN CURSOR statement. If PGMA then executes 1000 FETCH statements, a single line of output displays for this activity. If the application then transfers control to PGMB and executes a SELECT statement, a single line of output is built for this activity.
View SQL Trace Sort Detail

You can view DB2 sort statistics for selected SQL statements called by the program or DBRM.

Follow these steps:
1. Do one of the following:
   - Enter R (Sort) next to a workunit on the SQL Trace Workunit Flow display
   - Enter S (Select) next to a program on the SQL Trace Program Sort Summary display.

   Press Enter.
   The SQL Trace Sort Detail panel appears.
2. Type S next to a sequence number or total row and press Enter.
   The SQL Trace Sort Detail panel appears. The data is reformatted into a single page display. All the information shown in the other (multicolumn) view is present. To view the other sequences that are included in the total, press F7 (Up) and F8 (Down) to scroll the panel.

View SQL Trace Program Sort Summary Data

You can view DB2 sort statistics for the selected program or DBRM.

To view SQL trace program sort summary data, type R (sort) next to a program on the SQL Trace Program Display panel and press Enter.

The SQL Trace Program Sort Summary panel appears. Use F1 to view descriptions of the column data.

View SQL Trace Object Scan Data

You can view DB2 objects accessed by the application on the SQL Trace Object Scan Display panel. It shows the types of scans that were processed by DB2 in satisfying the SQL requests issued by an application. It also provides various scan related data values so you can understand application access path scan activity and the volume of data being referenced. The output is formatted by objects accessed and scan type.

To view SQL trace object scan data, type C next to an object on the following panels:
   - SQL Trace Workunit Display
   - SQL Trace Program Display
   - SQL Trace Program Statement Summary display
Press Enter.

The SQL Trace Object Scan Display panel appears.

**Note:** The output displayed depends on where the SQL Trace Object Scan Display request originated from. For example, if it is requested from the SQL Trace Workunit Display, then it formats all scan activity it can locate for the unit of work being viewed. If it is invoked from the SQL Trace Program Display, then it displays all scan activity it can locate that originated as a result of SQL activity from the program selected.

**View Application Lock Acquire Activity**

You can view application lock acquire activity. You can use it to gain an understanding of the lock types and lock states used by DB2 in an effort to provide concurrent data access and ensure data integrity. Output also includes the lock resource for all locks acquired. The display is formatted by lock type, state, and resource. A count field indicates how many times the lock was acquired.

To view application lock acquire activity, use the L line command on the following panels:

- SQL Trace Workunit Display
- SQL Trace Program Display
- SQL Trace Program Statement Summary
- SQL Trace Workunit Flow Display.

The SQL Trace Locks Acquired Display panel appears.

**Note:** The output displayed depends on where the SQL Trace Locks Acquired Display request originated from. For example, if it is requested from the SQL Trace Workunit Display, then it formats all lock acquire activity it can locate for the unit of work being viewed. If it is invoked from the SQL Program Summary Display, then it displays all the lock acquire activity it can locate that originated as a result of SQL activity from the program selected.
View Application Resource Use by SQL Statement Type

You can determine application resource use by SQL statement type within a program.

To view application resource use by SQL statement type, use the $ line command on the SQL Trace Program Display panel.

The SQL Trace Program Statement Summary panel appears. SQL call activity originating from a selected program is formatted. A line of output is formatted for each unique SQL call type and statement number issued from the program. When viewing this display you can evaluate SQL activity for a selected program.

View SQL Activity Calls

The SQL Trace Call Detail display appears when you select an SQL call using the $ line command on the SQL Trace Program Statement Summary display.

The SQL Trace Call Detail display formats SQL activity at the SQL call level. A single line of output is produced for each SQL call invocation. This display provides a detailed, granular analysis of SQL call activity originating from a selected program.

The SQL Trace Call Text display appears when you select a PREPARE SQL call sequence number (SEQNO) on the SQL Trace Call Detail display. The SQL Trace Call Text display formats the actual SQL text for a selected PREPARE SQL Call.

The SQL Trace Call Text display shows the actual SQL text being executed at the time the SQL trace record was captured. In the example above, the SQL statement INSERT displays.

View Application Event Flow

The SQL Trace Event Display appears when you use the V line command on the SQL Trace Workunit display, SQL Trace Program display, SQL Trace Workunit Flow display, and the SQL Trace Program Statement Summary display. This display shows each DB2 event for a workunit. It lets you view event application flow from the absolute lowest level of granularity.

The display columns GETPAGE through ROW_DLRI only contain values for End Scan events. If you want to view this type of data for SQL calls, use the other Trace Detail displays.
Stop an SQL Trace

View DB2 Distributed Flow for a Workunit

The SQL Distributed Flow Display appears when you use the D line command on the SQL Trace Workunit display, SQL Trace Program display, SQL Trace Workunit Flow display, and the SQL Trace Program Statement Summary display. This display shows each DB2 distributed flow for a workunit. It lets you view distributed application flow from the absolute lowest level of granularity.

The display columns GETPAGE through ROW_DLRI only contain values for End Scan events. If you want to view this type of data for SQL calls, use the Trace Detail displays.

Stop an SQL Trace

Once started, you can stop an SQL trace automatically by specifying an expiration time limit on the SQL Trace request panel or setting resource limits so that, when the trace data set in use by the trace request becomes full, the trace automatically stops.

You can also stop an SQL trace manually using a stop SQL trace request and STOP modify commands.

Follow these steps:

1. Type 8 (SSID, Active thread, SQL trace displays) from the CA Detector) Main Menu and press Enter.

   The Additional Features panel appears.

   **Note:** Use the SQLTRACE global command to jump from another product to the SQL trace feature.

2. Type 3 (View/Term SQL Trace) to stop an SQL trace and press Enter.

   The SQL Trace Status Display panel appears. The header fields determine which traces are included in the trace list at the bottom of the panel and the status of each.

3. Update the header fields to determine which traces (active, pending, or completed) are included and press Enter.

   The SQL Trace Status Display panel is refreshed.

4. Type T (terminate) next to the trace you want to stop and press Enter.

   The command is processed. It may take a few moments for the trace request to complete termination processing.

   **Note:** Only the user that started the SQL trace can terminate it from the SQL Trace Status Display panel or using modify commands.

5. Press Enter to refresh the panel after successful termination of the trace.

   The status of the trace changes to TERM(USER).
More information:

STOP(SQLTRACE) Command—Stop an SQL Trace (see page 198)

SQL Traces Control Using Modify Commands

You can start and stop SQL traces with a console command, an automated operation tool, or any other technique in which a modify command can be issued. You can specify all start and stop SQL trace options available on the SQL Trace panels, as well as additional options, using modify commands with keywords.

You must specify the appropriate keywords when issuing the command to the Xmanager started task on the appropriate system. In most cases, the operand can include masking characters to limit the selection. Related error messages are issued in the form of WTO (write-to-operator) messages. Check the JES joblog or the MVS syslog for any WTO messages that indicate a successful or failed result.

However, because of the number of parameters that are required to specify all the SQL trace options, you will usually issue the commands with the RUN modify facility.

START(SQLTRACE) Command—Start an SQL Trace

The START(SQLTRACE) modify command starts an SQL trace using thread, control file, data set related, deferred start, time limit, and data collection keywords. Each keyword corresponds to an option on the SQL Trace Request panels.

You can start an SQL trace using an MVS modify command with the following syntax:

```
F PTXMAN,START(SQLTRACE),DB2(ssid),[keyword(value),keyword(value),...]
```

**PTXMAN**

(Required) Specifies the name of the Xmanager started task. PTXMAN is usually specified.

**START(SQLTRACE)**

(Required) Starts the SQL trace. This is the first keyword and related operand. It must be provided as the first keyword, exactly as shown. The keyword starts in position 1.

If this command is accepted, SQL trace processing begins. The status of the trace request can be monitored using the SQL Trace Status panel. If the trace fails and the status panel contains no record of the trace, it is possible that the allocation or update of the trace control file failed. In this case, the modify command parameters associated with the control file should be reviewed to ensure they are appropriate.
DB2(ssid)
(Required) Specifies the DB2 subsystem on which processing occurs.
The DB2 subsystem must reside on the same z/OS system as the one from which
the modify command is issued.

keyword(value)
Specifies the thread, control, data set, deferred start, time limit, and data collection
options you want to include in your trace.

More information:
Start an SQL Trace (see page 179)

Thread Related Keywords

Use the following optional keywords when you start an SQL trace to specify thread
related options and provide a descriptive name of the trace you are initiating:

DESC(text)
Specifies an optional description of the trace. To use embedded spaces in the
description, enclose the entry in single quotes.
Limits: 1 to 32 characters

PLAN(planname)
Specifies the plan name for which the SQL trace is requested.

AUTH(authid)
Specifies the authorization ID for which the SQL trace is being requested.

OAUTH(original authid)
Specifies the original authorization ID for which the SQL trace is being requested.

CONN(connid)
Specifies the connection identifier for which the SQL trace is being requested.

CORR(corrid)
Specifies the correlation identifier for which the SQL trace is being requested.
Control File Related Keywords

Use the following SQL trace control file related keywords to specify the data set name and allocation parameters for the trace control file. The VSAM trace control file is used to maintain historical and statistical information about your SQL trace requests.

CDSN(*dsn*)

(Required) Specifies the fully qualified name of the data set name of the control file you want used to use to store a history of your trace requests.

CVOL(*volume*)

Specifies the volume where the trace control file should be created. This keyword is required if the trace control file does not already exist. If specified, you must also specify CUNIT, CPRI, and CSEC. This keyword is not required if you use the SMS keywords instead.

CUNIT(*alloc unit*)

Specifies the data set allocation in tracks (TRKS) or cylinders (CYLS). This operand is required if the data set does not already exist.

CPRI(*primary space*)

Specifies the primary space quantity to be allocated in the allocation units specified in the CUNIT keyword. This field is required if the data set does not already exist.

CSEC(*secondary space*)

Specifies the secondary space quantity to be allocated in the allocation units specified by the CUNIT keyword. This field is required if the data set does not already exist.

CSMSS(*sms storage class*)

Specifies the SMS storage class for the control data set.

CSMSD(*sms data class*)

Specifies the SMS data class for the control data set.
Data Set Related Keywords

Use the following SQL trace data set related keywords to specify the high level data set name and allocation parameters for the trace data set. The trace data set contains the data collected during the trace. A new trace data set is created, using the specified high level of the name and allocation parameters, each time you start a new SQL trace.

**TDSN**(dsn)

(Required) Specifies the high level qualifier of the data set name you want used to store data collected during the trace. The remaining data set name qualifiers are generated automatically. A new name is generated each time you start an SQL trace.

**TVOL**(volume)

Specifies the volume where the trace data set should be created. This field is required if the data set does not already exist. It is not required if the SMS keywords are used instead.

**TUNIT**(alloc unit)

Specifies the data set allocation in tracks (TRKS), cylinders (CYLS), or TPRI (primary space).

**TPRI**(primary space)

Specifies the data set primary space quantity to be allocated using the allocation unit identified in the TUNIT keyword. This keyword is required if the data set does not already exist.

**TSEC**(secondary space)

Specifies the data set secondary space quantity to be allocated using the allocation units specified in the TUNIT keyword.

**Note:** Avoid the use of secondary space allocation if possible. Due to the intensity of trace data collection activities, the prolonged wait activity that can occur during data set extend processing may result in the loss of trace data.

**TSMSM**(sms mgmt class)

Specifies the trace data set SMS management class.

**TSMSS**(sms storage class)

Specifies the trace data set SMS storage class.

**TSMSD**(sms data class)

Specifies the trace data set SMS data class.
Deferred Start, Time Limit, and Data Collection Keywords

Use the following SQL trace request keywords to schedule your trace for a future time, specify the time limit for data collection, and indicate if data locking, scan data information, or both should be collected:

**STIME(start time)**

(Optional) Specifies the time you want the trace to begin in hours and minutes. The start time and start date, if specified, must be greater than or equal to the current time and date.

The default time is 0000 if this keyword is not issued.

**SDATE(start date)**

(Optional) Specifies the date on which you want the SQL trace to begin by month, day, and year (MMDDYYYY). The start time and date, if specified, must be greater than or equal to the current time and date.

The default is the current date.

**TLIM(trace time)**

(Required) Specifies how long in hours and minutes the trace remains active. For example, to request a trace for thirty minutes, enter 0030. To request a trace time of five hours and twenty-three minutes, enter 0523.

**LOCK(y|n)**

(Optional) Collects locking information if set to Y. If omitted or set to N, locking information is not collected.

**SCAN(y|n)**

(Optional) Collects scan information if set to Y. If omitted or set to N, scan information is not collected.

**USER(userid)**

(Required) Identifies the TSO user ID or job name on whose behalf the SQL trace is being started. All data set authorizations are based on this ID; therefore, this keyword is required for proper control file updating and externalization of SQL trace data.

**ID(trace id)**

(Optional) Specifies an eight character identifier associated with the trace. This keyword is optional, but it must be specified for the trace to be stopped using a modify command. The trace ID must be unique for all currently active and pending traces. The SQL trace does not start if the specified value is not unique.

**TPLAN(planname)**

(Required) Specifies the first CA Detector plan name in the list of plan names specified during product installation. This is not the plan name for which trace data is being collected (see PLAN keyword).
STOP(SQLTRACE) Command—Stop an SQL Trace

You can stop SQL traces started with the START(SQLTRACE) modify command if the ID (trace id) keyword was specified when the trace was started. If the trace was started using the ISPF CA Detector panels, or if the ID (trace id) keyword was omitted, then the trace cannot be stopped using this method.

You can stop an SQL trace using an MVS modify command with the following syntax:

F PTXMAN,STOP(SQLTRACE),ID(traceid)

PTXMAN

(Required) Specifies the name of the Xmanager started task. PTXMAN is usually specified.

STOP(SQLTRACE)

Specifies the SQL trace stop keyword. This is the first keyword and related operand. It must be provided as the first keyword, exactly as shown. The keyword starts in position 1.

ID(trace id)

Specifies the identifier of the active or pending trace to be stopped. The identifier was optionally specified when the trace was started. If an identifier was not specified, you cannot stop the trace with a modify command.

Limit: 1 to 8 characters

More information:

Stop an SQL Trace (see page 192)

Execute Modify Commands from the Xmanager Parmlib Member

You can execute the Xmanager modify commands to start and stop an SQL trace by placing the commands in an Xmanager parmlib member and then issuing the RUN modify command. This method is useful when you need to automate the use of CA Detector modify commands. It is also useful when the parameters for the modify command exceed the maximum allowable length for a MVS console command, JCL, or other technique.

To create the Xmanager parmlib member, use ISPF Edit or another appropriate method.

Note: The member name should not begin with D because the members that are created by CA Detector in this PDS begin with D.
Edit the member so that it contains the appropriate modify commands to be issued to Xmanager. The commands must be in the format previously described, such as START(SQLTRACE), and so on. Do not use the full MVS command structure. In other words, assuming that PTXMAN is the name of the Xmanager started task, the F PTXMAN portion of the command is not required. For example, you can enter the following into a member:

-STOP(SQLTRACE), ID(PDTSAMP)

This member, when executed using a modify RUN command, causes the trace PDTSAMP to be stopped.

**Note:** All commands must be preceded with a hyphen (-) character followed immediately by the command. The command can be split across lines, if necessary. The hyphen indicates the start of the next command.

To issue the modify command you have entered in the Xmanager parmlib member, use the RUN command, as shown in the following example. This example assumes the parmlib member is MODIFY01, and that PTXMAN is the name of the Xmanager started task.

F PTXMAN, RUN(MODIFY01)

The example command causes Xmanager to find each of the commands in MODIFY01 (the name of the Xmanager parmlib member that contains the modify commands you want to issue to Xmanager) that start with a hyphen and execute them as if they were issued by a modify command. For example, if MODIFY01 contained the -STOP(SQLTRACE) command as previously illustrated, the trace would be stopped.
Chapter 8: Analyzing Current Subsystem Activity

This section contains the following topics:

- **DB2 Threads** (see page 201)
- **View a DB2 Subsystem List** (see page 202)
- **View Subsystem Threads** (see page 202)
- **View Current Thread Activity** (see page 203)

## DB2 Threads

You can view current DB2 thread activity to evaluate current DB2 users and related resource use information easily. You can view a list of current threads located on a selected DB2 subsystem and observe and evaluate current thread activity. You can also easily switch between multiple DB2 subsystems by changing the DB2 subsystem ID. Thread identification information as well as selected performance related values are available so you can evaluate current thread activity and resource use.

**Note:** Class 2 and Class 3 times are only available if their respective accounting classes are active on the subsystem being viewed.

You can select and evaluate a thread entry in detail for the following information:

- Thread timing information
- Buffer manager information
- SQL statistics
- SQL call information
- Locking information
- Distributed statistics

**Note:** If you have installed the Thread Termination\Dynamic DSNZPARM Value Pack component, you can also terminate a thread. You can also jump to CA Plan Analyzer to explain SQL call text if you have CA Plan Analyzer installed.

**More information:**

- **Using SQL Traces** (see page 177)
View a DB2 Subsystem List

You can view a list of all DB2 subsystems defined on your z/OS system. This list provides the subsystem identifiers, recognition characters, status, version, data sharing group information, date and time started, number of active connections and threads.

Follow these steps:

1. Type 8 (SSID, Active thread, SQL trace displays) in the Option line on the CA Detector Main Menu, and press Enter.
   The Additional Features menu appears.
2. Type 1 (View Subsystem List) and press Enter.
   The Thread Terminator DB2 Subsystem Display panel appears. This panel lists all DB2 subsystems defined on the z/OS system you are currently logged on to.
   Note: You can use the LISTDB2 global command to jump directly to this panel from other products.

View Subsystem Threads

You can view current DB2 subsystem users (thread activity) for a selected DB2 subsystem.

Note: Use the LISTTHD global command to jump directly to the active thread display panel.

Follow these steps:

1. Type 8 (SSID, Active thread, SQL trace displays) in the Option line on the CA Detector Main Menu and press Enter.
   The Additional Features menu appears.
2. Type 1 (View Subsystem List) and press Enter.
   The Thread Terminator DB2 Subsystem Display panel appears.
3. Type $ next to a subsystem to select it and press Enter.
   The Thread Terminator Active Threads Display panel appears. The selected subsystem is then used for subsequent active threads displays.
   Note: To evaluate threads on a different subsystem, enter the subsystem ID in the SSID field and press Enter.
View Current Thread Activity

You can evaluate an active threads activity in a granular fashion.

**Note:** If you have installed the Thread Termination\Dynamic DSNZPARM Value Pack component, you can also terminate a thread. You can also jump to CA Plan Analyzer to explain SQL call text if you have CA Plan Analyzer installed.

**Follow these steps:**

1. Type **8** (SSID, Active thread, SQL trace displays) from the CA Detector Main Menu and press Enter.
   
The Additional Features menu appears.

2. Type **2** (View current users) to view current thread activity for a subsystem and press Enter.
   
The Thread Terminator Active Threads Display panel appears.

   **Note:** Use the LISTTHD global command to jump directly to this panel from other products.

3. Select a thread from the list and press Enter.
   
The Thread Terminator Thread Time Display panel appears.

4. Change the Option field value to display additional information for the thread as follows:

   **B (Buffer)**
   
   Identifies DB2 buffer manager statistics for a currently active thread from a buffer pool perspective. Analysis of thread buffer manager statistics provides insight into data buffer access and I/O requests performed.

   **Note:** To see this information in report format, select an entry from the list and press Enter.

   **S (SQL)**
   
   Identifies active thread SQL call activity information to identify the type and count of SQL calls that have been issued by an active thread.

   **C (SQLCall)**
   
   Identifies current SQL call information for an active thread. SQL call text appears for threads being viewed that are currently executing in DB2. SQL call text is unavailable if the thread being viewed is not executing within DB2. In such a case only the program name, SQL statement number, and call type are available for the last SQL call that was executed.

   **Note:** When SQL call text is available, you can invoke CA Plan Analyzer (if installed) to perform access path analysis for the currently active SQL call.
L (Lock)

Specifies evaluation information for locks currently held by a thread. This include the lock type, lock state, lock resource, and the number of locks held for all locks owned by a thread. Output is grouped by lock type, state, and resource.

I (Parallelism)

Identifies accounting information that is available for analysis of I/O parallelism. Named I/O parallelism provides multiple, concurrent, I/O streams to service I/O requirements generated for SQL or utility access.

P=Program

Specifies package level accounting information. This information lets you view a subset of accounting information at the DBRM or package level.

R=Drain

Specifies information about thread claim and drain activity to serialize and manage access to DB2 pagesets.

Z=SQLtrace

Specifies thread related options and a descriptive name for traces.

D=Dist

Specifies distributed relationship information for the thread if available.

T=Time

Specifies detailed, time related information about a currently active thread. Included information helps you analyze thread elapsed time, INDB2 time, and CPU use.

Press Enter.

Depending on your selection, a new panel appears and displays the requested thread activity information for analysis.
Chapter 9: Sample Scenarios

This chapter contains examples of how to create a reporting profile and how to use CA Detector with other products. The samples demonstrate how CA Detector can answer the questions typically asked by DB2 administrators or performance analysts.

This section contains the following topics:

How to View Application Activity (see page 205)
Perform Access Path Analysis with CA Plan Analyzer (see page 206)
Use SQL Trace (see page 209)

How to View Application Activity

Reporting profiles let you view application use and resource consumption by application. The plan or package data is grouped for you according to the reporting profile you define. To view application activity, you must first define a reporting profile.

Follow these steps:

1. Create a reporting profile for your application and add entries to it.
   
   **Note:** If you have not already started collection, do so now.

2. View collection data for your applications as follows:

   a. Type 1 in the Option line on the CA Detector Main Menu and press Enter.
   
      The Planname Summary Display panel appears.

   b. Type F (profile) in the View By field and press Enter.
   
      The Application Profile Display panel appears. The package or plan data is grouped using the current profile's application profile definitions. This lets you view resource consumption according to the applications at your site. The NO PROFILE application entry represents all applications that were not otherwise assigned to a profile category.
CA Detector interfaces with CA Plan Analyzer to perform powerful DB2 access path analysis on DB2 packages and DBRMs and SQL calls. You can invoke CA Plan Analyzer to evaluate access path information for the package or DBRM you are currently viewing with CA Detector. This built-in link to CA Plan Analyzer is more powerful than using Interproduct Interface commands because it takes you directly to the Quick Explain panel for the currently selected package or DBRM.

**Note:** You must have a license for CA Plan Analyzer to perform this analysis. For more information about using CA Plan Analyzer to perform access path analysis, see the CA Plan Analyzer documentation.

**Follow these steps:**

1. Specify the DB2 subsystem ID and option 1 on the CA Detector Main Menu and press Enter.
   
   The DETECTOR Planname Summary Display panel appears.

2. Select the plan name to view and press Enter.
   
   The DETECTOR Plan Program Display panel appears.

3. Select the package or DBRM to be explained and press Enter.
   
   The PPA Quick Explain panel appears.

4. Press Enter to generate the PPA Quick Explain Batch Processor statements.
   
   CA Plan Analyzer generates and shows the Enhanced Explain processor input statements on the PPA Batch Processor Card Display panel.

5. Edit the statements if needed to specify more information and press Enter.
   
   The EXPLAIN is executed using the Batch Processor. This lets you see a variety of enhanced explain reports for the selected DBRM.
Example: Sample Cost Report

A sample Cost report follows:

<table>
<thead>
<tr>
<th>PLAN/COLLECTION</th>
<th>DBRM/PKG</th>
<th>PACKAGE VERSION</th>
<th>STMT#</th>
<th>TYPE (DESC ORDER)</th>
<th>RELATIVE COST (E+06)</th>
<th>NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1212,SEL</td>
<td>+0.4184 E+06</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1104,SEL</td>
<td>+0.2487 E+06</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1206,SEL</td>
<td>+0.2421 E+06</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1174,SEL</td>
<td>+0.2421 E+06</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1200,SEL</td>
<td>+0.2421 E+06</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1187,SEL</td>
<td>+0.2393 E+06</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1167,SEL</td>
<td>+0.2386 E+06</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPAMB41A .........,PPA$BPDR, .................., 1168,SEL</td>
<td>+0.2384 E+06</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example: Sample Access Path Analysis

In this example, the access path analysis for statement number 1160 in this DBRM is shown:

```
------------  AUDIT Message File  --------------

USER ID: USER01
SOURCE DB2 SSID: D91A DATE/TIME: 06/01/02, 09:33:16:02
PLAN: PPAMB41A DB2 SSID : D91A
DBRM: PPA$BPDR SQL CODE : +0
STMT: 1160 MESSAGE :

SQL STATEMENT TEXT:
DECLARE CSR_STATV1_E
CURSOR
FOR
SELECT A.NAME , A.PDSNAME , A.PLNAME , A.PLCREATOR
FROM SYSIBM.SYSDBRM A
WHERE A.PLNAME = :R_PLAN_NAME
AND A.PLCREATOR = :R_CREATOR
ORDER BY A.PLNAME , A.PDSNAME , A.NAME

ACCESS PATH ANALYSIS:
THIS ACCESS PATH CONSISTS OF 1 QUERY BLOCK(S).
QUERY BLOCK 1 HAS 2 STEP(S).

+++ QUERY BLOCK:  1
|
| THIS QUERY BLOCK CONSISTS OF 2 STEP(S).
|
|
| +++ PLAN STEP:  1
| | THE TABLE SYSIBM.SYSDBRM IS THE 1ST TABLE ACCESSED
| | TO SATISFY THIS QUERY. DATA FROM THIS TABLE WILL BE ACCESSED VIA
| | A SEQUENTIAL SCAN (TABLESPACE SCAN) OF THE TABLES DATA PAGES.
| | THIS TABLE'S TABLE SPACE WILL BE LOCKED IN INTENT SHARE MODE.
| | THIS PROCESS MAY READ DATA IN THE TABLE OR TABLE SPACE, BUT
| | NOT CHANGE IT. OTHER APPLICATION PROCESSES MAY BOTH READ
| | AND CHANGE THE DATA.
| | DATA PAGES WILL BE READ IN ADVANCE BY PURE SEQUENTIAL PREFETCH.
| | +++ PLAN STEP:  1 END.
|
| +++ PLAN STEP:  2
| | THIS PLAN STEP WILL PERFORM EXTRA REQUIRED SORTING. NO NEW
| | TABLE WILL BE ACCESSED.
| | THE COMPOSITE TABLE WILL BE SORTED FOR AN ORDER BY CLAUSE.
| | +++ PLAN STEP:  2 END.
|
+++ QUERY BLOCK:  1 END.
```

Perform Access Path Analysis with CA Plan Analyzer
Use SQL Trace

SQL trace lets you monitor SQL execution in CA Detector and in our other CA Database Management Solutions for DB2 for z/OS. This provides information about application flow and resource usage. For example, when used with CA SQL-Ease, you can monitor and tune DB2 application performance during the development cycle.

The following scenario describes the process of accessing the CA Detector SQL trace component from CA SQL-Ease to trace a poorly performing SQL statement.

Initiate a Trace on Your Current SQL Statement

You can initiate a SQL trace on your current SQL statement.

Follow these steps:

1. Enter the SQLTRACE global command in the command line of CA SQL-Ease (or from any CA Database Management Solutions for DB2 for z/OS).
   The SQL Trace Main Menu panel appears.
2. Type 2 in the Option field and press Enter.
   The SQL Trace Request panel appears.
   From this panel, you can initiate a trace of application SQL activity for any plan name and auth ID combination. The trace can include:
   ■ What data you want to collect.
   ■ Where you want to allocate the collected data.
   ■ What trace time limit, if any, you want imposed.
   ■ Whether to collect scan and locking information.
   Note: Qualifying your trace request minimizes associated data collection overhead.
3. Enter data in the fields as needed. Press Enter to accept the data and move to the next panel, on which you can specify the data sets used to store the data collected during an SQL trace.
   The third panel appears. This panel lets you schedule your trace for a future time, specify the time limit for data collection, and indicate whether data locking and scan data information should be collected.
4. Fill in all the fields on the SQL Trace Request panels, and press Enter to process your selections.
   If you specified a Deferred Start Time and Date, a message informs you that the trace has been submitted to the pending trace request queue. Otherwise, a message informs you that your start request is being processed.
Execute Test SQL

After collection starts, you can go to your CA SQL-Ease session to execute your test SQL on the Dynamic SQL Execution panel.

To execute the SQL, type S in the Option field, and press Enter.

View SQL Trace Data

You can view the SQL trace data using the SQLTRACE command.

Follow these steps:

1. Type 1 (View/Stop an SQL Trace) from the SQL Trace Main Menu panel and press Enter.
   The SQL Trace Status Display panel appears. This panel lets you view, restart, and purge active, pending, and completed SQL trace requests.
2. Type S (Select/View) in the line command field next to the UserID column, and press Enter.
   The SQL Trace Thread Display panel appears. This panel presents data for each thread that was active during the trace.
3. Type S next to a thread and press Enter.
   The SQL Trace Workunit Display panel appears. This panel shows data for all commit, abort, and inflight work units active for that thread during the trace.
4. Use the following line commands to access the various SQL Trace panels. Valid values are:
   - P (program)
   - F (flow)
   - C (scan)
   - L (lock)
   - V (event)
   - D (dist)
Example: View the Work Unit Display at the Program Level

Type P (program) on the SQL Trace WorkUnit Display panel to view the workunit display at the program level. The SQL Trace Program Display panel appears, showing data for each program in the specified workunit. From this panel, you can enter a line command to view information about DB2 sort activity.

If you enter R next to the program, a modified SQL Trace Program Display appears. To view the next level of granularity, enter S next to a program and press Enter. The SQL Trace Sort Detail panel appears.

View Event Application Flow

You can view event application flow from the absolute lowest level of granularity.

Follow these steps:

1. Select option 1 (View/Stop an SQL Trace) from the SQL Trace Main Menu panel and press Enter.
   The SQL Trace Status Display panel appears.
2. Type S (Select/View) in the line command field next to the UserID column, and press Enter.
   The SQL Trace Thread Display panel appears.
3. Type S next to a plan name to select a thread and press Enter.
   The SQL Trace Workunit Display panel appears.
4. Type V in the line command field next to the StartTime column.
   The SQL Trace Event Display panel appears.

View Distributed Application Flow

You can view distributed application flow from the absolute lowest level of granularity.

Follow these steps:

1. Select option 1 (View/Stop an SQL Trace) from the SQL Trace Main Menu panel and press Enter.
   The SQL Trace Status Display panel appears.
2. Type S (Select/View) in the line command field next to the UserID column, and press Enter.
   The SQL Trace Thread Display panel appears.
3. Type S next to a plan name to select a thread and press Enter.
   The SQL Trace Workunit Display panel appears.
4. Type D in the line command field next to the Start Time column.
   The SQL Distributed Flow Display panel appears.

**Terminate the SQL Trace**

An SQL trace automatically terminates when the trace data set becomes full or the time limit specified on the SQL Trace Request Display expires. You can terminate an SQL trace at any time.

**Follow these steps:**

1. Select option 1 from the SQL Trace Main Menu and press Enter.
   The SQL Trace Status Display panel appears.
2. Type T in the line command field to the left of the UserID field and press Enter.
   The SQL trace is terminated.
Appendix A: Quick Reference

This appendix presents introductory information about most of the CA Detector facilities to help you learn how to use the product. Review the individual component chapters for detailed information. You can also access online help at any time for field descriptions, options, and suggestions by pressing PF1 (Help).

This section contains the following topics:

- Access the Tutorial (see page 213)
- Access CA Detector (see page 213)
- Initialize a Datastore (see page 214)
- Create Collection Profiles (see page 217)
- Start Collection (see page 219)
- View Standard Activity Collection Data (see page 220)
- View Exception SQL Collection Data (see page 221)
- View SQL Error Collection Data (see page 222)
- View a History Display (see page 223)
- Stop Collection (see page 224)

Access the Tutorial

You can access the CA Detector online tutorial for a summary of CA Detector functions and features, information about new features, and information about contacting Technical Support.

To access the CA Detector tutorial, enter T in the Option line on the CA Detector Main Menu and press Enter.

The Detector Tutorial - Help panel appears.

Access CA Detector

You can access CA Detector from the CA Database Management Solutions for DB2 for z/OS Main Menu.

To access CA Detector, enter DT in the Option line on the CA Database Management Solutions for DB2 for z/OS Main Menu and press Enter.

The CA Detector Main Menu appears. From this panel, you can access all CA Detector options.
Initialize a Datastore

A datastore is a set of data sets used to retain historical CA Detector data. CA Detector collects data over a specific interval of time. When the time interval expires, the collected data is discarded and new data collection begins. If you want to review the data from previous intervals, then you must instruct CA Detector to save the old data in a datastore before it begins collecting for a new interval.

To initialize a datastore, you must create data sets that will consume DASD. If this is the first time you have used CA Detector, you might want to skip this section and come back to it later. You can go to the Create a Collection Profile section without creating a datastore. However, if you skip this section, you will not be able to collect exception SQL data and therefore will not be able to perform the steps detailed in View Exception SQL Collection Data.

To initialize a datastore

1. Type 7 in the Option line on the CA Detector Main Menu and press Enter.
   The Create Datastore Display appears.
2. Specify the following information to create the datastore:
   - Enter a subsystem identifier in the Datastore SSID field. This specifies that the datastore is to retain CA Detector-collected data for the specified subsystem.
   - Enter a high-level qualifier in the High Level field. This specifies a high-level qualifier for the CA Detector datastore.
   - Type TUTORIAL in the Datastore Name field. This field provides a name for the CA Detector datastore. See Authorization and Security in the “Operational Considerations” chapter for information about authorization.
   - Type 005 in the Max intervals field. This specifies that CA Detector will store five intervals in the datastore. When CA Detector completes a sixth interval, it erases the first interval and stores the sixth interval in the space previously held by the first interval.
   - Type Y in the Exception SQL field to specify that you want the CA Detector datastore to store data for exception SQL collection.
   - Type Y in the SQL Error field to specify that you want the CA Detector datastore to store data for SQL error collection.
   - Type N in the Batch JCL field to specify that you want CA Detector to create the datastore online (rather than in batch) in this TSO session.

Press Enter.

The Standard Activity Collection Dataset display appears.
3. Specify standard activity SQL collection data set information as follows:
   - Enter a volume name in the Volume field. Do this only if your installation does not use SMS.
   - Type **CYLS** in the Alloc unit field to specify that the primary and secondary space quantities are to be allocated in cylinders.
   - Type **0050** in the Primary space field to specify that primary space allocations should be 50 cylinders.
   - Type **0025** in the Secondary space field to specify that secondary space allocations should be 25 cylinders.
   - Enter SMS information in the SMS MGMT class, the SMS STORAGE class, and the SMS DATA class fields. Enter these options only if your installation uses SMS. Names will be installation-specific; if necessary, see your database administrator for SMS information.

Press Enter.

The Exception SQL Collection Datasets display appears.

4. Specify the following allocation criteria:
   - For datastore data set allocation criteria:
     - Enter a volume name in the Volume field. Do this only if your installation does not use SMS.
     - Type **CYLS** in the Alloc unit field to specify that the primary and secondary space quantities are to be allocated in cylinders.
     - Type **0050** in the Primary space field to specify that primary space allocations should be 50 cylinders.
     - Type **0020** in the Secondary space field to specify that secondary space allocations should be 20 cylinders.
     - Enter SMS information in the SMS MGMT class, the SMS STORAGE class, and the SMS DATA class fields. Enter these options only if your installation uses SMS. Names will be installation-specific; if necessary, see your database administrator for SMS information.
   - For datastore workfile dataset allocation criteria:
     - Enter a volume name in the Volume field. Do this only if your installation does not use SMS.
     - Type **CYLS** in the Alloc unit field to specify that the primary and secondary space quantities are to be allocated in cylinders.
     - Type **0040** in the Primary space field to specify that primary space allocations should be 40 cylinders.
Initialize a Datastore

- Type **0020** in the Secondary space field to specify that secondary space allocations should be 20 cylinders.

- Enter SMS information in the SMS MGMT class, the SMS STORAGE class, and the SMS DATA class fields. Enter these options only if your installation uses SMS. Names will be installation-specific; if necessary, see your database administrator for SMS information.

Press Enter.

The SQL Error Collection Dataset panel displays.

5. Specify the following information for the data sets to hold SQL error collection:

- Enter a volume name in the Volume field. Do this only if your installation does not use SMS.

- Type **CYLS** in the Alloc unit field to specify that the primary and secondary space quantities are to be allocated in cylinders.

- Type **0050** in the Primary space field to specify that primary space allocations should be 50 cylinders.

- Type **0025** in the Secondary space field to specify that secondary space allocations should be 25 cylinders.

- Enter SMS information in the SMS MGMT class, the SMS STORAGE class, and the SMS DATA class fields. Enter these options only if your installation uses SMS. Names will be installation-specific; if necessary, see your database administrator for SMS information.

Press Enter.

The Create Datastore Display appears with the following message at the top of the page under the Command line:

DT603I: Detector datastore initialization successfully processed

**More information:**

[Create a Datastore](#) (see page 40)
Create Collection Profiles

Collection profiles are used to control and customize dynamic SQL collection and to manage and control SQL error collection. You can set up a collection profile to control the SQL activity captured. For example, you can set CPU time, getpage, or rows returned thresholds that limit the SQL activity captured.

Collection profiles can also control SQL error collection by excluding SQL error conditions which are of no interest, or limiting the amount of SQL error information to be retained. You can create a collection profile to collect exception SQL and SQL error data.

Follow these steps:

1. Type 4 in the Option line on the CA Detector Main Menu and press Enter.
   The Collection/Reporting Profiles Menu appears.
2. Type 1 in the Option line and press Enter.
   The Collection Profiles Display appears.
3. Type S in the Create Profile field and press Enter.
   The Create Collection Profile Display appears.
4. Complete the following fields:
   - **Profile SSID**
     Identifies the subsystem where the collection profile will be used.
   - **Profile ID**
     Provides a unique identifier for the collection profile. For example, you can specify the subsystem identifier as the first four characters and 0000 as the last four characters.
   - **Profile desc**
     Provides a description of the collection profile. For example, you could specify TUTORIAL_TEST_PROFILE.
   - **Highlevel**
     Specifies a high-level qualifier.
   - **Volume**
     Specifies a volume name. This field is required only if your site does not use SMS.
   - **Alloc unit**
     Specifies the allocation unit to use for primary and secondary space quantities. Valid values are T for tracks and C for cylinders.
Create Collection Profiles

**Primary space**

Specifies the initial space allocation in tracks or cylinders. For example, specify 0005 to allocate five tracks or cylinders based on the value specified for Alloc unit.

**Secondary space**

Specifies the secondary space allocation quantity in tracks or cylinders. For example, specify 0005 to allocate five tracks or cylinders based on the value specified for Alloc unit.

**SMSMGT class, SMS STORAGE class, and SMS DATA class**

Specifies SMS information if your site uses SMS. The names are site-specific. You may need to contact your database administrator for this information.

Press Enter.

The profile is created and the following message appears:

DT624I: Detector profile successfully initialized

This profile uses the sample application and resource groups created automatically by CA Detector.

**More information:**

- Using Collection Profiles (see page 137)
- Using Collection Services (see page 31)
Start Collection

To start the data collection process, specify the SSID for which CA Detector collects data and the time interval for each collection. In addition, specify whether the collected data is written to a datastore for later review. You also indicate whether to collect information about exception SQL and SQL error activity.

**Note:** A collection profile is required to start collection.

**Follow these steps:**

1. Type **5** in the Option line on the CA Detector Main Menu and press Enter.
   
   The Start Collection Display appears.

2. Specify the following information to define collection processing:

   - **Type 01:00** in the Interval Time field to specify that CA Detector is to gather and retain data in one-hour periods. When this interval has passed, CA Detector automatically reinitializes and begins collection for a new one-hour period. The statistics and information that CA Detector displays represents activity over a period no greater than the interval you specify here.

   - **Type N** in the Sysplex Interval field. For the purposes of this Quick Steps tutorial, you do not engage this parameter. The Sysplex Interval is an option to the Interval Time parameter. This interval is an Xmanager-managed time interval that can be used to provide a consistent collection interval across all collections in a Sysplex complex.

   - **Type 08:00** in the Time Limit field to specify that CA Detector is to stop collection after eight hours.

   - **Type Y** in the Externalize field to specify that CA Detector is to save its collected data to a datastore. When the interval time has elapsed, CA Detector reinitializes its data collection. If you did not create a datastore previously in the Initialize a Datastore section, then enter **N**.

   - Enter the high-level qualifier for the datastore you created earlier in the High Level field. Enter this field only if you entered **Y** in the Externalize field.

   - **Type TUTORIAL** in the Datastore Name field. Enter this field only if you entered **Y** in the Externalize field.

   Press Enter.

   The second Start Collection Display appears.
3. Specify the following information to define collection processing further:
   - Type N in the Triggered SQL collection field. Doing so specifies that you do not want to collect SQL statistics for triggered SQL activity and SQL originating from native SQL stored procedures.
     
     **Note:** Collecting triggered SQL activity or SQL originating from native SQL stored procedures introduces a slight additional overhead.
   - Type N in the Plan Exclude List field. For the purposes of this procedure, your collection profile does not include a Plan Exclude List.
   - Type Y in the Dynamic SQL field to indicate you want to collect dynamic SQL data.
   - Type Y in the Static SQL field to indicate you want to collect static SQL data.
   - Type Y in the SQL Error field to indicate you want to collect SQL error data.
   - Type Y in the Host variable collection field to indicate you want to collect host variable data.
   - Type 1 in the Exception cache size field. Doing so indicates you want the number of 1-MB blocks in the exception collection storage cache to be 1.
   - Enter the name of your collection profile in the Collection Profile field.
   - Press Enter.

   The first Start Collection display appears and displays the following message appears:

   DT202I: START REQUEST HAS BEEN SUCCESSFULLY PROCESSED.

   **Note:** If you receive a message that collection is already active, CA Detector is already collecting information for the SSID you specified.

---

**View Standard Activity Collection Data**

You can use CA Detector to view standard collection data on application workload and resource use for the collections you have started. You begin at the plan level, view programs within one of the plans, and, finally, view the SQL calls and call text issued by a specific program.

**Follow these steps:**

1. Type 1 in the Option line on the CA Detector Main Menu and press Enter.
   - The Planname Summary Display appears.

2. Type S in the field before the planname you want to select and press Enter.
   - The Plan Program Display panel appears. This panel identifies the programs (DBRM and packages) for the selected plan.
3. Type $ in the field before the name of the program you want to select and press Enter.

   The Plan SQL Display panel appears. From this panel, you can view the SQL call types and statement numbers for the SQL activity that has been executed from the program being viewed. You can also identify the SQL call types and statement numbers for the program (DBRM or package) you selected.

4. Type $ in the field before the SQL call you want to select and press Enter.

   The SQL Call Text Display appears. This panel shows the call text originating from the static SQL application.

   Note: To invoke EXPLAIN from CA Plan Analyzer, type $ in the Select for the SQL statement field and press Enter. The PPA Quick Explain Data Editor display appears. The fields on this display are filled in with the information about the SQL statement you selected. A license for CA Plan Analyzer is required. For more information about using Quick Explain, see the “Sample Scenarios” chapter or the CA Plan Analyzer for DB2 for z/OS User Guide.

---

**View Exception SQL Collection Data**

You can use CA Detector to view exception SQL collection data on application workload and resource use. You begin at the user level, view exception requests captured by plan for a user, view the reasons that the exception request was captured, and, finally, view the SQL statements issued by the request.

Because exception SQL collection requires externalization of data, you can only complete this procedure if you created a datastore.

You can also view collection data using history displays instead of the current displays. See View a History Display for more information.

**Follow these steps:**

1. Type 1 in the Option line on the CA Detector Main Menu and press Enter.

   The Planname Summary Display appears. This panel shows the standard activity collection data.

2. Type X in the View Type field and press Enter to access the exception SQL view type.

   The Exception SQL User Summary Display appears. This panel lists all users who are executing exception SQL requests.

3. Type $ in the line command field before the OPID and press Enter to select a user.

   The Exception SQL Request Summary Display appears. This panel lists the exception SQL request captured by plan for the user during the collection interval.
4. Identify why the exception SQL request was collected:
   ■ Type **MAX** in the Command field to scroll to the last column of the display.
   ■ Press PF11 (Right) to execute the scroll. The display scrolls to the TRANSACTION_NAME column.
   ■ Press PF10 (Left) nine times to scroll left six columns.

5. Analyze the reason for the collection.
   The APPLICATION_GROUP, RESOURCE_GROUP, THRESHOLD, and PROGRAM columns appear next to the START_TIME column.
   The values in the APPLICATION_GROUP and RESOURCE_GROUP columns show which application group and resource group contain the threshold values that were applied to the exception request. The value in the THRESHOLD column indicates why CA Detector collected the exception SQL request.

6. Type **MAX** in the Command field and press PF10 (Left) to execute the scroll and return to the previous position on the panel.

7. Type **S** in the command field before START_TIME and press Enter to view the SQL call statements for the exception request.
   The Exception SQL Request Detail display appears. This panel lists the SQL call statement types that make up the exception SQL request you selected for viewing.

---

**View SQL Error Collection Data**

You can use CA Detector to view SQL error data collected on applications. You begin by viewing the SQL codes generated during collection, viewing the users that generated the code, and, finally, viewing the SQL calls and call text issued by the user for the code.

**Follow these steps:**

1. Type **1** in the Option line on the CA Detector Main Menu and press Enter.
   The Planname Summary Display appears. This panel shows the standard collection data.

2. Type **E** in the View Type field and press Enter.
   The SQL Error Summary Display appears. This panel lists all the SQL error codes that have occurred during the current collection interval.

3. Type **S** in the line command field before the SQLCODE column and press Enter to select the SQL code for which you want to view error activity.
   The SQL ERROR Users/SQLCODE Summary display appears. This panel summarizes by user the SQL error data for the SQL error code being viewed.
4. Type $ in the line command field next to the OPID column and press Enter to select the user.

The SQL Error Detail Display appears. This panel lists the SQL call statement types that generated the SQL error being viewed. Use the PF10 and PF11 keys to scroll left and right through the data columns.

5. Type $ in the line command field before the PROGRAM column and press Enter to select the SQL statement.

The SQLCA Data Area Display appears. This panel shows the SQLCAQ area for the SQL call that issued the return code.

6. Type X in the Display field and press Enter to display the SQLCA area in hexadecimal or enter M and press Enter to display the SQLCA area messages.

**View a History Display**

You can save collected data for later review using a datastore if at least one CA Detector collection time interval has passed, and the collected data has been externalized.

**Follow these steps:**

1. Type 2 in the Option field from the main menu and press Enter.

   The Datastore Display panel appears. From this panel, you can select a datastore to view, view audit data on the intervals externalized in the datastore, or delete a datastore.

2. Type $ in the line command field to select a datastore and press Enter.

   The Datastore Interval Display panel appears. This panel shows the collection intervals that are available in the selected datastore. From this display, you can evaluate application workload at various times of the day for the subsystem you are viewing. You can also select a specific interval for further display.

3. Type $ in the line command field before the DATE field and press Enter to view the data collected during the selected interval.

   The Planname Summary Display panel appears. This panel shows data collected during the interval you selected. You can examine this data as described in View Exception SQL Collection Data and View SQL Error Collection Data.

**More information:**

[Using Collection Services](#) (see page 31)
Stop Collection

You can stop collection for a specific subsystem.

**Follow these steps:**

1. Type 6 in the option line on the CA Detector Main Menu and press Enter.
   The Terminate Collection Display appears.

2. Enter a subsystem identifier in the DB2 SSID field and press Enter to identify the subsystem for which you want to stop data collection.
   If the termination is successful, the following message appears:
   DT204I: Stop request has been successfully processed
## Index

### %
- % masking symbol • 18

### _
- _ masking symbol • 18

### A
- access path analysis • 206
- accounting trace values • 203
- activity collection data set • 43
- allocate datastore space • 43
- application groups
  - about • 141
  - modifying plan entries • 146, 174
  - viewing plan entries • 146
- authorizations
  - and security • 19
- AUTO keyword • 48

### B
- Batch JCL field • 40
- batch utility
  - batch profile and datastore utilities • 26
  - utility requirements • 27

### C
- CA Plan Analyzer interface • 206
- CA SQL-Ease interface • 209
- CA Subsystem Analyzer
  - interface • 14
  - panel integration with • 122
  - synchronizing collection with • 59
- call text • 115
- CDBAPXMP • 58
- CHV keyword • 48
- collect distributed SQL traces • 179
- collection
  - controlling using MVS modify commands • 48
  - datastore space • 43
  - externalization options • 32
  - overhead • 35
  - starting • 32
  - stopping • 39, 56
- Collection Profile field • 32
- collection profiles
  - about • 137
  - creating • 138
  - data set information • 138
  - data set name • 138
  - using MVS modify command • 56
- collection services • 32
  - about • 31
  - exception collection storage cache • 38
  - externalization options • 32
  - interval time and time limit options • 32
  - overhead • 35
  - SQL collection options • 32
  - SQL exception collection options • 32, 45, 46
- commands
  - EXCLUDE • 16
  - EXPAND • 18
  - FREEZE • 16
  - K (keys) • 66, 70, 73, 79, 87
  - LISTDB2 • 202, 203
  - LISTTHD • 203
  - MVS modify • 48
  - PDT global • 15
  - PLEXINTV • 32
  - PROF(DTR) • 56, 138
  - PSTAT(DTR) • 38, 48, 57, 137
  - REPORT • 16
  - RUN • 58, 59
  - SCROLL • 18
  - SETWIDTH • 18
  - SORT • 16
  - SQLTRACE • 177, 179, 181, 192
  - START(DTR) • 48, 59
  - START(SQLTRACE) • 193
  - STOP(DTR) • 48, 56
  - STOP(SQLTRACE) • 198
- control file • 13
- create batch reports • 27
- Create Collection Profile Display • 138
- cross-system support • 23, 38
- current profile • 171
D

data set authorizations • 22
data sharing groups
  accessing SQL error collection • 46
datatstores
  creating • 40
  Datastore Display panel • 115, 131
  Datastore Name field • 32, 40
  Datastore SSID field • 40
  estimating size of for SQL error • 46
  for exception SQL collection • 45
  initializing • 214
date and time formats • 16
DB2 keyword • 48, 56, 57
DB2 SSID field • 23
DB2 SSID/Location Selection panel • 15
DB2 trace requirements for collection • 25
Delete this member field • 23
distributed communications environment • 23
DST keyword • 48
DTTR0000 • 23
DTTRxx • 23
DUR keyword • 48
DYN keyword • 48
Dynamic SQL field • 32
Dynamic SQL Text Statistics field • 32

E

error messages • 48, 193
estimating datastore size • 43
evaluate access path information • 206
Exception SQL field • 40
exceptions
  cache pool monitoring • 57
  Cache Size field • 32
  SQL collection panels • 93
EXCLUDE command • 16
EXPAND command • 18
EXT keyword • 48
external security • 20
Externalize field • 32
externalizing data • 59

F

FREEZE command • 16

G

global commands • 14

H

high-level field • 32, 40
history displays, how to view • 131
HLVL keyword • 48
host variable collection field • 32

I

IFCID • 179
Index Activity Display panel • 123
INT keyword • 48
integrated data sharing statistics • 38
integration with CA Subsystem Analyzer • 122
interface with CA Plan Analyzer • 14
interface with CA Subsystem Analyzer • 14
interface with Thread Termination\Dynamic
  DSNZPARM • 14
interproduct functionality • 14
Interval Time field • 32

K

K (keys) command • 66, 70, 73, 79, 87

L

LISTDB2 global command • 202, 203
LISTTHD global command • 203
load a collection profile • 56
LPAR requirements • 38

M

masking
  % symbol • 18
  _ symbol • 18
Max intervals field • 40
Member field • 23
MVS authorization requirements • 22
MVS modify commands
  executing from Xmanager parmlib • 58
START(DTR) command • 48, 59
starting collection • 48
stopping collection • 56
using • 56, 57
N
Number of 32K Writer Buffers field • 23

P
PDT global command • 15
percent masking symbol (%) • 18
PLEXINTV command • 32
PLEXINTV keyword • 48
PLNX keyword • 48
POOL keyword • 48
PRF keyword • 48
product authorization • 20
product features • 12
PROF(DTR) command • 56, 138
profile parameters • 23
PSTAT command • 57
PSTAT keyword • 57
PSTAT(DTR) command • 38, 48, 57, 137

R
Read Interval fields • 23
REPORT command • 16
REPORT RESET command • 16
reporting profiles
about • 169
changing • 171
creating • 170, 205
data set name • 170
data set size for • 170
resource groups • 154
retain exception SQL data • 42
RND keyword • 48
Round Interval field • 32
RUN command • 58, 59

S
sample SYSIN statements
batch utility • 27
examples • 27
SCA keyword • 48
scroll and expand fields • 18
SCROLL commands • 18
SDE keyword • 48
select an SSID/location • 15
SETWIDTH command • 18
SORT command • 16
SQL
call text • 115
collection options • 32
error collection panels • 46, 106
SQL Error field • 32, 40
SQL error groups • 162
SQL Error Text field • 32
SQL traces
and DB2 performance • 179
collect distributed • 179
controlling using modify commands • 193
SQL Trace component • 177
stopping using MVS modify command • 198
with CA SQL-Ease • 209
SQLT keyword • 48
SQLTRACE global command • 177, 179, 181, 192
SQT keyword • 48
start collection • 32, 219
START(DTR) command • 48, 59
START(SQLTRACE) command • 193
static SQL field • 32
stop collection • 39, 224
STOP keyword • 56
STOP(DTR) command • 48, 56
STOP(SQLTRACE) command • 198
synchronizing collection • 59
SYSIN statement examples • 27
sysplex interval • 32, 38
sysplex support • 23

T
Thread Termination\Dynamic DSNZPARM interface • 203
time formats • 16
time limit field • 32
Trace Buffer Size field • 23
trace requirements for collection • 25
TRG keyword • 48
tiggered SQL collection field • 32

U
underscore masking symbol (_) • 18

V
VCT keyword • 48
viewing
call text • 115
dynamic SQL exceptions • 83
extection SQL collection data • 221
global defaults for collection profiles • 139
history display • 223
individual threads • 203
list of active threads • 203
reporting profiles • 170, 172
SQL error collection data • 222
standard activity collection data • 220
subsystem list • 202
thread activity • 203

X

Xmanager
about • 23
create parmlib member • 58
cross-system support • 38
executing from a parmlib • 58
PLEXINTV command • 32
start and stop collection • 58