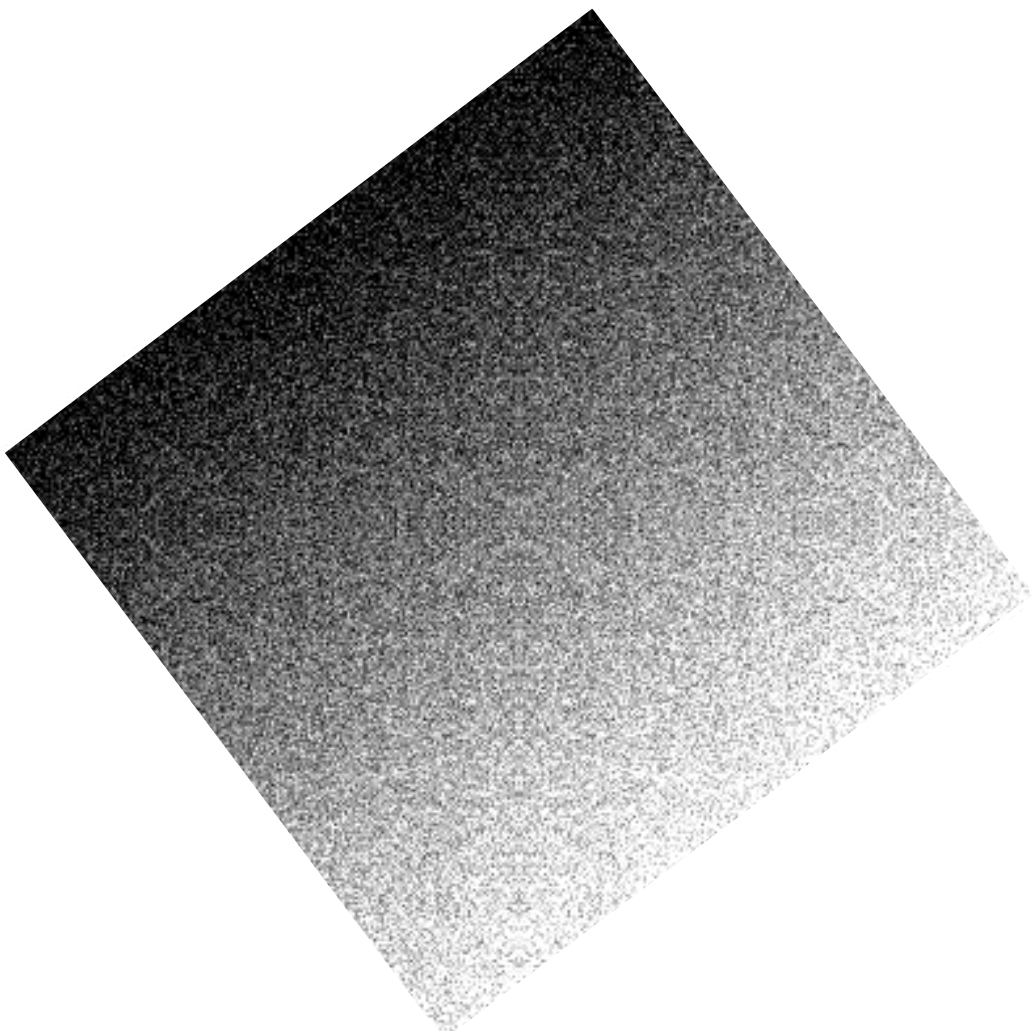
Intelligent Decision Server

IBM

Query User's Guide

Version 1



SC26-8901-00

Intelligent Decision Server

IBM

Query User's Guide

Version 1

SC26-8901-00

Note

Before using this information and the product it supports, be sure to read the general information under “Notices” on page vi.

First Edition (October 1996)

This edition applies to Version 1 of Intelligent Decision Server, 5639–IDS, and to any subsequent releases until otherwise indicated in new editions or technical newsletters. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1996. All rights reserved.**
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Notices vi
Trademarks vi

About This Book viii
Who Should Read This Book viii
We'd Like Your Comments viii

Chapter 1. What Is Query? 1
Querying a Database 1
Formatting and Presenting the Results 3

Chapter 2. Getting Started with Query 5
Configuring a Query Icon 5
Securing a Query Icon 7
Navigating Through Query Windows 8
Query Control Window 8
Query Catalog Window 9
Query Window 11
Understanding Associated Windows 13

Chapter 3. Building a Query 15
Selecting Items from the User Catalog 15
Retrieving One Table 17
Retrieving Several Columns 18
Copying Columns One at a Time 18
Copying Several Columns Together 18
Joining Tables 19
Understanding Default Joins 20
Creating Manual Joins 20

Chapter 4. Constraining Data 23
Constraint Types 23
Single Values 24
Multiple Values 24
Wild Cards 25
Date Arithmetic 26
@-Variables 26
Applying Constraints 27
Specifying the Constraint Source 28
Working in the Comparison Options Window 29
Renaming the Column 30
Selecting an Operator 30
Entering a Value 30
Selecting Values from the Choices Window 32
Displaying Constrained Choices 34

© Copyright IBM Corp. 1996

Changing or Removing a Constraint 35
Using More than One Constraint 36
Setting Options in Joins 37

Chapter 5. Editing a Query 39
Rearranging Query Data 39
Sorting Data 39
Eliminating Duplicate Rows 40
Changing a Column Heading 41
Using a Different Database Category 41
Looking at the DBA Catalog 41
Selecting a New Database Category 42

Chapter 6. Working with Query Data 44
Retrieving Data from the Database 44
Displaying the Query Data 44
Stopping the Query 44
Capturing the SQL for a Query 45
Using the Contents of the Query Log Window 45
Stopping the SQL Capture 46
Saving Query Data 46
Removing Query Data 47
Clearing Retrieved Data 47
Clearing the Query 47
Deleting Parts of a Query 47
Printing Query Data 48

Chapter 7. Using Query Views 49
Creating a View 49
Changing a View 50
Using Multiple Views 51
Creating Two Views 51
Comparing Views 52
Performing Calculations with Views 52

Chapter 8. Performing Calculations 54
Creating a Computed Column 54
Building the Query 54
Understanding the Computed Columns Options Window 54
Setting Up the Calculation 56
Using Decimal Arithmetic to Calculate Results 57
Changing the Calculation 58
Before Retrieving Computed Data 58
After Retrieving Computed Data 58

Chapter 9. Using Expert Functions to Constrain Data 60
Adding Expert Functions 60
Using the General Qualifier 60

Entering Text Strings 61
Using Wild Cards 61
Specifying Embedded Blanks 62
Using Query Strategy 62

Chapter 10. Using Query in Capsule Applications 63
Setting Up the Application 63
Controlling the Data Flow 64
Stopping the Data Flow 65
Transferring Data Attributes 65
Specifying a Destination for Query Data 65
Selecting a Function for the Source Data 66
Controlling Nested Capsule Applications 66
Using Queries to Set Controls in Capsule Applications 66
Connecting a Query to a Nested Capsule Application 67

Chapter 11. Modifying the Database 68
Editing the Database 68
Deleting Rows of Data 70
Inserting Rows of Data 70

Chapter 12. Troubleshooting 71
Retrieving Data 71
Viewing Retrieved Data 73
Verifying Retrieved Data 74
Opening a Query Icon 75
Working with Views 75
Enhancing Performance 76

Glossary 77

IDS Publications 83

Index 84

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling (1) the exchange of information between independently created programs and other programs (including this one) and (2) the mutual use of the information that has been exchanged, should contact:

IBM Corporation
555 Bailey Avenue, W92/H3
P.O. Box 49023
San Jose, CA 95161–9023

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks

The following terms are trademarks of IBM Corporation in the United States or other countries or both:

ACF/VTAM	DXT	PROFS
AIX	Enterprise System/9000	PS/2
Application System/400	ES/9000	QMF
APPN	IBM	RACF
AS/400	MVS/ESA	RISC System/6000
Capsule	MVS/XA	RS/6000
DATABASE 2	OfficeVision	S/390

© Copyright IBM Corp. 1996

DataJoiner	OfficeVision/VM	SQL/DS
DB2	Operating System/2	System/360
DB2 for OS/2	Operating System/400	System/370
DB2/6000	OS/2	Virtual Machine/Enterprise Systems Architecture
DFSORT	OS/400	VM/ESA
Distributed Relational Database Architecture	Personal System/2	VTAM
DRDA	Presentation Manager	Win-OS/2

C-bus is a trademark of Corollary, Inc.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

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

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

About This Book

The IBM Intelligent Decision Server (IDS)*Query User's Guide* explains how to perform database queries and use retrieved data.

Who Should Read This Book

This guide is intended for all IDS users who need to retrieve and update current database values, to create calculations, or to build complex reports. Procedures are provided for the beginning Query user and for the expert application developer.

We'd Like Your Comments

Your feedback is important to us in providing you with the most accurate and high-quality information. If you have any comments about this book or any other IDS documentation, please visit our Website at:

http://www.dss.ibm.com

There you'll find the Feedback Connection page where you can enter comments and send them to us.

© Copyright IBM Corp. 1996

Chapter 1. What Is Query?

The Query tool allows you to retrieve data that is stored in your company's databases. Using your mouse, you can select the tables you want to access, and then specify various types of selection criteria. Query is a powerful tool that users at all levels of expertise can employ to:

Specify the databases that you want to access

Choose the type of information you need

Apply constraints to further define the data to suit your needs

Create calculated columns of retrieved data

Because the Query tool is based on objects that are selectable with a mouse, you can build elaborate sequences of operations without using Structured Query Language (SQL). For advanced users, SQL-based editing capability is available and can be used with expert functions.

Retrieved query data can be included in a text document, formatted into a free-form report, passed to a spreadsheet, or graphically charted using the Plot tool. In addition to its basic features, you can use the Query tool to perform calculations, and combine various types of data for comparison and analysis. Calculated results can also be sent to other tools, such as Spreadsheet, Text, and Plot.

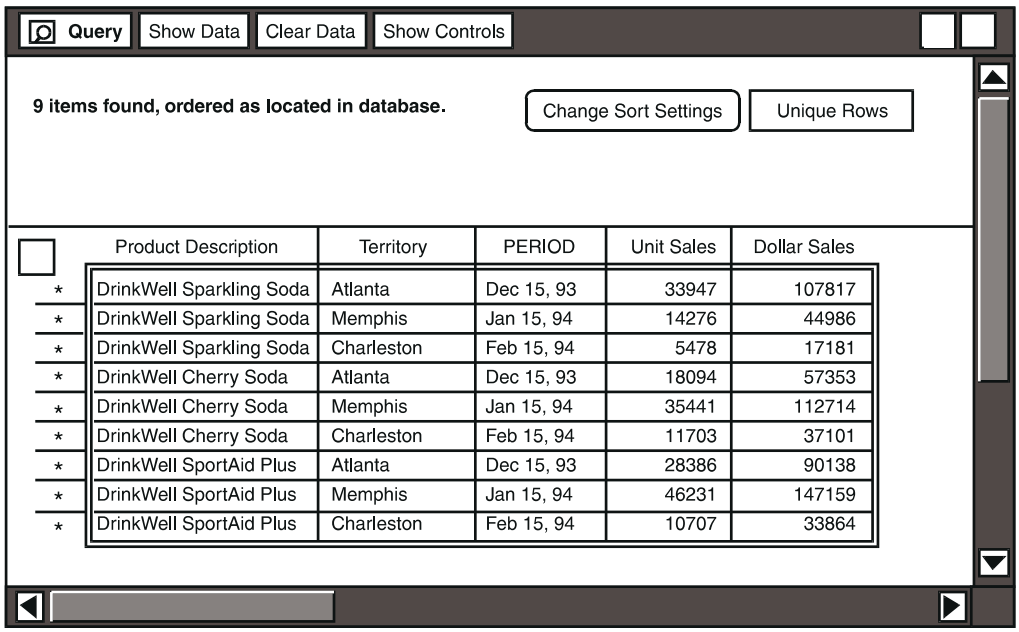
Using Query and	You can perform the following tasks
Capsule applications	Perform database retrieval.
	Transfer data to other tool icons to automate the data access, data analysis, and desktop management sequence.
Layout	Transfer data and format it into a free-form report.
Plot	Show the retrieved data in a graphic representation.
	Print the charted results.
Spreadsheet	Perform analysis and complex computations.
	Print results.
	Edit a database.
Text	Format your data for presentation or inclusion in a document.
	Print the results.
Transformer	Manipulate the retrieved data.
	Sort or remove null values.

Querying a Database

The Query tool provides a view of the databases your company uses. A typical database might list consumer packaged goods, personnel information, or products that your

© Copyright IBM Corp. 1996

company produces. For example, you might need to query a database to discover what products netted greater sales in a three-month period. A sample output for such a Query is shown in Figure 1 on page 2.



	Product Description	Territory	PERIOD	Unit Sales	Dollar Sales
*	DrinkWell Sparkling Soda	Atlanta	Dec 15, 93	33947	107817
*	DrinkWell Sparkling Soda	Memphis	Jan 15, 94	14276	44986
*	DrinkWell Sparkling Soda	Charleston	Feb 15, 94	5478	17181
*	DrinkWell Cherry Soda	Atlanta	Dec 15, 93	18094	57353
*	DrinkWell Cherry Soda	Memphis	Jan 15, 94	35441	112714
*	DrinkWell Cherry Soda	Charleston	Feb 15, 94	11703	37101
*	DrinkWell SportAid Plus	Atlanta	Dec 15, 93	28386	90138
*	DrinkWell SportAid Plus	Memphis	Jan 15, 94	46231	147159
*	DrinkWell SportAid Plus	Charleston	Feb 15, 94	10707	33864

Figure 1. Sample output for a Query

Your company might use this type of database query frequently, although some of the parameters for a query might change from month to month. For repetitive processes, you can save a database query view so that you can retrieve that data again without having to rebuild the query.

If you are a database administrator and you sometimes need to make minor modifications to your company's databases, you can use a Query icon combined with a Spreadsheet icon to read data into a database. You can perform this administrative function from your desktop, providing you have the required privileges for doing so.

Placing a Query icon inside a capsule application window permits you to retrieve data each time you click on the **Run** button. You can also take advantage of the variables used in the capsule tool's control window.

A sample application using a Query as the first processed icon is shown in Figure 2 on page 3. In this application, a stand-alone Data Entry icon is used to read the variables from the capsule application. The Query tool retrieves the data, then transfers it to the Spreadsheet tool for analysis. The data is then sent to a Plot tool which is set up to chart appropriate analysis trends, and finally, the results of this application are placed into a folder as a permanent record. You could add a mail tray icon to this sequence and mail your results to another user.

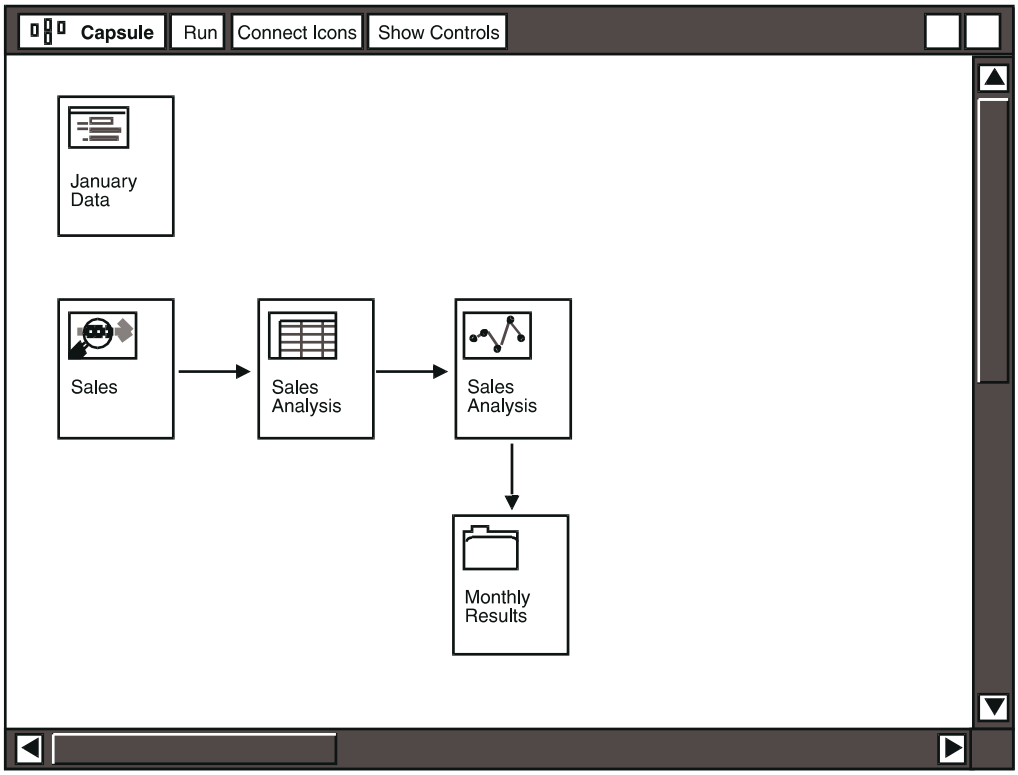


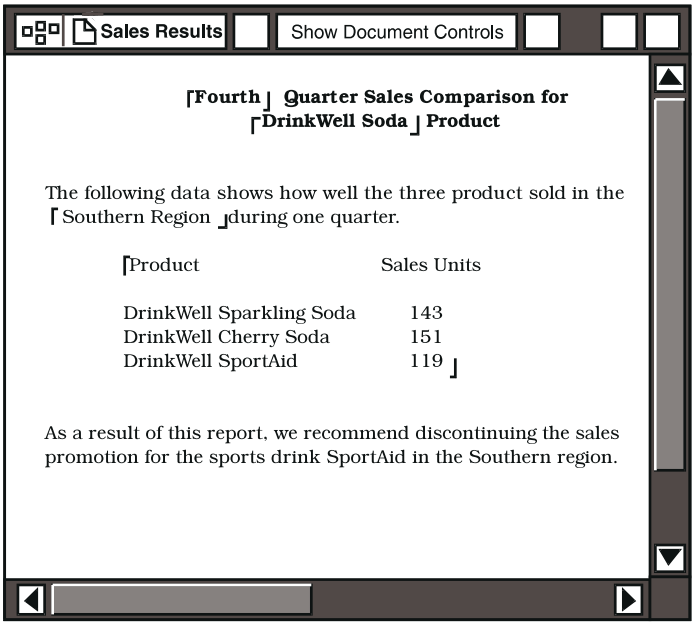
Figure 2. A sample application using a Query as the first processed icon

Formatting and Presenting the Results

After you run the capsule application, transferring the query data to a Text tool, you can view your results and send them to another user. If you wish, your data can be copied to other IDS tools for printing. For example, if you want to chart the data, send it to a Plot icon then print the resulting chart. Or, you can set up regions in a Text window and format the transferred query results in any way you choose. To create a free-form report using Query data, you can connect the Query to a Layout icon.

Figure 3 on page 4 shows a Text icon that contains the results of the 3-month analysis for DrinkWell soda sales. The regions in this report are defined using @-variables which can be modified at the capsule level whenever needed. If sales trends change, the data might require a different report and you could present the same items with a different summary in the Text document.

To graphically depict the discrepancy between product sales trends, you might also choose to transfer the data to a Plot icon and display the results in a pie chart, similar to the one in Figure 3 on page 4.



Query results shown in two different forms of presentation a document and a pie chart.

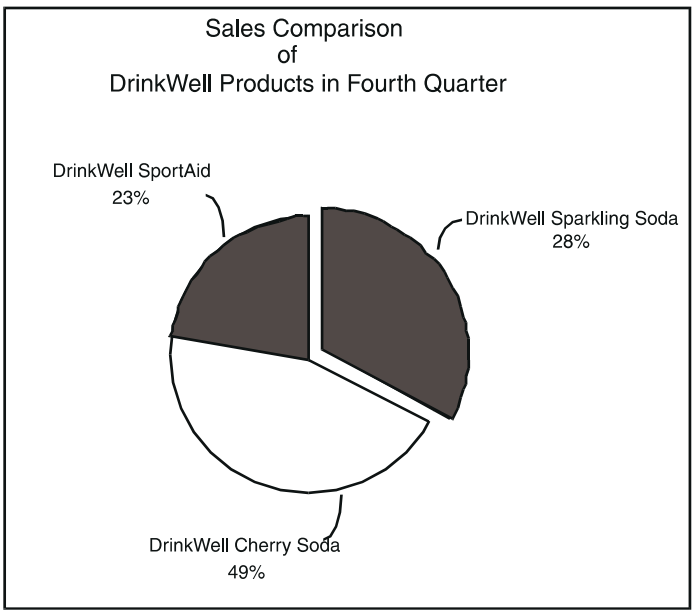


Figure 3. Analysis of DrinkWell soda sales

Chapter 2. Getting Started with Query

Before you can use a Query icon, it must be configured to an appropriate database. After you have a configured icon, you need to know how to use the Query windows. The following subjects are discussed in this chapter:

Configuring a Query icon

Securing a Query icon

Navigating through Query windows

Understanding associated windows

Configuring a Query Icon

After you copy a Query icon from the Blank Icons file drawer, you must configure it for a specific database before you can open it. To configure the icon, you must know the password and the name of the database you want to access. You also need to know which fields are required for your specific case. If you do not have this information, consult your system administrator or database administrator.

To configure a Query icon:

1. Select a Query icon and open the Icon Options window.
2. Select **Other Data** in the **Display** field to display the Other Data page.
3. Using the guidelines provided in Table 1, enter the required information into the text fields and select the appropriate choices in the other fields.

   If your **server** stores more than one physical database, you must identify the database you want to access in the **Database Name** field. It is unnecessary to supply a database name if only one physical database exists, or if you want to access a default database.

Table 1 (Page 1 of 3). Query icon configuration guidelines

Field	Guidelines
Gateway	Type the name of the database gateway that you want the Query icon to access. If the gateway service is on a different realm, immediately follow the gateway name with a colon and the realm name. For example: mydbs:HQ

© Copyright IBM Corp. 1996

Table 1 (Page 2 of 3). Query icon configuration guidelines

Field	Guidelines
Database Name	Type the name of the physical database you want the Query icon to access. For DB2 databases, type the name of the DB2 subsystem you want to access. For SQL/DS databases, type the name of the virtual machine running SQL/DS. For AS/400 databases, it is not necessary to specify anything in this field, because the AS/400 is a single-database system. To connect to a supported, remote IBM database using Distributed Relational Database Architecture (DRDA), immediately follow the local database name with a period and the name of the DRDA server through which the connection is being made. For example: DSN.TD1 To connect to a remote SYBASE or Red Brick database, follow the SQL Server name with a vertical bar and the name of the database you want to access. For example: SYBSERVER2|SYBPROD To connect to a remote ORACLE database, specify the full database name or its alias. For the full database name, specify the standard single-character prefix used for the remote connection (such as T for TCP/IP), followed by a colon, the name of the target host, another colon, and the system ID (SID) of the target ORACLE database. For example: T:host1:DBS
Category	Type the print name or internal name of the database category for this Query icon. The name you specify appears in the window header of the category window.
WTDD Owner	If the Query tool is configured to a database that supports multiple WTDDs and qualified table names, type the name of the owner of the WTDD tables. This field makes it unnecessary to drop and recreate synonyms whenever you access a different WTDD within the same database.
SQL ID	If you are using DB2, type the table qualifier name. The ID entered in the **SQL ID** field overrides the default table qualifier name, which is the name entered in the **User Name** field. If you have questions about the table qualifier name, consult your database administrator.
Database Logon	Select **Specified Here** if you want database access to be based on information you type in the **User Name** and **Password** fields. Select **From Data Access Controls** if you want access to be based on information in the desktop's Data Access Controls window. Consult your database administrator if you have security questions.
Secured	Select **Yes** if you want the icon's **user ID** field to be set to the user ID of the desktop on which it resides. See "Securing a Query Icon" on page 7, for information on securing an icon.

Table 1 (Page 3 of 3). Query icon configuration guidelines

Field	Guidelines
user ID	Type a valid user ID. This might be the same as your desktop user ID depending on how your database security is set up. When the Query icon connects to the RDBMS, this is the user ID sent to the RDBMS. If you are using the DB2 for OS/2 DRDA gateway to connect to a DRDA server and your DB2 for OS/2 user ID is different from your DRDA server user ID, you must specify both user IDs separated by a period. For example: DB2 for OS/2_userid.DRDAID
Password	Type the database user's password. This password might be the same as your desktop password depending on how your database security is set up. For security, the password is never displayed. If you are using the DB2 for OS/2 DRDA gateway to connect to a DRDA server and your DB2 for OS/2 password is different from your DRDA server password, you must specify both passwords separated by a period. For example: DB2 for OS/2_password.DRDA_password

Sometimes your database administrator will configure icons and store them in a file drawer for your use. Configured icons are usually named to indicate the type of data or database category they use.

Securing a Query Icon

When you have built a database query that you wish to protect, you can secure the Query icon so that only those users who are authorized to use the specified database can open the icon. When you secure a Query icon in the Icon Options window, the user ID of the desktop where the icon resides becomes the value in the **User Name** field. When a secured icon is mailed, the value in the **User Name** field changes to the user ID of the recipient. The icon cannot be opened or used by the recipient unless his or her user ID has the appropriate privileges for the database. You cannot change the **User Name** field of a secured icon.

If you receive a secured icon from another user and the database does not recognize your user ID, you will not be able to open the icon.

Important: Securing an icon is an irreversible operation. Before you secure an icon that you plan to distribute to other users, check with your database administrator to ensure that their user IDs are authorized to use the specified database.

The preferred way to secure a Query icon is to configure it to read the user ID and password setting from the desktop's Data Access Controls window. For more information about the Data Access Controls window, see *Working With the Intelligent Decision Server*, or consult your system administrator.

Navigating Through Query Windows

To look at the Query windows, select a preconfigured Query icon. If there are no preconfigured Query icons in the *Query Icons file drawer, consult your system administrator. When you open a new Query icon, three windows appear: the Query window, the Query Catalog window, and the Query Control window.

If only one window appears when you open the Query icon, click on the **Show Controls** button in the Query window, then click on the Query Control window.

When you open a Query icon, the tool must access the database, so it takes a few moments for the windows to appear. If you are opening a preconfigured icon for the first time, the windows usually appear with the Query Control window in front. When opened, the windows can be sized, moved, and repositioned individually.

If you close the Query window, all three windows close. Whenever you reopen a Query icon, the windows appear in the same configuration that you last viewed them. That is, if you previously closed the Query Control and Query Catalog windows, reopening the Query icon causes the Query window to appear by itself. When the Query window appears by itself, its window header contains a button labeled **Show Controls**. Click on this button to reopen the other two windows.

The following sections provide more detail about the specific components of each window. The Query Control window provides the blank area where you build the database query. The Query Catalog window provides the column names from which you select or copy. Finally, the Query window is where you view the results and modify the retrieved data.

Query Control Window

The Query Control window is where you create a depiction of the database query. Its window header buttons are described in Table 2.

Table 2. Query Control window buttons

Button	Function
Show Query	Click on this button to display the column names you have chosen from the tables in the user catalog. When you click on the **Show Query** button, your selections are placed in the Query window and in the Query Control window. For more information about showing the structure of your query, see Chapter 3, “Building a Query” on page 15.
Clear Query	Click on this button to delete the contents of the Query window. This button functions whether or not you have already retrieved data. For more information, see Chapter 6, “Working with Query Data” on page 44. Clicking on the **Clear Query** button is an irreversible operation. Be aware that the query setup and data are removed after you click on this button.

When you have finished selecting data for your query, you can close the Query Catalog and Query Control windows so that just the Query window is open on your desktop. To reopen the Query Control window, click on the **Show Controls** button that appears in the Query window header.

Query Catalog Window

The Query Catalog window displays the column names used to build a database query. The window header contains one button that reflects the type of catalog displayed. The Query Catalog usually shows the user catalog, containing the information in a particular database category. This catalog is the one you are most likely to use and most descriptions in this guide refer to the user catalog page unless otherwise stated. The window header contains the **Show DBA Catalog** button that you use to view the entire database administration catalog for your company. If you click on the **Show DBA Catalog** button, the **Show User Catalog** and **Reset** buttons appear in the window header.

The Query Catalog window provides two display choices**: Column Names** and **Descriptions**. Selecting one of these choices determines whether **Column Names** or **Descriptions** are displayed in the user catalog, and whether **Table Names** and **Descriptions** are displayed in the DBA catalog.

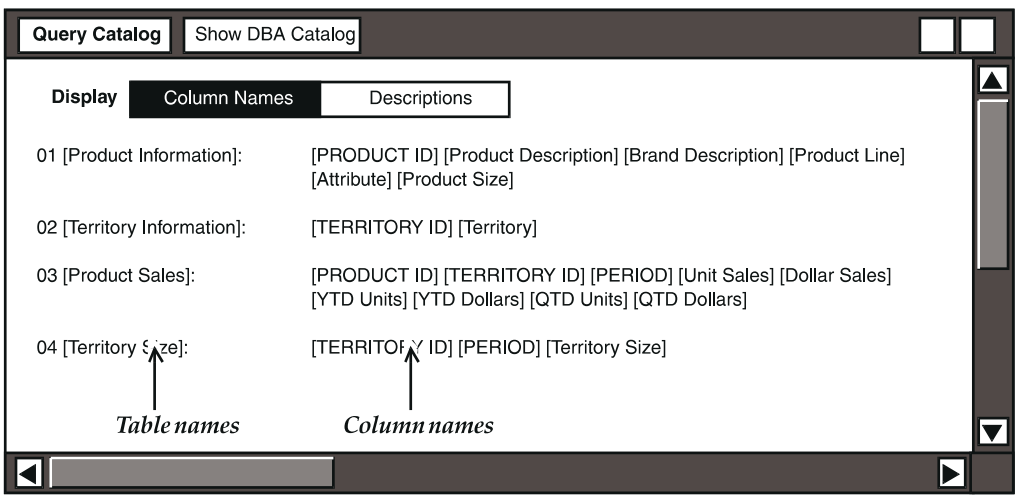


Figure 4. Query Catalog window displaying user catalog column names

In Figure 4, the default choice, **Column Names**, is selected. Choosing **Descriptions** displays database descriptions. Generally, database table descriptions are defined only by the database administrator; you cannot change them from within the Query tool, except when you are saving a Query view. The ability to save views is described in Chapter 7, “Using Query Views” on page 49.

Figure 5 on page 10 shows the user catalog with the column descriptions displayed.

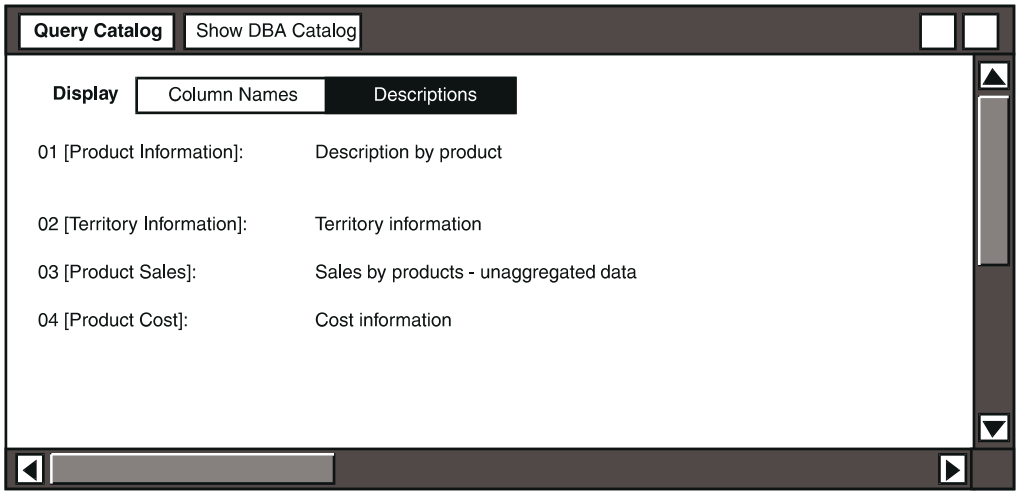


Figure 5. Query Catalog window displaying user catalog descriptions

Below the **Display** options are, from left to right, a column of table numbers, the table names that you can choose from, and descriptions or column names that pertain to these tables.

The table numbers are not selectable; they merely serve to remind you how many tables are available for the query. The table and column names are selectable. Use these items to build your query. Try to think of this portion of the window as a selection area. You cannot change options for any of these items until you have copied them into the Query window or the Query Control window. In the user catalog, you can:

- Select and copy single or multiple tables into the Query Control or Query window
- Select and copy single or multiple columns into the Query Control or Query window
- Change the **Display** option so that it displays table descriptions instead of column names
- Add data to or edit data in the database

The first two techniques are described in detail in Chapter 3, “Building a Query” on page 15. Procedures for changing the **Display** option are described in Chapter 7, “Using Query Views” on page 49. Adding or editing data in the database is discussed in “Inserting Rows of Data” on page 70.

The Query Control window is where you create a picture of your query by copying table and column names from the user catalog, or by selecting one or more items in the user catalog. Figure 6 on page 11 shows a Query Control window with information already in it.

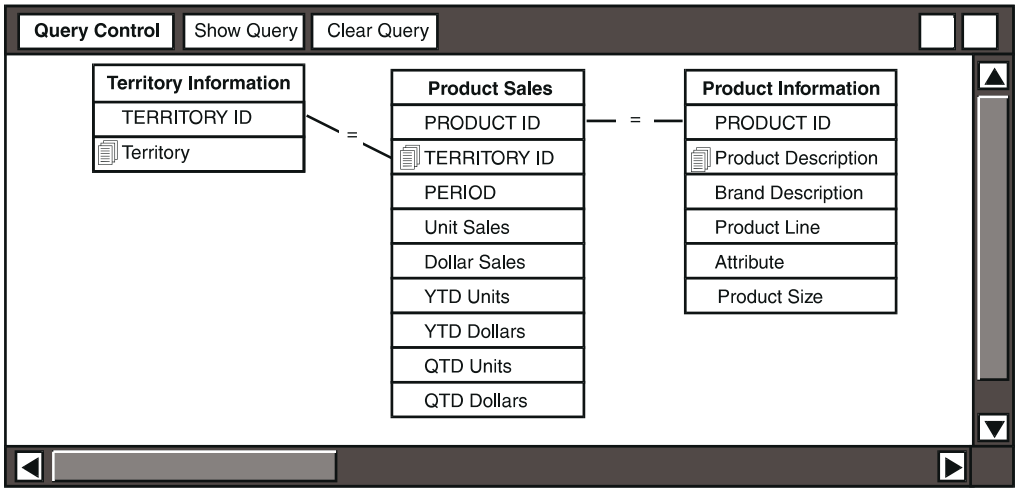


Figure 6. Query Control window

In this window, you can:

Copy column names or tables names

Edit or delete constraints

Create manual joins between columns

Display a depiction of a view that you have created and saved

Techniques for copying column and table names are described in Chapter 3, “Building a Query” on page 15 The procedures for editing or deleting constraints are described in Chapter 4, “Constraining Data” on page 23.

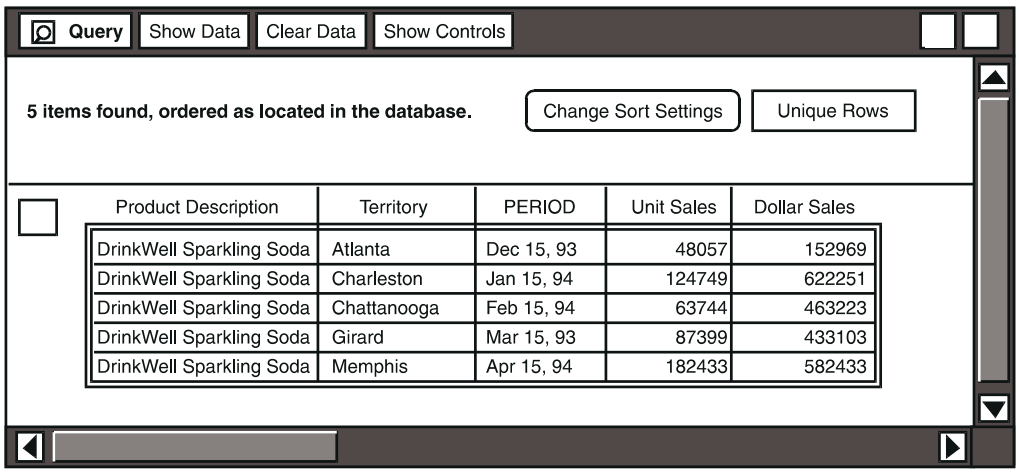
You cannot control the layout of the Query Control window's depiction area. Whenever you add new items and click on the **Show Query** button, the layout of the depiction area is reorganized for maximum efficiency. However, if the area contains many tables, you might have to scroll to see the entire contents.

Query Window

The window behind the Query Control window is called the Query window. Its title block contains the name of the Query icon. The top portion of the Query window contains two side-by-side buttons, **Change Sort Settings** and **Unique Rows**, which can be used to arrange the data in your query. These buttons are discussed in detail in Chapter 5, “Editing a Query” on page 39.

This area of the window is also used to display system messages. When you use the Query tool, informational messages might appear to inform you that the system is acting upon the function you have selected, or indicate when the operation is complete. For example, when you click on the **Show Data** button, you see a message informing you that the Query tool is awaiting data from the database server. Another message appears to tell you that the items have been retrieved.

The results of the query appear below the horizontal line, displayed in rows and columns, similar to the format used by the Spreadsheet tool. This area also contains the **Select All** button, which is useful when you want to transfer data columns into another tool, such as a spreadsheet. Figure 7 on page 12 shows a Query window with retrieved information about one product line.



Product Description	Territory	PERIOD	Unit Sales	Dollar Sales
DrinkWell Sparkling Soda	Atlanta	Dec 15, 93	48057	152969
DrinkWell Sparkling Soda	Charleston	Jan 15, 94	124749	622251
DrinkWell Sparkling Soda	Chattanooga	Feb 15, 94	63744	463223
DrinkWell Sparkling Soda	Girard	Mar 15, 93	87399	433103
DrinkWell Sparkling Soda	Memphis	Apr 15, 94	182433	582433

Figure 7. Query window with retrieved data

Within the Query window, you can perform the following functions:

- Copy column headings from the user catalog or the depiction area
- Delete or modify column headings
- Display only unique rows of data
- Sort column data
- Copy the results of a query into another tool

A prebuilt Query can contain different buttons and fields than those previously described. For example, another user might have chosen to add expert functions to the Query tool. In this case, a button labeled **Create Column** appears in the Query window header and two blank fields are added to the Query Control window. These features are described in Chapter 9, “Using Expert Functions to Constrain Data” on page 60.

The standard Query window buttons are described in Table 3

Table 3 (Page 1 of 2). Query window buttons

Button	Use to
Show Data	Retrieve the data you have specified in the Control window. For more information about showing the structure of your query, see Chapter 3, “Building a Query” on page 15.

Table 3 (Page 2 of 2). Query window buttons

Button	Use to
Clear Data	Remove the retrieved rows of data. For more information about manipulating data, see Chapter 6, “Working with Query Data” on page 44.
Show Controls	Reopen the Query Control window if you previously closed it.

Understanding Associated Windows

Table 4 provides a summary of all the windows associated with the Query tool. Refer to this table when you need to know how to display a specific window, or how to perform a particular task. Use the table to direct you to a chapter where you can find more complete information about each window and its function.

Table 4 (Page 1 of 2). Windows associated with Query functions

Window name	How to display	How to use
Choices	Click on the **Show All Choices** or **Show Constrained Choices** buttons in the Comparison Options window.	Use the choices window to view and select constraint values. For more information, see Chapter 4, “Constraining Data” on page 23.
Column Options	Select a column name in the results area of the Query window, and open the options window.	Use this options window to change a column heading. See Chapter 5, “Editing a Query” on page 39, for more information.
Comparison Options (for column names)	Select a column name in the depiction area, and open the options window.	Use this options window to apply constraints. See Chapter 4, “Constraining Data” on page 23, for more information.
Comparison Options (for joins)	Select a join line in the depiction area and open its options window. You can also select a column name, extend your selection to a second column name, then open the options window.	Use this options window to set options in joins. See Chapter 4, “Constraining Data” on page 23, for more information.
Computed Column Options	Click on the **Create Column** button in the Query window header.	Use this options window to create computed columns. See Chapter 8, “Performing Calculations” on page 54, for more information.
New View Options	Click on Query Control window.	Use this options window to create a view of a query. See Chapter 7, “Using Query Views” on page 49, for more information.

Table 4 (Page 2 of 2). Windows associated with Query functions

Window name	How to display	How to use
Query Catalog	Open the Query icon or click on the **Show Query Catalog** button in the Query Control window header.	Use this window to copy column headings into the **Query Control** or Query window. See the section of this chapter titled "Query Catalog Window" on page 9, for more information.
Query Control	Open the Query icon or click on the Query window header.	Use this control window to build a database query depiction. See "Query Control Window" on page 8, for more information.
Query Log	Click on **Capture SQL** in the Special button menu of the Query window.	Use this capture window to contain a record of the SQL for a query. See Chapter 6, "Working with Query Data" on page 44, for more information.
Sorting Options	Click on the **Change Sort Settings** button in the system area of the Query window.	Use this options window to change the sorting sequence of retrieved data. See Chapter 5, "Editing a Query" on page 39, for more information.

Chapter 3. Building a Query

This chapter explains how to construct a database query and manipulate the retrieved data within the Query windows. It covers the following topics:

Selecting items from the user catalog

Retrieving one table

Retrieving several columns

Joining tables

Selecting Items from the User Catalog

Using the Query tool, you can select and copy the data items into the Query Control window, and then see a visual representation of your database query before you actually retrieve data. The user catalog, introduced in Chapter 2, “Getting Started with Query” on page 5, displays an index of information for a particular database category. By just clicking on items in the user catalog, you can choose any combination of column data to build your query.

To select column data from the user catalog:

1. Open a Query icon.
2. In the Query Catalog window, click on the desired name from the user catalog.
3. Click on any other column names you wish to include and copy them to the Query window.
4. Click on the **Show Query** button in the header of the Query Control window.

Figure 8 on page 16 shows how the Query windows appear after you click on the **Show Query** button.

© Copyright IBM Corp. 1996

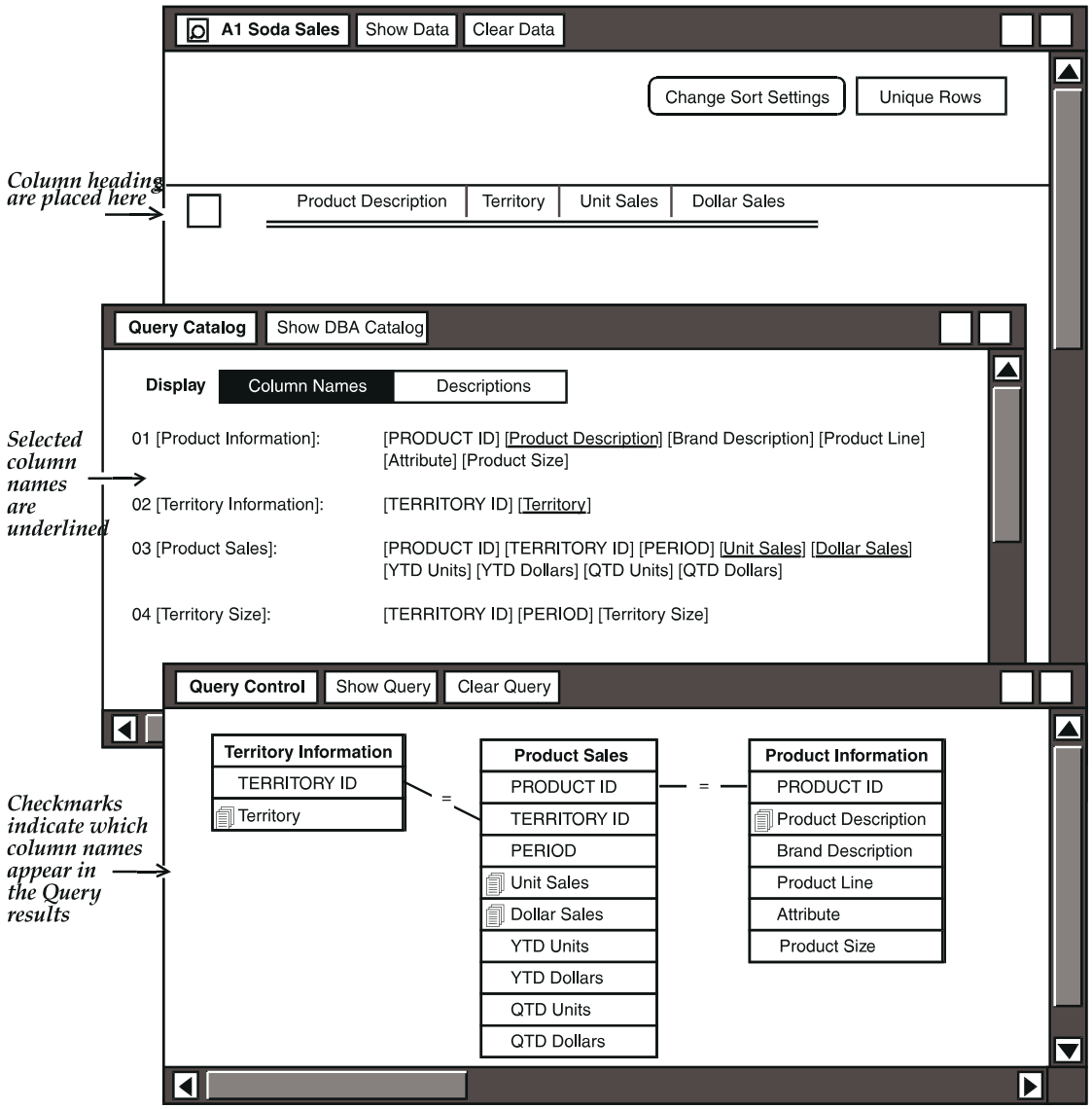


Figure 8. Query windows after clicking on the Show Query button

Look at the Query Control window again. Within the tables, each individual selection is checkmarked, and selections in the user catalog are underlined. Column headings appear in the results area of the Query window. The only thing left to do is retrieve your data by clicking on the **Show Data** button in the Query window header. But first, the following section describes more specific ways of getting column information into the Query window's results area.

Retrieving One Table

There are two ways to retrieve data for one table of information. You can either copy the table name from the user catalog directly to the results area of the Query window, then click on the **Show Data** button, or you can select the table in the user catalog and select the Query window header. Both methods are described below.

To copy the table name into the Query window:

1. Select the table name and copy it into the results area of the Query window. The columns appear in the same order as the one in the user catalog.
2. Select the Query window header to retrieve data from the table.

You should only do this with small tables or when you really need all the information contained in a large one. Otherwise, such a query could take a long time to return data. In the example in Figure 9, the table only contains two columns.

To get the same results by retrieving data in the Query window:

1. In the Query Catalog window, select the table name in the user catalog.

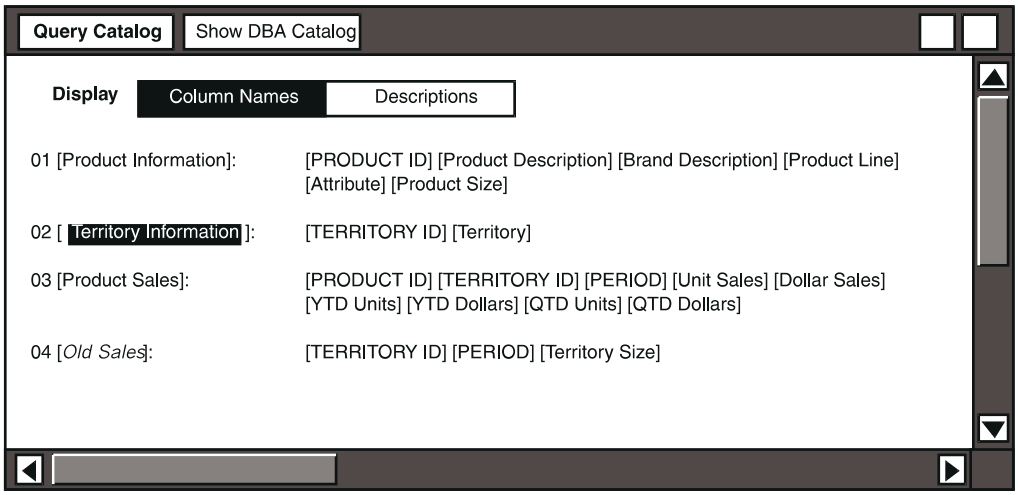


Figure 9. Retrieving a whole table from the user catalog

2. Select the Query window header.

Both selection methods provide the same results. Regardless of the method you use, selected entries are automatically duplicated in the Query Control window, resulting in a copied table like the one shown in Figure 10 on page 18. The order in which the columns are displayed is determined by the order in which you selected them. Notice that column names are checkmarked to show that they have been included in the results area of the Query window. The same names are also underlined in the user catalog as if you had selected them individually.

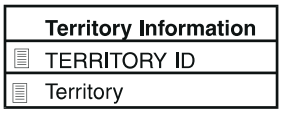


Figure 10. Table as it appears in the Query Control window

Retrieving Several Columns

You can copy several columns to the Query window, either by copying them one at a time or by copying several simultaneously. Both methods are described below.

Copying Columns One at a Time

The advantage of copying columns one at a time is that you can specify in what order the headings appear as you go along.

To copy columns one at a time:

1. Select the first column name in the user catalog.
2. Copy it into the results area of the Query window. The first name appears in the leftmost position.
3. Repeat steps 1 and 2 for each additional column.

This method permits you to change the order of the columns, after you have placed the first two. For example, suppose you placed two columns in the Query window, and decide that you want to insert a column between the first and second one. To insert a column between two others:

1. Select the new column name in the user catalog.
2. Copy it directly on top of the first column in the Query window.

   The new column appears in the second position from the left.

You cannot use the copy method, described in the previous set of steps, to replace the column in the first or leftmost position. To insert a column immediately to the right of the **Select Al** button, you must duplicate the leftmost column, insert the column you wish to replace it with between the duplicate columns, then delete the first duplication.

Copying Several Columns Together

Copying several columns together is more efficient but requires you to specify the columns in the order that you want them to appear.

To copy several columns together:

1. Select the first column name in the user catalog.
2. Extend to the next column that you want to include in your query, and copy the column or columns into the results area of the Query window.

The column headings appear in the window in the order that you selected them in the user catalog. The selected column names are then underlined in the user catalog portion of the Query Control window.

The table from which columns are selected appears in the Query Control window. The selected column names are checkmarked in the listing to show that they are included in the Query window, as shown previously in Figure 10 on page 18.

You can delete items that appear in the Query depiction, but you cannot move them around in the window. When you click on the **Show Query** button, the tables are often rearranged by the tool so that all your selections are visible.

Joining Tables

In a database, the column names are often duplicated in several of the tables. This allows the system to match columns of the same data type. The ability to store and maintain data in separate tables, and to link those tables during data retrieval, is one of the most important features of a relational database system.

Common items can have different column names in the different tables (for example, **div number**, **division #**). However, they are both numerical values representing the same item. Figure 11 shows an example.

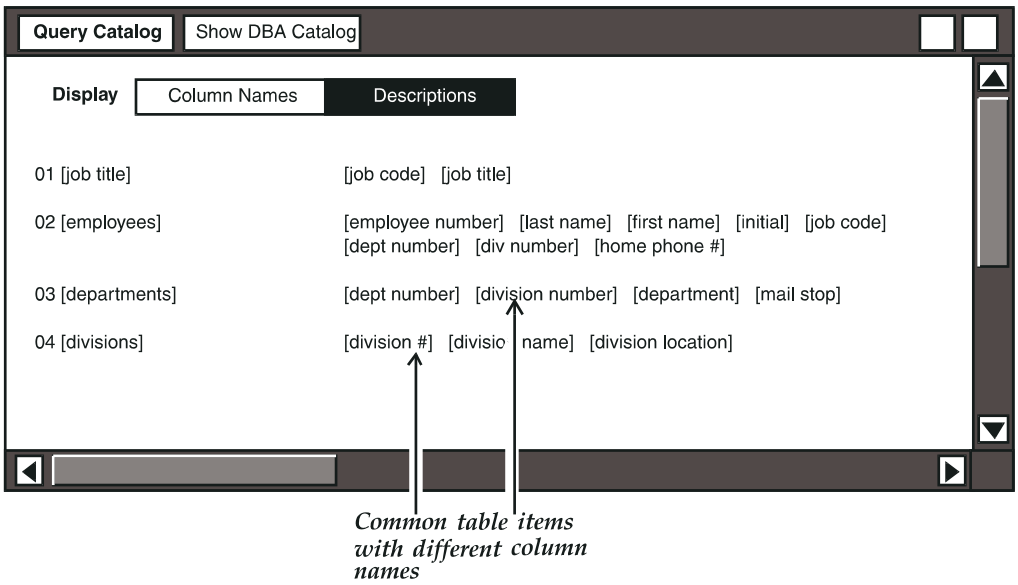


Figure 11. User catalog showing common items with different names

The query can automatically match up an employee's last name with the name of the division she works in by matching them through the division number contained in both tables.

Two types of joins are used in Query: default joins and manual joins. Both kinds of joins are described in the sections that follow.

Understanding Default Joins

When you select Query Control window header, a line appears that connects the two tables through the data column. This line usually contains an equal sign (=) and is called a *default join*. Using the database category shown in Figure 11 on page 19 as an example, the line represents the path that the system takes to find the department name that is associated with the last name. The Query tool creates the default joins between tables as follows:

1. The system finds the **last name** and the **dept number** in the employees table.
2. It matches the **dept number** with the same **dept number** in the departments table.
3. It then retrieves the **dept name** that matches that number.

The specification for default joins is usually set by the database administrator or someone who has access to the DBA tool. For more information about using the DBA tool to create default joins in a database, see *DBA Tool User's Guide*.

Creating Manual Joins

Depending on the design of your particular database, there are some cases when you might need to create a manual join. For example, the database can be set up so that a default join does not occur between certain tables. Then you must create the join manually in the Query Control window.

To create a manual join:

1. Select the appropriate column name in the first table.
2. Extend to the matching column name in the second table. The join line is drawn between the two column names.

You might also need to create a manual join to improve the performance of the query. Creating manual joins improves the performance of the query because it eliminates the time needed to search a table for the related column name. Manual joins are also required when you are using two or more views (a predefined query) in a single query. For more information, see Chapter 7, “Using Query Views” on page 49.

For example, a database might be set up so that a default join between a departments table and a divisions table occurs through the employees table as shown in Figure 12 on page 21.

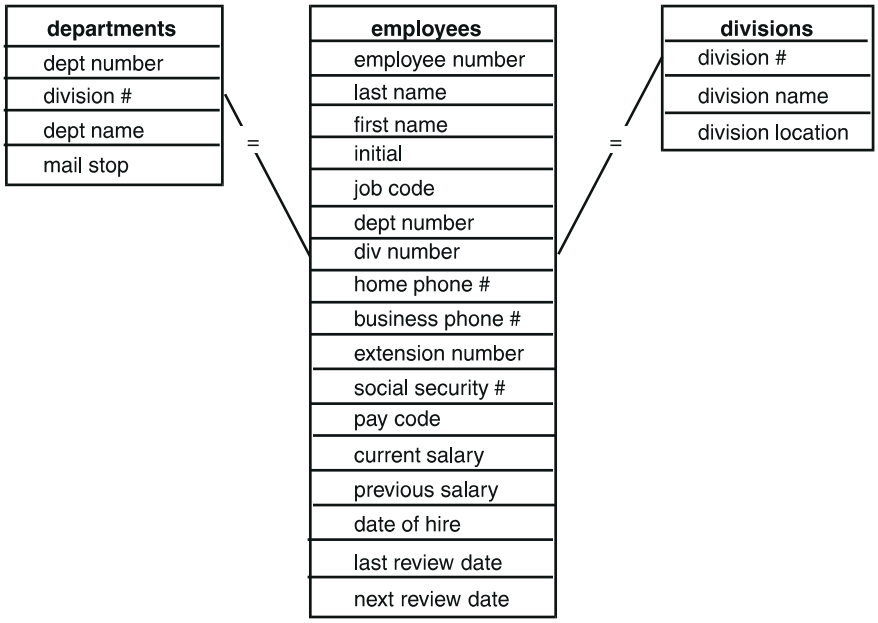


Figure 12. Default joins between three tables

If you do not need the employees table information in your query results, you can delete that table from the Query depiction, then manually connect the remaining tables through the **division number** column using the right and left mouse buttons.

To delete an unwanted table and create a manual join:

1. In the Query depiction, delete the heading of the unwanted table.

   If you have already clicked on the **Show Data** button in the Query window, you cannot delete the table from the Query depiction. You must first click on the **Clear Data** button, then individually delete any column headings in the Query window's results area that are associated with that table.

2. In the departments table, select the column name you want, and extend to the column name you want to join it to in the divisions table. A join appears between the two column names, like the one shown in Figure 13.

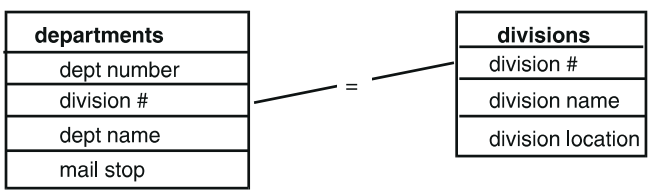


Figure 13. Manual join after deleting unwanted table

If you make a mistake or need to delete a join:

1. Select the join line. Both ends of the line become highlighted.
2. Delete the join by pressing the Delete key or by using the pop-up function pad.

Chapter 4. Constraining Data

The Query tool permits you to limit retrieved data by applying *constraints*. A constraint is a technique used to retrieve data that meets a certain criteria. For example, you might wish to retrieve all data about a single product or all data about that product during a specific time period.

You constrain your data to specific values in the selected columns by using a Comparison Options window associated with each column name. You can restrict the range for any column of data whether or not it is to be displayed in the Query window. This chapter discusses the following:

- Understanding constraint types
- Applying constraints
- Specifying the constraint source
- Changing or removing a constraint
- Using more than one constraint
- Setting options in joins

Constraint Types

The Query tool provides a variety of constraint options for making your data more specific. You can:

- Constrain data to a single value
- Set constraints to multiple values
- Use wild cards to search for matching pattern strings
- Constrain using date arithmetic expressions
- Insert user-defined variables

Constraints appear in the depiction area attached to the column name you wish to limit. The first item in the constraint string is always the operator, and the default operator is an equal sign. This tells the system to retrieve all data for the item that is equal to the constraint you have chosen. In “Applying Constraints” on page 27, specific procedures for limiting your query are described.

Constraint types are described in more detail in subsequent chapters; this section is intended to give you a brief introduction to each one, and show you how each constraint type appears in the depiction area. A more detailed description of constraint operators appears in “Selecting an Operator” on page 30. Figure 14 on page 24 shows a query depiction with some sample constraint types.

© Copyright IBM Corp. 1996

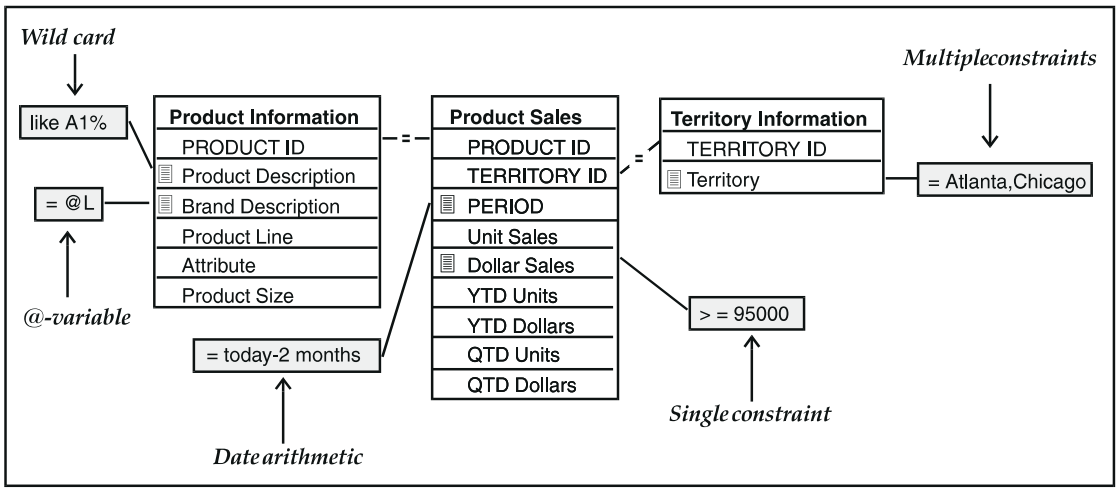


Figure 14. Constraints types

Single Values

The simplest way to retrieve data is to set up the Query to retrieve all data of a certain kind. For example, you can request all information about one particular product, such as A1 Soda, by applying that constraint to **Product Description**. Figure 15 shows a constraint that specifies a single value. This type of constraint is described in "Applying Constraints" on page 27.

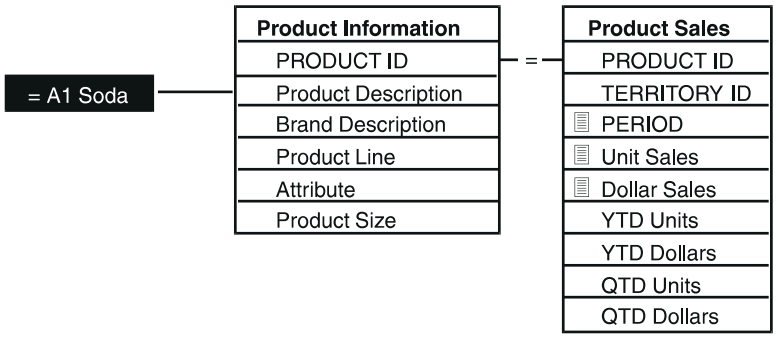


Figure 15. Constraining to a single value

Multiple Values

You can obtain information about several products by putting multiple values in a constraint. For example, you could set the constraint on **Product Description** to retrieve information about A1 Soda, Diet Soda, and Diet Seltzer.

Figure 16 on page 25 shows an example of a multiple constraint on **Product Description**. Notice that you indicate multiple constraints by including separating commas. If you type the constraints, you must also type these commas. However, if you select constraints from a list of displayed values, the system inserts commas for you. The ability to select constraints from a list of choices is described in “Applying Constraints” on page 27.

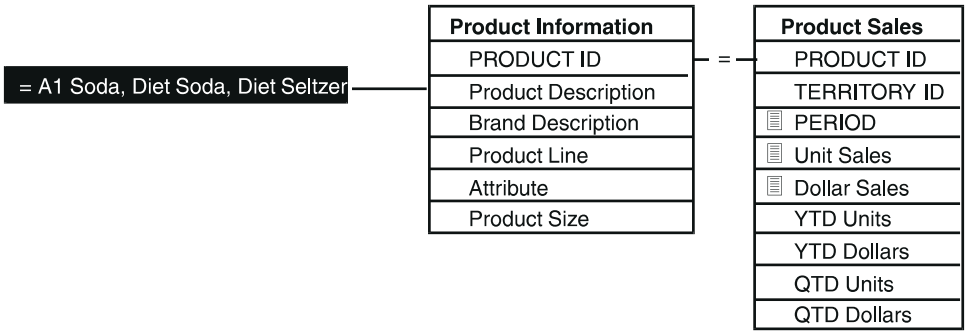


Figure 16. Constraining to multiple values

Wild Cards

Structured Query Language permits the use of wild card characters to search for items that match a particular character pattern. The Query tool permits you to use two wild card characters: the percent sign (%) and the underscore (_). The percent sign permits you to search for a sequence of characters. The underscore matches a single character.

To search for all products in the database that begin with the letters A1 and end with the word Soda, constrain **Product Description** to A1%Soda. Use the underscore when you want to search for any single character in a value.

A wild card character can be used anywhere in the character string. For example, constraining **Product Description** to **%Diet** would retrieve information for all products whose names end with Diet.

You can also use the percent character to specify a range of values, which is often a useful shortcut to typing lengthy names. However, if you combine the wild card character with the **Between** operator to specify a constraint, be careful to ask for the exact data you need. For example, if you use the wild card character to specify a range between *A%* and *D%*, the query returns only data names that begin with the letters **A**, **B** and **C**. In other words, combining the wild card character with the **Between** operator tells the Query tool to retrieve data for only those items whose names begin with **A**, **B**, and **C**, and not for any items named **D**. If you were constraining territories, your data would then omit any information about Dallas or Denver. To retrieve information up to and including all territories beginning with the letter **D**, you would need to constrain between *A%* and *E%*. See “Using More than One Constraint” on page 36 for more information about using the **Between** operator.

Date Arithmetic

Instead of just specifying a single date or range of dates, you can perform date arithmetic when setting constraints. Since the Query tool stores the current date, it can recognize the following word combinations in constraints:

today

this week

this month

this quarter

this year

With this capability, you can constrain **PERIOD** to **this month**, and reuse the query indefinitely, since it would always retrieve data for just the current month. You could also add an operator, and write constraints that tell the query to retrieve information for today - 1 month. The tool evaluates an expression in a constraint date. Table 5 shows how date arithmetic expressions work if the month is August, 1994.

Table 5. Date arithmetic expressions

Constraint date	Results
this month - 1 month	July, 1994
this month - 6 months	February, 1994
>this month-6 months	February, 1994 onward
between this month- 1 year and this month	August, 1993 through August, 1994

@-Variables

User-defined variables can be employed across applications for purposes of convenience and efficiency. In the Intelligent Decision Server, **@-variables** are used as a shortcut to typing and to make the data more portable to other tools. Specifically, @-variables are used in tools that you use inside capsule application windows where they take values from the capsule application's control window.

For example, if you know that the variable **@L** always stands for **Brand Description** in your capsule application, you can use this as a typing shortcut. Additionally, you can use the same @-variable many times over, but assign different values each time. If **@A** is always defined as **Territory**, and you assign it the value **Buffalo**, then assign the value **Diet Soda** to **@L**, running the query tells you how well that particular brand is selling in Buffalo. You can then change the value of **@L** in the capsule application's control window, perhaps so that it retrieves Tropical Nectar information, and run the query again. This time you get all information available for the brand Tropical Nectar in the territory **Buffalo**.

You can combine @-variables in date arithmetic constraints, such as specifying **@F - @G weeks**.

Be sure to add a space between the value and the first letter of the date arithmetic expression, otherwise a single-letter @-variable will be interpreted as two-letter variable. For example, **@Gweeks** would be interpreted as **@GW** followed by the word **eeks**.

Figure 17 shows a depiction area with two constraints containing @-variables where **@L** specifies a brand description, and **@A** tells the query to look for all information about a specific territory.

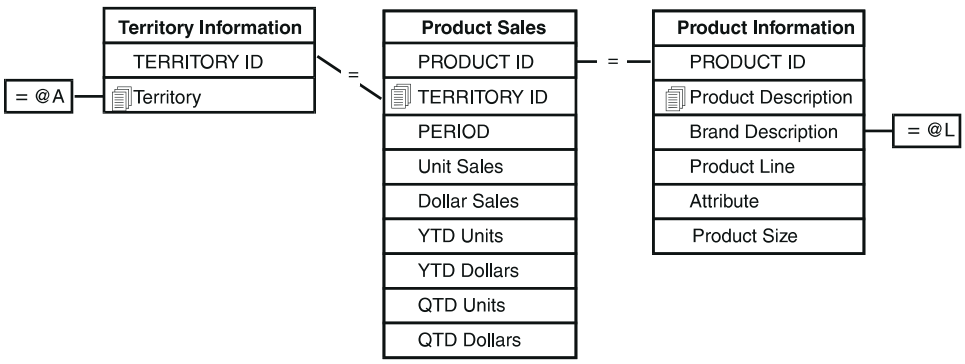


Figure 17. Constraints using @-variables

The values for @-variables are generally specified in the capsule tool's User Input Control window. You might also wish to check with your database administrator for a list of @-variable values. For more information about how @-variables can be used across tools, see *Capsule User's Guide*.

Applying Constraints

Before you can apply any constraints, you need to open the options window for the item you wish to constrain. To open the options window, select a column name in the depiction area and display its options window.

Figure 18 on page 28 shows a Comparison Options window. The name of the selected column appears in the **Label** field; in this example, the column name is **Territory**.

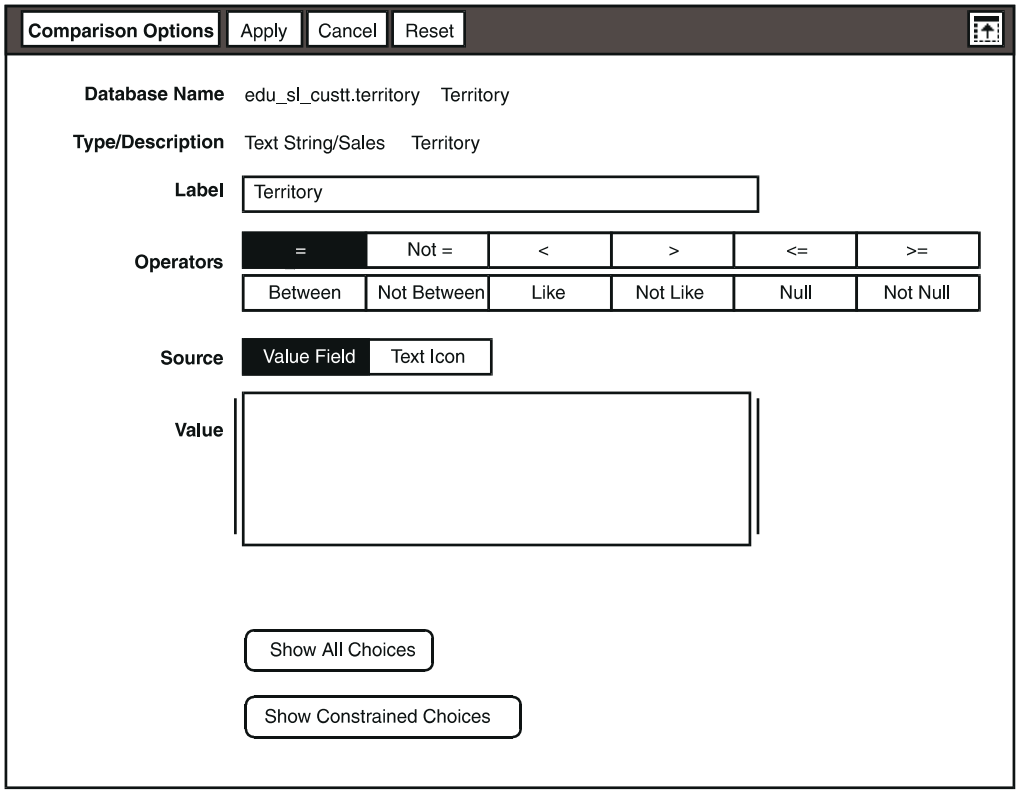


Figure 18. Comparison Options window

To constrain **Territory**, you enter a value for the selected column and then close the Query Control window. Notice that a small rectangle is now attached to the column name **Territory**. The rectangle contains the constraint value you entered in the **Value** field. The constraint frame remains highlighted until you deselect it. Notice that the default selection in the **Operator** field, the equal sign, also appears in the constraint.

After you specify the value, the constraint is attached to the specified item wherever it fits most conveniently in the available space.

Specifying the Constraint Source

When you open the Comparison Options window, you can choose to set constraints equal to or like those in a value field, or in a Text icon. The default choice is to use the **Value** field. If you are entering values just once in a specific Query icon, or if you are entering just a few values, it is easier to use the **Value** field.

However, if you are entering many values or if you want to save the values that you use frequently, select a Text icon as your source.

The Text icon should be placed in the same container, capsule application, or folder, as the Query icon. If the Query icon is on the desktop, place the Text icon on the desktop or enter the entire path name for the Text icon in the **Text Icon Name** field. When you use a Text icon as the source, the search strategy for the source icon is:

1. The container where the Query icon resides is searched first.
2. Next the desktop is searched.
3. Finally, the search continues at the System|File Drawers level

If the Query icon that uses the values is located in a capsule application, the path name might look like:

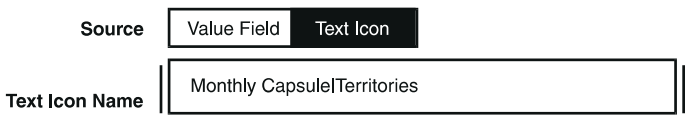


Figure 19. Constraint operators

To create a Text icon that can be used as a source for constraint values:

1. Copy a blank Text icon to your desktop or into the container where your Query icon resides.
2. Open the Text icon and type the constraint values for the column, separated by commas. For example:

   Los Angeles, San Francisco, Seattle

3. Close the icon and enter a unique name in the **Text Icon Name** field.

The sections that follow provide detailed information about the other fields and buttons in the Comparison Options window.

Working in the Comparison Options Window

The first two fields in the Comparison Options window are read-only fields. These fields identify the column name in two ways. The *Internal Name* is the unique internal name of the table and its associated column. The database server only understands internal names, and not the user-specified label.

The **Type/Description** field provides the type of data presented in the column (for example, date, text string, amount) and a description of the column which corresponds to the column name. Even if you change the name in the **Label** field, this name remains unchanged. It is uniquely defined by the column name you selected before opening the Comparison Options window.

Renaming the Column

The **Label** field contains the column name as the default when you first open the window. You can delete the default name and enter one of your own choosing. You can enter a name consisting of up to 30 characters. If you change the name, that change also takes effect in both Query windows. For example, you might want to change the name **Territory** to something more specific, such as Query window.

Selecting an Operator

The **Operator** field appears with the equal sign highlighted as the default. You can change the operator at any time by selecting the one of your choice. Use the **Operator** field to set an operator for the constraint, then enter a value, as discussed in "Entering a Value." For example, if you select the **not =** operator, and then type Atlanta in the **Value** field, the query would return all cities within the territory, excluding Atlanta.

The **Between** operator returns a range of values, including the starting and ending values you specify.

When you use the **Like** and **Not Like** constraint operators, you can use the percent sign (%) and the underscore (_) as wild card characters. The percent sign represents multiple characters and the underscore represents a single character.

Figure 19 on page 29 lists the operators and their respective functions.

Operator	Returns data that
=	equals the value(s) entered
Not =	does not equal the value(s) entered
<	is less than the value entered
>	is greater than the value entered
<=	is less than or equal to the value entered
>=	is greater than or equal to the value entered
Between	is within the range of values entered
Not Between	is not within the range of values entered
Like	matches the string data entered in conjunction with a wild card
Not Like	does not match the string data entered in conjunction with a wild card
Null	is specified as null in the database
Not Null	is not specified as null in the database

Entering a Value

When you first open the Comparison Options window, notice that the insertion point is in the **Value** field. You are expected to enter a text variable for the operator to act on. Another way to enter a value is to click on Choices window. This feature is discussed in "Selecting Values from the Choices Window" on page 32.

Typing the Value into the Field

If you select **Between** as the operator, two value fields appear, labeled **Beginning Value** and **Ending Value** so that you can specify a range of values. For example, to constrain the column name **Last Name**, you can select **Between**, type Evans in the **Beginning Value** field, and type Leone in the **Ending Value** field. The query returns only those names that occur alphabetically between the two that you entered, assuming the values are sorted alphabetically. An exception might be the months of the year, sorted in chronological order.

Be sure to specify beyond the range of values if you use a wild card in constraints that contain a range of values.

Typing Multiple Values

You can specify multiple values by placing commas between them. When you use a comma as a thousands separator or a decimal separator, always use a comma and a space between values. Otherwise, the value is interpreted incorrectly. For example, to constrain the column name **division number**, you can select **=** and enter the values as 231, 345. The query returns only data related to those divisions and ignores any data related to the other division numbers.

The Set Preferences window lets you specify the number format for your desktop. For more information about choosing a number format, see *Working With the Intelligent Decision Server*.

Entering Date Values

Date values that you type directly are case-insensitive. The date you enter into the **Value** field can change format depending upon the settings for your particular desktop.

Table 6 shows some sample formats for entering dates directly into the field.

Table 6. Date formats

To specify	Formats
Day	January 15, 1995; Jan 15, 1995; Jan 15, 95; Jan 15 95; 15 Jan 95; 15 January, 1995
Month	January, 1995; Jan, 1995; January, 95; Jan, 95; Jan 95
Year	1995; 95
Odd-Bimonth	DJ, 1995; DJ 95
Even-Bimonth	JF, 1995; JF 95
Quarter	1Q95

While all date resolutions can be displayed, for non-numeric data types, the date resolutions **Week**, **Quad Week**, and **Qtr** do not translate correctly to date format when used as a constraint in the Query tool.

Selecting Values from the Choices Window

You can select the **Show All Choices** button near the bottom of the Comparison Options window to display all the valid choices for the **Value** field. Using this method is advantageous because it lets you select the values from a list, thereby minimizing typing errors and eliminating the need to memorize all the possible values in your database.

To select values from the Choices window:

1. Open the Comparison Options window and click on the **Show All Choices** button.
2. Select one or more of the displayed values and close the Choices window.

When you click on the **Show All Choices** button, you might see a message in the message area indicating that the request has been sent to the database. Depending upon the number of choices in that particular list, the window can take a few moments to open.

In the Choices window, values are always listed alphabetically for character data, in ascending order for numeric data, and in chronological order for dates. You can select the values from the displayed list in any order that you wish.

If you are working with Query *views*, described in Chapter 7, “Using Query Views” on page 49, you cannot see the buttons that display the Choices window. A view is a predefined query that has already been constrained. Therefore, the **Show Choices** and **Show Constrained Choices** buttons are not available.

Selecting Multiple Values

If you choose more than one value, your choices appear separated by commas and spaces. Remember that no values appear in the Choices window:

1. Select the first choice.
2. Continue to highlight choices by clicking the right mouse button. You can highlight choices in succession, or select random choices in the list.
3. Close the Comparison Options windows.

When you close the Choices window, the values you selected appear in the **Value** field in the order that the Choices window uses, which can be alphabetical, ascending or descending, or chronological, depending on the data type. If you select randomly in the Choices window, perhaps by choosing **Los Angeles** first, then **Dallas**, and then **Kansas City**, the **Value** field lists the names in the order you used to select the cities, as shown in Figure 20 on page 33.

Figure 20. Value field reflects Choices window order

Appending Additional Values

To include additional values, you can type the values in manually. You can also copy them from the Choices window by selecting one or more values and closing the window, or by copying them directly from the window.

To append a value by typing or copying it into the **Value** field:

1. Open the Comparison Options window and do one of the following:

   Type a separating comma after the existing values in the **Value** field, then type the new value.

   Display the Choices window and select the value or values you wish to add. The comma is added after the existing values.

2. Close the Comparison Options window.

Notice that the Query tool does not sort the appended value with the others in the **Value** field. The value is simply added to the list of values. Accidently selecting the same value twice cannot adversely affect your results. The Query tool does not retrieve duplicate rows of data when you are selecting choices.

Deciding When to Display All Choices

You should have a rough idea of how many choices might be available for a particular value before using the **Show All Choices** button. Even though displaying all choices provides a typing shortcut, there might be several hundred choices to display for a particular item. In such cases, it takes a long time for Query to display the list. An additional button, named **Show Constrained Choices** provides a way to avoid displaying unwanted information. This button is described in "Displaying Constrained Choices."

Interrupting the Choices Display

If you select the **Show All Choices** button and decide you do not want to wait for all the values to be returned, you can stop the search by pressing Stop. Doing so does not affect the Query data you have already specified in the Comparison Options window.

Displaying Constrained Choices

The **Show Constrained Choices** button at the bottom of the Comparison Options window lets you display a list of choices based on the constraints you have already applied to one table. This feature allows you to limit the database query to just those values that you want to see.

Using the **Show Constrained Choices** button invokes a Choices window similar to the one that appears when you click on the **Show All Choices** button. However, if you have already applied constraints, the Choices window might display fewer choices, and therefore, values are easier to locate.

Suppose you want to retrieve information about a product named Sparkle-Slim Soda, but you only want to find out about which product sizes sell best. Using this example, you might build a query as follows:

1. Select **Product Description** in the **Product Information** table in the depiction area.
2. Constrain **Product Description** to just the product varieties you want to know about. For example, you might choose two popular soda flavors, such as **Lemon-Lime** and **Cherry.**

After you have built the query and applied constraints, you can look at a list of constrained choices:

1. Select **Product Size** in the depiction area.
2. Open the Comparison Options window.
3. Click on the **Show Constrained Choices** button to display a list of available product sizes for the two soda flavors **Lemon-Lime** and **Cherry**.

Instead of a complete list of all the products that exist in the **Product Information** table, the Choices window displays a list of values that are more pertinent to your needs.

If you click on the **Show Constrained Choices** button before applying any constraints, the Choices window that appears lists all available choices just as if you had clicked on the **Show All Choices** button.

You can copy the constrained values into the Comparison Options window just as you would from the list that appears when you select the Choices window for constrained choices by selecting its title block, or by closing the Comparison Options window.

Changing or Removing a Constraint

After you have closed the Comparison Options window, a constraint containing the value you entered appears for a single column name in the depiction area. A line connects the constraint to the column name box. After clicking on the **Show Data** button in the Query window, you might wish to run the same query again, but with a different constraint. For example, you might decide to change the constraint on the column name **Territory** from **Buffalo** to **Philadelphia**.

To change an existing constraint:

1. Select the constraint frame so that it becomes highlighted, as shown in Figure 21.
2. Open the Comparison Options window using the pop-up function pad or by pressing the Options key.

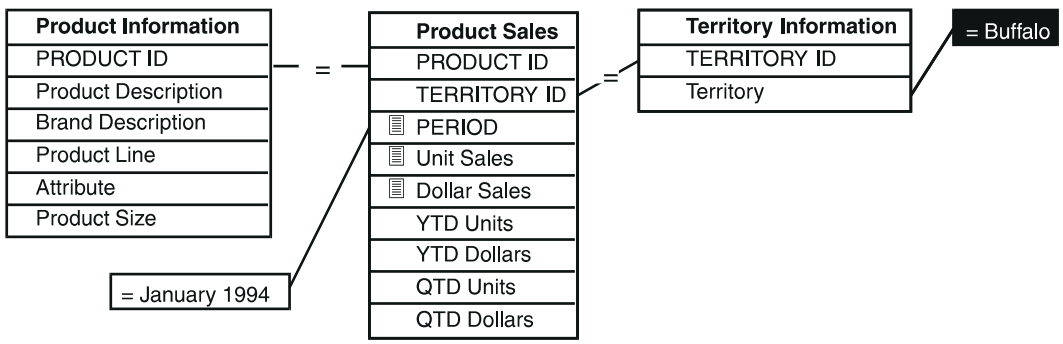


Figure 21. Changing the constraint

The Comparison Options window opens, just as if you had selected a column name. You can remove, change, or add values to the **Value** field, and you can change the operator. Constraints can be removed by simply selecting and deleting them. Each time you make a change, then close the window, the new changes are reflected in the depiction area.

Using More than One Constraint

Occasionally, you might want to add a separate constraint on the same column name. For example, suppose you want to constrain **Territory** to all cities listed alphabetically between **Chicago** and **Los Angeles**, but also include **Philadelphia**. You can do this by selecting individual city names and putting them in one constraint, then selecting the next territory and adding it as a second constraint.

First, you specify the range of choices as follows:

1. In the depiction area, select the column name **Territory**.
2. Open the Comparison Options window.
3. In the **Operator** field, select **Between**. Two value fields appear.
4. Type a value, such as Chicago, in the **Beginning Value** field.
5. Select inside the **Ending Value** field (to move the insertion point), and type a second value, such as Dallas.
6. Close the Comparison Options window by selecting its title block.

The constraint now appears as shown in Figure 22. Notice that the range of values is shown with dots between the two names in the constraint.

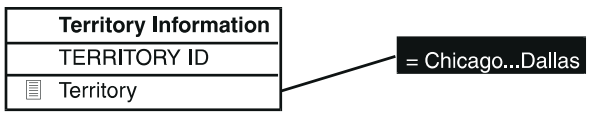


Figure 22. Appearance of the constraint

If you combine a wild card with the **Between** operator in a constraint, remember that the placement of the percent sign delineates the items you are requesting. Therefore, you need to constrain the query to retrieve all data between **C%** and **M%**, not just **C%** and **L%**. Doing so ensures that you get data for **Los Angeles**.

Now you can apply a second constraint to include one of the values that falls outside the range you specified.

To apply the second constraint:

1. Select the column name **Territory**. Do not select the constraint on **Territory**.
2. Open the Comparison Options window and select the **=** operator.
3. Type a value, such as Philadelphia, in the **Value** field.
4. Close the Comparison Options window.

Your depiction area now has two constraints attached to **Territory**. Notice that the constraint frames have been rearranged by the system to allow the second constraint to fit.

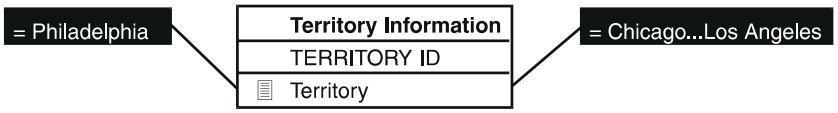


Figure 23. Adding a second constraint

These two constraints return territories that include **Chicago** and **Dallas**, and also **Los Angeles**. Other territories are excluded from retrieved data.

Setting Options in Joins

You can use the Comparison Options window to constrain a join between tables. Suppose you need to specify a relationship between columns in different tables. For example, you might have a column named $ sales in one table and a column named forecast $ sales in another table. You first create a join between these two columns. Then open a Comparison Options window, and select the < operator to depict only those cases where the total for $ sales is less than forecast $ sales.

To open this type of Comparison Options window:

In the depiction area, select the join connecting the two column names that you want to compare, and display its options window. Figure 24 shows the Comparison Options window for joined columns.

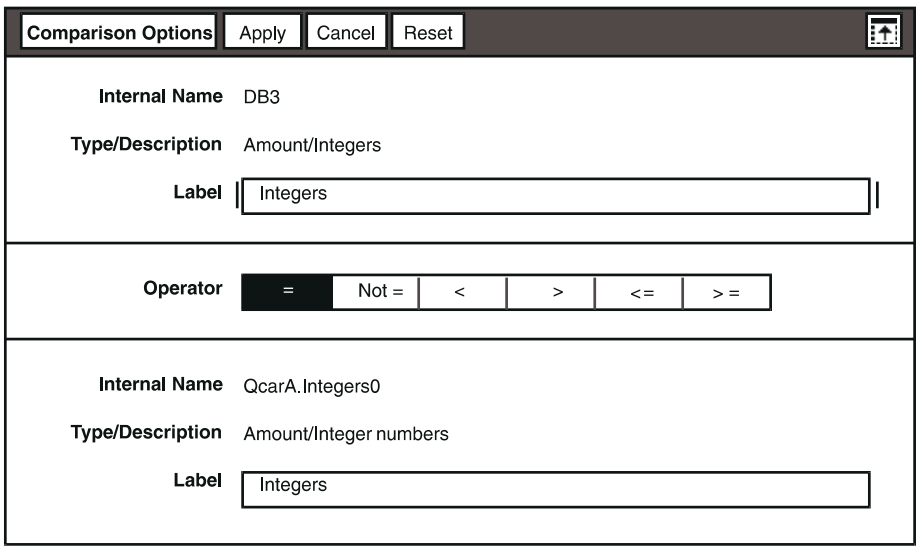


Figure 24. Comparison Options window for joined column names

The Comparison Options window for a join is similar to the one you use to constrain a single column name, except this window is divided into three sections.

The first section gives the **Internal Name** and the **Type/Description** for the leftmost column name and first **Label** field.

The second section allows selection of an operator. In this window, the **Between** operator is not included. (Notice that there are no **Value** fields.)

The third section is identical to the first section, except that its contents apply to the rightmost column name.

If you created a join between two tables, as suggested in the first part of this section, then select the < operator to indicate where the total for $ sales is less than forecast $ sales, the join line in the depiction area would appear with the < symbol instead of the more common equal sign. Figure 25 shows an example.

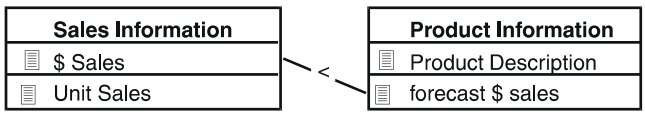


Figure 25. Example of a constrained join between tables

You can easily change join options using the same basic steps you used to set them. For example, if you wished to make a comparison between the **$Sales** and **forecast $ sales** shown in Figure 25, you simply select the current join, and delete it and then use the right and left mouse buttons to create the desired join. However, the easiest way to change an operator in a join is to keep the join intact, and select a new operator in the Comparison Options window. The following steps explain how to change an operator in a join.

To change the operator in a join:

1. Select the join and display its options window.
2. In the Comparison Options window for joins, select a new operator from the **Operator** choices.
3. Close the Comparison Options window. The new operator appears in the join line in the depiction area.

For more information about joins, see *DBA Tool User's Guide*.

Chapter 5. Editing a Query

This chapter shows you how to rearrange retrieved data to make it easier to work with. The following topics are covered:

Rearranging query data

Changing a column heading

Using a different database category

Rearranging Query Data

The two buttons in the upper section of the Query window allow you to arrange the data in your query in an order that makes analysis easier: the **Change Sort Settings** button and the **Unique Rows** button.

Sorting Data

The retrieved data in the columns is arranged in the same sequence as in the database. You can change this sequence by using the **Change Sort Settings** button. Selecting this button displays the Sort Controls window.

This window allows you to set the sorting sequence for up to seven data levels. For each level, the options are:

A sequence setting to let you select ascending sorting order **(0-9, A-Z**) or descending sorting order (**Z-A, 9-0**)

Column name fields where you copy the name of the column to sort

To select the column name for sorting:

1. Select the **Change Sort Settings** button in the Query window to open the **Sort Controls** window.
2. In the Query Catalog window, select a column name from the user catalog, or select a column in one of the tables in the Query Control window. If you select from the user catalog, the selection must be one that is already included in the depiction area's tables.
3. Copy the selected column into the column field of the sorting level you wish to change.

The internal name of the selected column appears in the field. You can sort up to seven columns in one database query. In Figure 26 on page 40, a name has already been copied into the **Column** field.

© Copyright IBM Corp. 1996

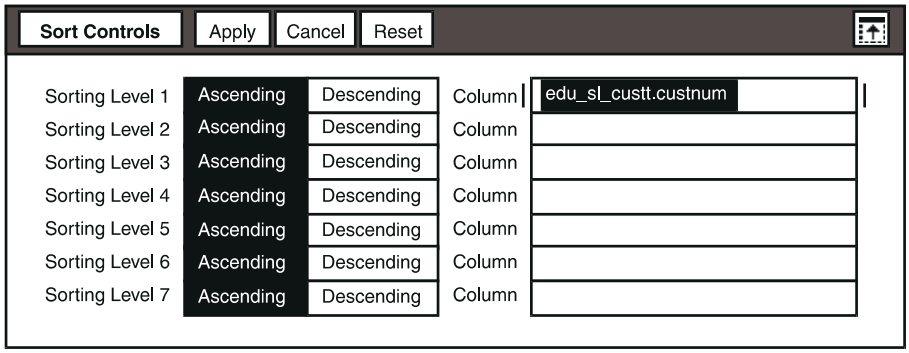


Figure 26. The Sort Controls window

As soon as you apply the new sort setting, a message appears in the system area of the Query window, just above the horizontal line, indicating that a sort is in effect.

Removing a Sorting Level

To remove a sorting level, select the column field in the Sort Controls window and delete it. If you have set sort controls for more than one column, the column at the lower level moves up to the next higher level. For example, if you delete a column name at **Sorting Level 1**, the column name at **Sorting Level 2** moves up to the **Sorting Level 1** position. You see this change only after closing and reopening the window.

When you remove a sorting level, and return to the default sorting sequence, the Sort in effect message disappears from the top portion of the Query window.

Looking at the New Sorting Order

If you have retrieved the data before clicking on the **Change Sort Settings** button, you must reselect **Show Data** for the data to appear in the new sorting order. Your changes will not be reflected by merely closing the Sort Controls window.

Eliminating Duplicate Rows

The **Unique Rows** button allows you to eliminate duplicate rows of data from the Query window's results area. Doing so helps to improve query performance, and ensures that you get only the data you have specified.

For example, suppose you retrieve data and the row for **Brand Description** is repeated three times, because there are three sizes of A1 Diet Seltzer in the database. Because each brand probably has a similar range, the Query tool must retrieve six items from the database. Since this is information you do not need, you can condense the **Brand Description** information into just one row.

When you are removing duplicate rows, be careful that you do not accidentally eliminate important data. Before clicking on the **Unique Rows** button, be sure to run the query and check the returned data to determine whether or not to use this button. You should always take this precaution before eliminating columns from the results area.

For example, if you have two people in your personnel database with the same last name, using the **Unique Rows** button can remove one instance of the name.

To eliminate unnecessary data and show only unique rows:

1. In the Query window, click on the **Unique Rows** button.
2. Click on the **Show Data** button again.

When you use this procedure, you are asking to see only the unique rows of data, and the Query tool responds by eliminating duplicate rows.

Changing a Column Heading

Occasionally, you might want to change the heading of a column in the Query window. The changed name does not affect the data that is retrieved for that column because the data still reflects the internal name that appears in the first field of the Column Options window.

To change a column heading in the Query window:

1. Select the column heading. The entire column is highlighted.
2. Open the Column Options window.
3. Delete the old column name in the **Label** field.
4. Type the new name in the **Label** field.

   The **Label** field has a 30-character limitation.
5. Close the Column Options window.

The renamed column appears in the Query Control window.

Using a Different Database Category

To access another database category, click on the **Show DBA Catalog** button in the Query Catalog window. The DBA catalog page of the Query Catalog replaces the user catalog.

Looking at the DBA Catalog

The DBA catalog appears similar to the window shown in Figure 27 on page 42.

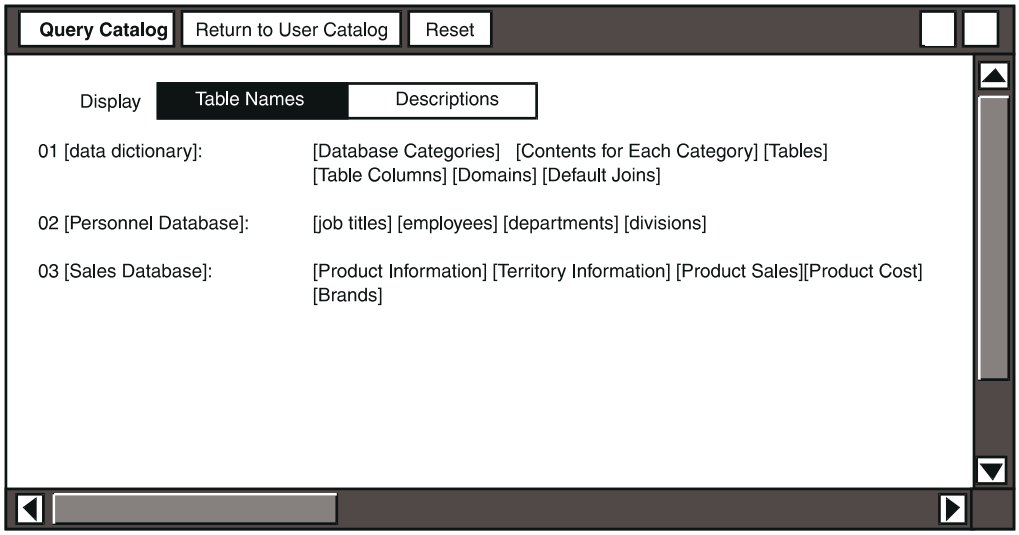


Figure 27. DBA catalog displayed in the Query Catalog window

Notice that the **Display** option now shows **Table Names** instead of **Column Names**. When the DBA catalog is displayed, you can scroll the window to see all entries in the database catalog, and you can use the right and left mouse buttons to choose from the database contents.

Selecting a New Database Category

To display the database catalog and select to a new category:

1. Click on the **Clear Query** button to clear the current selections in the user catalog and the depiction area.
2. Click on the **Show DBA Catalog** button. The DBA catalog appears, displaying all the categories available to you.
3. Choose a different database category from the catalog. You can select an entire table name, or just specify individual column names from the new category.
4. Click on the **Return to User Catalog** button to display the new items you just selected in the user catalog. To return to your original category, click on the **Reset** button.

Depending on the number of categories available, this process could take some time.

When changes have been made to the database structure, the DBA usually updates a table that informs a Query tool, as it is being opened, that changes have occurred. In this situation, you will see the following message as the Query tool updates its information:

Reading DBA Catalog from Database Server. -- Please wait.

As when you select additional database categories, this process could take some time.

Occasionally the DBA might change the database structure without updating the appropriate table. If you suspect that this situation has occurred, you can click on the **Read DBA Catalog** button from the **Special** menu of the Query Catalog window, while it is displaying the DBA catalog. The Query tool updates its information and displays the message shown above about reading the DBA Catalog.

Chapter 6. Working with Query Data

This chapter explains how to retrieve and manipulate the query data. Topics covered are:

Retrieving data from the database

Capturing the SQL for a query

Saving query data

Removing query data

Printing query data

Retrieving Data from the Database

After you have finished building your query, use the Query window to retrieve your data. The buttons in the Query window header permit you to retrieve data, clear data from the results area, and either display or hide the Query window. It is recommended that you leave the Query Control window open on your desktop.

Displaying the Query Data

After you have constructed the query, you can retrieve the information from the database:

1. Put the Query window in front of the Query Control window.
2. Click on the **Show Data** button in the window header of the Query window.

When the query is complete, the data appears in the results area.

The data is arranged in columns and rows, just like a spreadsheet. If you build your query without selecting Query Control window, clicking on the **Show Data** button causes the depiction in the Query Control window to be updated, just as if you had clicked on the **Show Query** button.

Stopping the Query

There are three methods for stopping a query after you have clicked on the **Show Data** button. To stop a query process:

Click on the **Stop Query** button in the **Special** menu of the Query window to halt the query without closing either window. This method of stopping the Query takes more time because you have to display the **Special** menu. Use it when you change your mind about retrieving the data, or when you wish to add constraints.

Press Stop to halt the query process at any time. This method is the fastest way of halting a retrieval process. You might press the Stop key when you realize that the query is trying to retrieve too many records.

Close the Query window.

© Copyright IBM Corp. 1996

All methods of stopping the process produce incomplete results. Although some data might be returned, you need to run the process again to be sure your results are correct.

Capturing the SQL for a Query

By using the **Capture SQL** item in the **Special** menu of the Query window, you can save all communications with the database server. This is a useful technique for keeping a record of your query session.

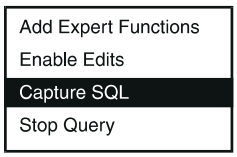


Figure 28. The Special menu

Suppose you have built a query, added constraints, and now you want to view the results.

To capture the SQL for your Query:

1. Display the **Special** menu and select **Capture SQL**.
2. Click on the **Show Data** button to begin the database query.

A window called the Query Log opens in the lower right corner of your desktop. While the Query tool accesses the database, the SQL information simultaneously appears in the Query Log window. The information in the Query Log is recorded using the internal names for the data columns you requested.

Using the Contents of the Query Log Window

The Query Log window is a capture window that provides all the same functions as the Text tool. Each time you use the **Capture SQL** item to record a query process, a separate Query Log window for that session is added to your desktop. This time-stamped window is unique to each query process.

A sample **Query Log** is shown in Figure 29 on page 46. Notice that the title block in the window header contains the current date and time, marking the exact second when the **Query Log** opened.

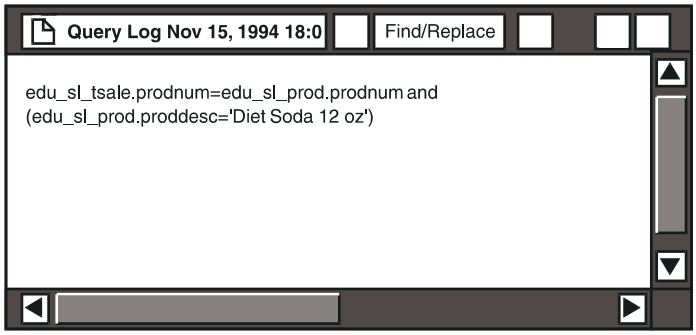


Figure 29. Query Log window

Figure 29 shows the Query Log window that appears after you select **Capture SQL** and then click on the **Show Data** button. The window's default size does not permit you to view the entire contents, but like any Text window, it can be moved, enlarged, and reformatted. To see the entire contents of the window, you might need to resize the window, then scroll up to view the beginning of the captured SQL.

You can record the SQL for multiple query processes by leaving the Query Log window open. For example, you might wish to perform a query, change a constraint, then run the same query again. The SQL information for the second query is appended to the information you captured when you ran the first query. The Query Log window continues to record the SQL for each query as long as the Query Log window and the Query window remain open on your desktop. Remember, however, that the time in the Query Log window title block always reflects the exact second when the Query Log window first opened, and does not record the start time of subsequent database queries.

Date constraints appearing in the Query Log window use the same format as the database.

You can close the Query Log independently of the Query window, or you can select the Query window's title block, and both windows close. After you close the Query Log the capture process stops. The Query Log icon remains on your desktop until you delete it.

You can expand the Query Log icon to view the date and time of the capture.

Stopping the SQL Capture

To stop the capture process, select the title block of the Query Log window. The Query Log window immediately closes, and its contents reflect whatever the capture gathered up to the second you selected the window title.

Saving Query Data

Query data is not stored in the Query icon when you close and reopen it. However, you can save query data by copying it to a Plot, Text, Layout or Spreadsheet icon. You can copy:

Any rectangular block of information (like selecting a range of cells in a spreadsheet)

Individual or multiple columns or rows

The entire results area by selecting the Query window

Removing Query Data

There are three ways of removing data from your query. You can:

Clear only the retrieved data but retain column headings and the depiction area information

Clear the entire query, including the results and depiction areas

Delete specific items from both query windows

You perform these operations by using the **Clear Data** button, the **Clear Query** button, or by deleting selected data. Each of these methods is described in the sections that follow.

Clearing Retrieved Data

Click on the Query window header to remove all data, but retain column headings and depiction area information.

Clearing the Query

Click on Query Control window header to completely erase the query depiction in the Query Control window. Clearing a Query removes retrieved data and the table and column names from the Query window.

Be careful, because clicking on the **Clear Query** button deletes your work and it cannot be retrieved. To save a copy of the query setup, save it as a view.

Deleting Parts of a Query

You can delete items that you are sure you no longer need, or even an entire query. To protect you from accidentally deleting important data, you cannot delete a column heading from the results area of the Query window if it is associated with:

Computed headings

Columns from other tables

Columns from any saved views

All Column Headings

To delete all column headings in the results area of the Query window:

1. If data is present, click on the **Clear Data** button.
2. Click on the **Select All** button in the results area and delete all the selected column headings.

Individual Column Headings

To delete individual column headings in the results area of the Query window:

1. If data is present, click on the **Clear Data** button.
2. Select one or more column headings and delete.

Deleting columns in the Query window removes the checkmarks from the column names in the Query Control window. However, you can still request a database query after you have deleted columns.

Columns or Tables in the Query Depiction

The depiction area's contents can only be deleted after the contents of the results area have been deleted, or if the table's columns are not being used in the Query window results area, that is, there are no checkmarks next to the column names.

To delete items in the query depiction:

1. In the Query Control window, select the table name and delete it.
2. Continue selecting and deleting tables as needed.

Printing Query Data

The Query icon cannot be copied to a Printer icon, so you must use other tools to print Query results. Data can be copied to a Spreadsheet, Plot, Layout or Text icon for printing.

Chapter 7. Using Query Views

A view is a predefined database query that you can list in the user catalog. Saving a query view allows you to capture a particular query, assign a meaningful name to it, and use it later. Saved views can be used many times to retrieve the same type of query data. Because you are not working with the whole range of database tables, selecting a view to work with provides a convenient shortcut. You can also use a view as a template, modify it by adding different constraints, then save it under a different name. Views also allow you to perform some calculations without using spreadsheets. This chapter provides information about creating, naming, and using views.

Creating a View

When you have a particular query that you know you will need later, you can save this query as a view. This feature lets you set up specialized query templates, that is, queries that look at specific, predefined items in the database. For example, if you wish to look at just the sales information for one brand of product, you can set this constraint in your view, and use the saved view every time you need it. The Query Control window lets you create and name reusable query views.

To create a view:

1. Set up your database query and test it. If you are sure it retrieves the data you want, proceed to the next step.
2. Working in the Control window, select **Save As View** from the **Special** menu. The New View Options window opens.
3. Enter a name for the view in the **Label** field. The name should be associated with the specific data you are focusing on.
4. Select in the **Description** field and add a comment describing the contents of that view. This step is optional, but recommended. Entries in the **Description** field can be useful to others who might use the database query.
5. Close the New View Options window.

When you close the New View Options window, a message appears in the message area of the Query window informing you that the new view has been added to the end of the user catalog in the Query Catalog window. The name of the new view appears in italics at the end of the list of categories in the user catalog. If the user catalog is a large one, you might have to scroll that section of the window to see the last item.

If you added comments in the **Description** field, you can look at that information by clicking on the **Descriptions** button in the **Display** option, as shown in Figure 30 on page 50.

© Copyright IBM Corp. 1996

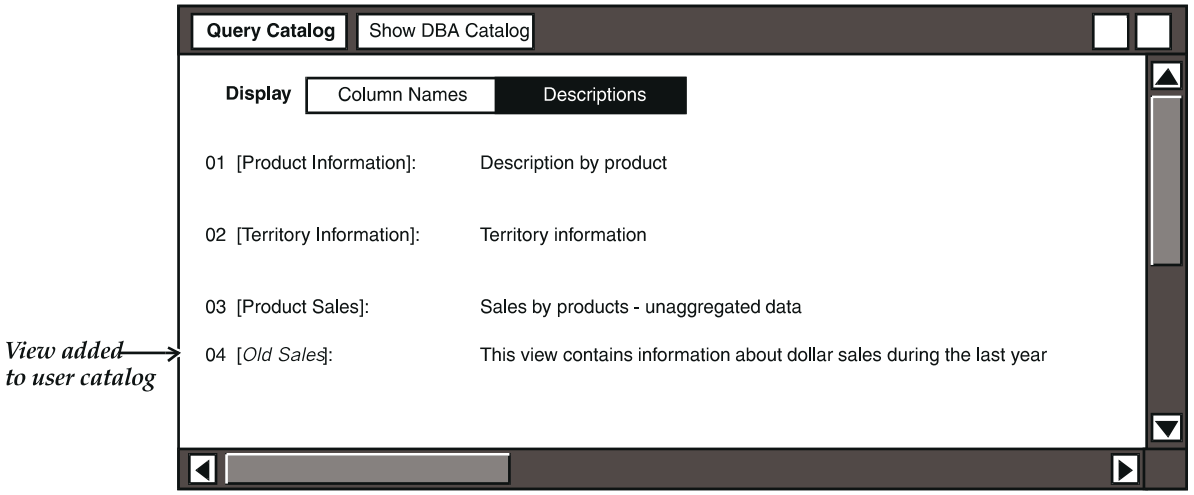


Figure 30. A saved view and its description

Remember that data retrieved for the view is not saved with the view. Only the table and column names are saved. It is also important to remember that if you previously changed the sort settings, or chose to display only unique rows, those changes are not saved with the view.

Changing a View

You might want to use a view to build and test a new database query setup. Or you might wish to simply change the query specifications of an existing view. To change a view:

1. Select the name of the view in the user catalog.
2. Click on **Show View** in the **Special** menu of the Query Catalog window. The saved view appears in the depiction area, and in the Query window. At this point, you can choose to perform any of the following modifications:

   Add a column by selecting an item in one of the tables in the depiction area, and copying it into the results area.

   Delete a column.

   Change a constraint by selecting the constraint in the depiction area and editing the constraint in the Comparison Options window.
3. Select **Save As View** in the **Special** menu of the Query Control window. The New View Options window appears with the **Label** field highlighted.
4. Close the New View Options window without changing anything in the **Label** field.

After you complete step 4, a message appears in the message area telling you that the old view has been replaced with the new one. The name still appears in the user catalog, but any changes you made to that view are reflected in the Query window.

If you want to keep the original version of the view, you must enter a new name for the new view in the New View Options window. You then have two views; the original, and the new one with the new name. The Query tool prevents users from creating two views with the same name.

When changing a view, you cannot display constraint choices by clicking on Comparison Options window. These buttons are not available when you are working in a view.

To delete a view, select the view in the user catalog and delete it.

Using Multiple Views

You can use multiple views in a database query. You might do this if you are building and testing a query to see if it retrieves the exact data you need. Or, suppose you want to look at how well a particular product sells during the holiday season, and perhaps compare two or more holiday sales periods side-by-side. There are three possible ways of performing this kind of analysis.

Use the Reporter tool to retrieve all the data and display it in a report. See *Reporter User's Guide* for more information on how to do this.

Create a query for each holiday period, then combine the data in a spreadsheet. See *Spreadsheet User's Guide* for more information about performing analysis with a spreadsheet.

Create two views in a single query. Instructions for doing this are provided in the next section.

Creating Two Views

Suppose you want to look at the sales for certain kinds of soda during the months of November and December. You would need to create a query that looks at A1 Soda, and constrain the period to between November, 1994 and December, 1995. Now save that query as a view named Current Sales.

In the depiction area, change the constraint on period to between November, 1994 and December, 1995. Save it again. In the New View Options window, save the new view as Old Sales. Figure 31 on page 52 shows you what the user catalog should look like.

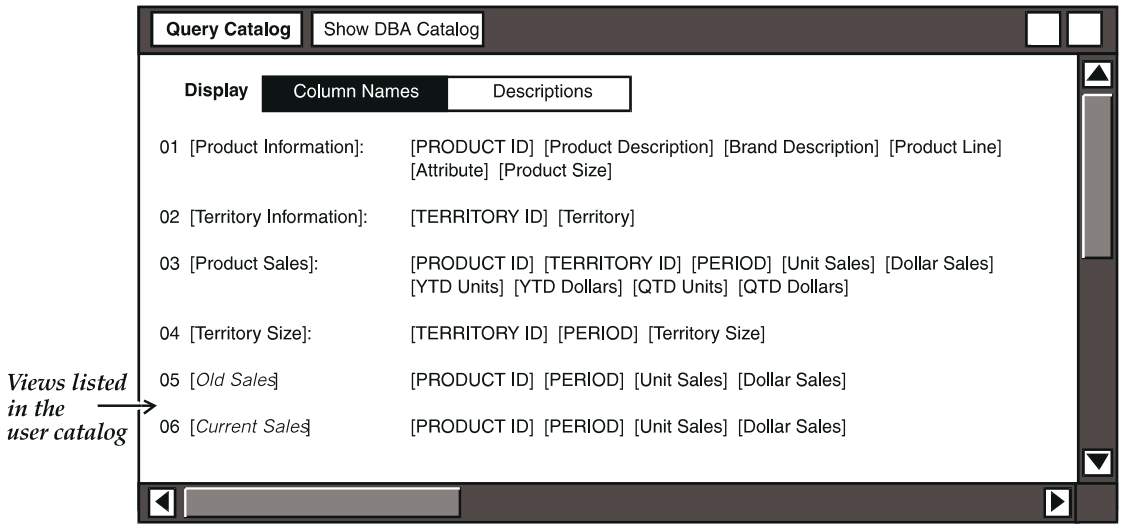


Figure 31. The user catalog with two saved views

Important: If you are using multiple views in a query, perhaps to test data retrieval, or to create different versions of a similar query, be sure to save each iteration as you make changes. You should also be aware that, while building a query depiction, clicking on the **Show View** button deletes the depiction that you have just created. You can compare two views in the same depiction, but you cannot display a view and still retain an unsaved query depiction.

Comparing Views

To look at both views side-by-side using the same views shown in Figure 31:

1. Scroll the user catalog upward so that you can see both views clearly.
2. Select the view named **Old Sales**, and copy it into the results area of the Query window.
3. Select the view named **Current Sales** and copy it into the results area of the Query window.

   Both sets of columns appear in the results area side-by-side. In the Query Control window's depiction area. The two views appear as separate tables with no joins between them.
4. To manually join the two tables, select **Territory** in the Current Sales table, and extend to **Territory** in the Old Sales table.

Performing Calculations with Views

The Query tool's ability to perform calculations is limited by the inherent structure of relational databases. However, some of these limitations can be overcome by using different views of a query. You can do this by creating a query with the basic computed

column, then saving this query as a view so that you can perform further calculations on the data. In effect, you are setting up a query with computed columns whose formula contains other computed columns.

To use views for performing calculations:

1. Create a view with the first set of calculations.
2. Save the view to the user catalog.
3. Display the view in the depiction area, and create a computed column to perform the final calculation.

For more information on performing calculations, see Chapter 8, “Performing Calculations” on page 54.

Chapter 8. Performing Calculations

You can perform a variety of calculations using the Query tool. This chapter describes:

Creating a computed column

Changing the calculation

Creating a Computed Column

Many of the steps below use techniques introduced in previous chapters. You already know how to select data from the database and constrain it, so those procedures are not described in detail. The steps for creating a computed column are provided in two parts. First you build the query, and then you must set up the calculation.

Building the Query

First, set up your query:

1. Construct the query to retrieve only the non-calculated columns.
2. Constrain the tables for all the values you want to use for your calculations.
3. Select Query window header (if you have retrieved data after constructing the query).
4. Delete any columns in the results area that you do not want to see in your results.
5. Select **Add Expert Functions** in the Query window's **Special** menu. When you add expert functions, two new fields appear in the depiction area of the Query Control window. Chapter 9, “Using Expert Functions to Constrain Data” on page 60 explains how to use expert functions. A new button, the **Create Column** button, also appears in the Query window header.
6. Click on the **Create Column** button that appears in the Query window header. The Computed Column Options window opens.

Understanding the Computed Columns Options Window

When the Computed Columns Options window first appears, the **formula** operator is always highlighted, because that is the default. If you choose to enter a formula, you can leave this operator highlighted, enter your label into the **Label** field, then move the cursor to the **Formula** field, and click the left mouse button. This will cause the insertion point to move so that you can either enter your formula into that field.

If you select any of the first five operators, **sum**, **count**, **average**, **min**, or **max**, a field labeled **Grouped By** is added at the bottom of the window. Figure 32 on page 55 shows the field that appears when you select the **sum** operator.

© Copyright IBM Corp. 1996

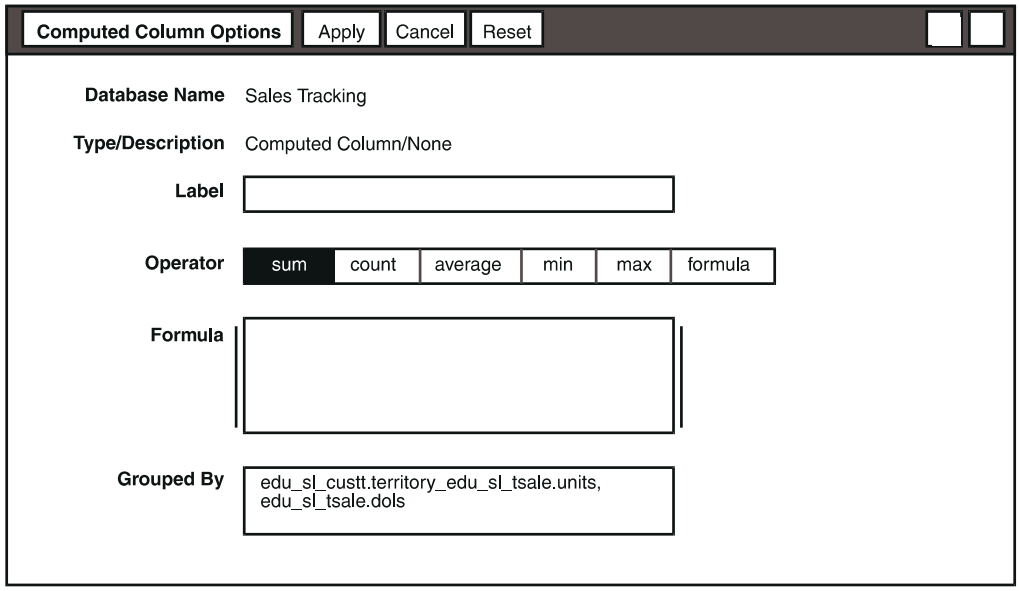


Figure 32. Computed Column Options window for a sum operation

The internal names in the **Grouped By** field appear automatically when you select **sum**, and represent the columns in the Query window. Table 7 describes the fields in the Computed Column Options window.

Table 7. Computed Column Options window fields

Field	Use to
Database Name	This read-only field gives the unique database name of the column. When you open the Computed Column Options window for the first time, it reads Not Yet Assigned and changes to the name of the database to which the Query icon is configured.
Type /Description	This read-only field tells you that the column type is **Computed Column** and the description is **None**.
Label	Enter a name for the new column. You are allowed up to 30 characters. If you change the name, that change also takes effect in the results area and the column is renamed.
Operator	Select a function from the choices in this field. If you selected a column prior to clinking on the **Create Column** button, the **sum** operator is the default. Otherwise, **formula** is the default.
Formula	Enter a formula for the operator to act upon.
Grouped By	This field appears for all operators except **formula**. It specifies how to break out the values from the data. The Query tool copies all non-calculated column names that are in the results area into this section.

Column names must be internal names (that is, **table.column**) for all text fields in this options window. However, since these names automatically appear when you copy a column name into a field, you do not need to know the SQL syntax to use the Computed Column Options window. You can also select the column heading, then click on the **Create Column** button, and the internal name appears in the area automatically.

Setting Up the Calculation

You must now specify the data you want to use for your calculations. To create a computed column:

1. In the Computed Column Options window, enter a heading name for the calculation in the **Label** field.
2. Select the operator from the list in the **Operator** field. For this example, **sum** is selected.
3. Select a column heading in the depiction area.
4. Copy the heading into the **Formula** field of the Computed Column Options window. The internal name for the heading you select appears in the **Formula** field is shown in Figure 33.

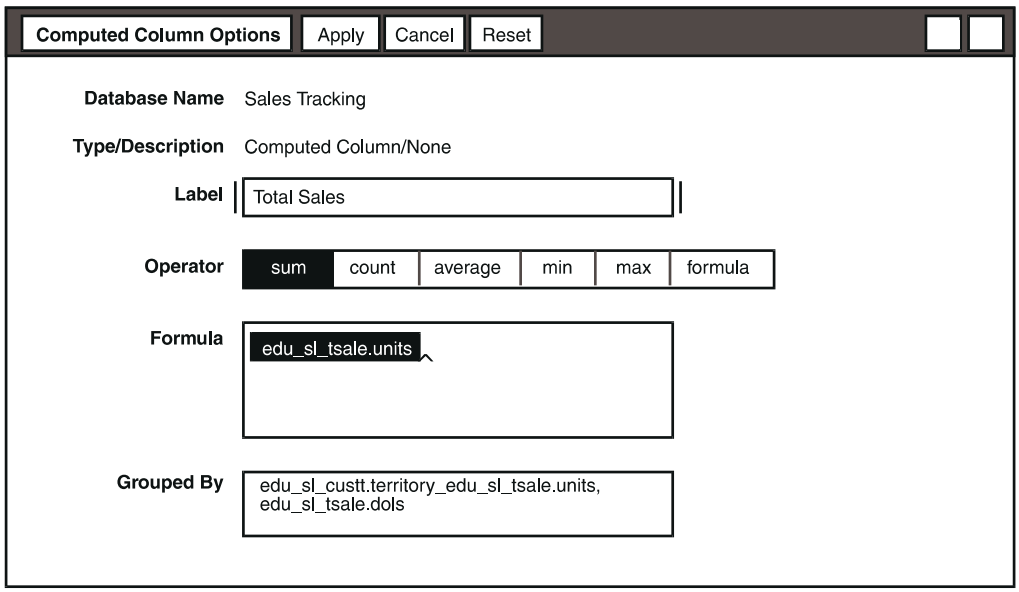


Figure 33. Computed Column Options window after copying the heading

5. Close the Computed Column Options window.

A new column heading with the name you specified in the **Label** field now appears in the Query window. When you click on the **Show Data** button, the query retrieves the calculated data you request, and places it in the new column, as shown in Figure 34 on page 57.

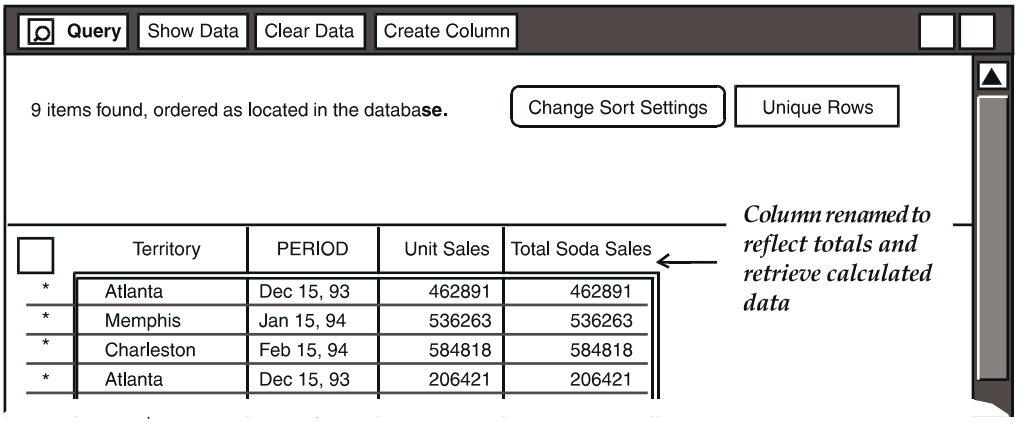


Figure 34. Query window showing the computed column

Using Decimal Arithmetic to Calculate Results

When you set up a calculation that involves data stored as whole numbers (like **Unit Sales**), the results are calculated as whole numbers. However, when you multiply or divide two values in the database, you might want your results calculated to two places of decimal precision. If you need decimal precision in your results, simply multiply the first column name that appears in the **Formula** field of the Computed Column Options window by **1.0**.

You cannot calculate results to more than two decimal places. Although the Query tool transfers full decimal precision, it only displays two decimal places of the calculated results.

A sample Computed Column Options window is shown in Figure 35 on page 58. Notice that the **formula** operator is used for this calculation, and therefore the **Grouped By** field does not appear in the window.

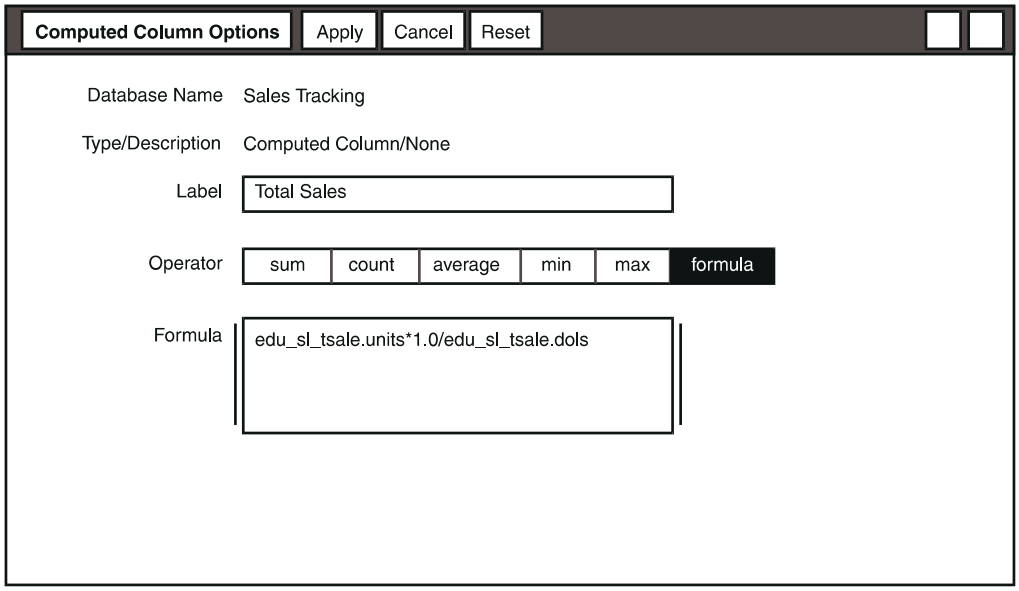


Figure 35. Computed Column Options formula for calculating decimal arithmetic

Changing the Calculation

The method you use to change a calculation depends on whether or not you have already retrieved the computed data. If you have not, then you need to delete the contents of the **Formula** field. If you have already retrieved data, then you must clear the data and reopen the Computed Column Options window. Both procedures are described in the next two sections.

Before Retrieving Computed Data

To delete or change the contents of the Computed Column Options window while it is still open:

1. Place the cursor anywhere inside the field, and click the left mouse button four times. This selects the entire contents of the field.
2. Delete the selected contents.

The information in the field is erased. You can now add your new selection using the instructions provided at the beginning of this chapter.

After Retrieving Computed Data

To change the computation after you have closed the Computed Column Options window and clicked on the **Show Data** button in the Query window:

1. Click on the **Clear Data** button.
2. In the Query window, select the computed column you previously created.

3. Display the options for the computed column and then change the contents of the fields.

Chapter 9. Using Expert Functions to Constrain Data

Chapter 4, “Constraining Data” on page 23 describes the various types of constraints you can apply to limit a query. However, the Query tool permits you to apply constraints in a more specific way using SQL. All operations described in this chapter use the SQL names for data items. This chapter explains how to use expert functions to add more advanced constraints. The following topics are covered:

Adding expert functions

Using the **General Qualifier** field

Using the **Query Strategy** field

Adding Expert Functions

Expert functions allow you to perform more elaborate queries. In Chapter 8, “Performing Calculations” on page 54, you used the **Add Expert Functions** menu item to create a computed column. You need to select the same item to add the **General Qualifier** and **Query Strategy** fields.

When you select **Add Expert Functions**, a new button appears in the Query window header. Notice the change in the Query window header after you select **Add Expert Functions.** The **Show Controls** button is still there, because the Query Control window is not displayed, but the **Create Column** button has been added. If you were to display the **Special** menu again you would see that **Remove Expert Functions** has been added to the list of items in the menu. You select **Remove Expert Functions** when you want to remove the fields from the depiction area.

Now look at the Query Control window. When you select **Add Expert Functions** in the **Special** menu of the Query window, the **General Qualifier** and **Query Strategy** fields appear in the top section of the Query Control window. Any tables already in the query depiction are automatically repositioned to make room for the new fields.

These two expert function fields appear whenever you select **Add Expert Functions**, and the fields remain in the Query Control window until you remove them.

The **Query Strategy** field is only intended for users who are connected to a DBS–250. If you are not connected to the DBS–250, and you attempt to enter a table name in the **Query Strategy** field, a syntax error occurs.

The **General Qualifier** field can be used by everyone. The sections that follow explain how to use these expert functions.

Using the General Qualifier

The **General Qualifier** is a text field that allows you to constrain the query in ways that are not possible using the constraint operators in the Comparison Options window. To do this, you use SQL. The **General Qualifier** allows you to specify a logical **or** between

© Copyright IBM Corp. 1996

values on different columns (the statement is true if either value is true), a logical **and** for values within a column (the statement is true only if both of the values are true), and a percent. For example, to get data on employees with pay codes not equal to zero or to retrieve names of employees in department 7000, enter the following in the **General Qualifier** field:

```
not &SQLNAME(edu_emp.pc)=  or &SQLNAME(edu_emp.dept)=7
```

In this SQL string, **edu_emp.pc** is the internal name for the **pay code** column and **edu_emp.dept** is the internal name for the &SQLNAME keyword instructs the system to substitute the correct SQL name. If you already know the SQL name for the column you want to constrain, you do not need to use the &SQLNAME keyword. This keyword allows you to take advantage of table name or column name indirection without having to know the exact SQL names.

You can enter a maximum of 4096 characters in this field. Although you could enter more than this number, the Query tool will accept only the first 4096 characters. The number of characters you can actually see in the field depends on the font you select in the Set Preferences window.

If you enter more than one line of input in the **General Qualifier** field, you can use the up arrow or down arrow on your keyboard to scroll between lines.

Entering Text Strings

SQL text strings must be entered with care, duplicating exactly those names that are stored in the database. Typing errors can easily occur, and such errors usually prevent the query from running correctly. For this reason, it is recommended that you copy the internal names rather than typing them.

You can enter any combination of characters in a date string providing your entry is in the correct database format. If the format is not valid, you receive a syntax error message when the query is sent to the database.

To copy a name into the **General Qualifier** field:

1. Select the column name.
2. Copy the selection into the **General Qualifier** field.

The internal database name automatically appears in place of the column name.

Using Wild Cards

You can embed wild card characters in the text strings in the **General Qualifier** field. You do so by using the &SQLNAME keyword with the wild card character. For example, to retrieve all products whose number starts with 12, enter the following string into the **General Qualifier** field:

```
&SQLNAME(edu_sl_prod.prodnum) like '12%'
```

Note that the wild card string must be enclosed in single quotes.

Specifying Embedded Blanks

When you are copying SQL names into the **General Qualifier** field, Query does not furnish punctuation in the form of separating commas, and embedded blanks must be treated in a special way. When you are using SQL strings you must add spaces around your operator (**and**, **or**), and use single right quotes to delineate your text constraints. For example if you copy the table name Product Line into the **General Qualifier** field, Query displays:

```
edu_sl_prod.prodline
```

If you want to specify that Product Line in this case is equal to *A1*:

1. Type the keyword &SQLNAME and place parentheses around the copied table name.
2. Type = to put the equal sign after table name.
3. Type A1 at the end of the line. Be sure to include the single right quotes.

The SQL string should look as follows:

```
&SQLNAME(edu_sl_prod.prodline) ='A1'
```

If you want to use the **or** operator in your constraint, type a space, the word or, and another space. Then copy another table name into the field, making sure to add a space between **or** and the new name, and type = followed by 'Total', making sure to include the single quotes. This combination of copying and typing results in the following SQL string:

```
&SQLNAME(edu_sl_prod.prodline) ='A1' or &SQLNAME (edu_sl_proddesc) ='Total'
```

This entry in the **General Qualifier** field tells the database to retrieve all products that are either in the A1 product line, or are product description totals.

Using Query Strategy

This expert field works only with the DBS–250. If you are not connected to a DBS–250, and you attempt to use this field, a syntax error is returned. If you are connected to the DBS–250, you can use the **Query Strategy** field to give a prioritized order of database tables that you can use to improve Query performance. You enter the table name followed by a priority number in parentheses:

```
edu_emp( )
```

List the tables in the order that you want them to be searched during a query. First, **edu_emp** is the internal database name for the [employees] table. It can be copied directly into the **Query Strategy** field. The database name automatically appears in place of the table name. The number tells the query which indexes to use when searching that particular table. Consult your database administrator for information about the appropriate number to use.

Chapter 10. Using Query in Capsule Applications

This chapter explains how to link Query with other tool icons to create automated capsule applications. The Query tool provides the database retrieval function and tools like Layout, Text, Plot, Spreadsheet, and Transformer are used to format, calculate, chart and report the results of the application process. Detailed instructions for using these tools can be found in their respective user's guides. See “IDS Publications” on page 83 for a list of all tools and their order numbers. Specific topics covered in this chapter are:

- Setting up the application
- Controlling the data flow
- Controlling nested capsule applications

Setting Up the Application

The usual order for a capsule application's processing sequence is from left-to-right or from top-to-bottom. If the Query icon is the first to be executed, you would normally place it in the first position of the capsule application's sequence. Any @-variables that you have used in the Query setup will obtain their values from the capsule application's controls window. For information about how to set @-variables, see *Capsule User's Guide*.

Figure 36 on page 64 shows a sample application and describes the processing sequence that occurs when you click on the **Run** button in the capsule tool's window header. The process appears very much like a functional flowchart where the arrow connecting the tool icons control the direction and flow of data or controls.

© Copyright IBM Corp. 1996

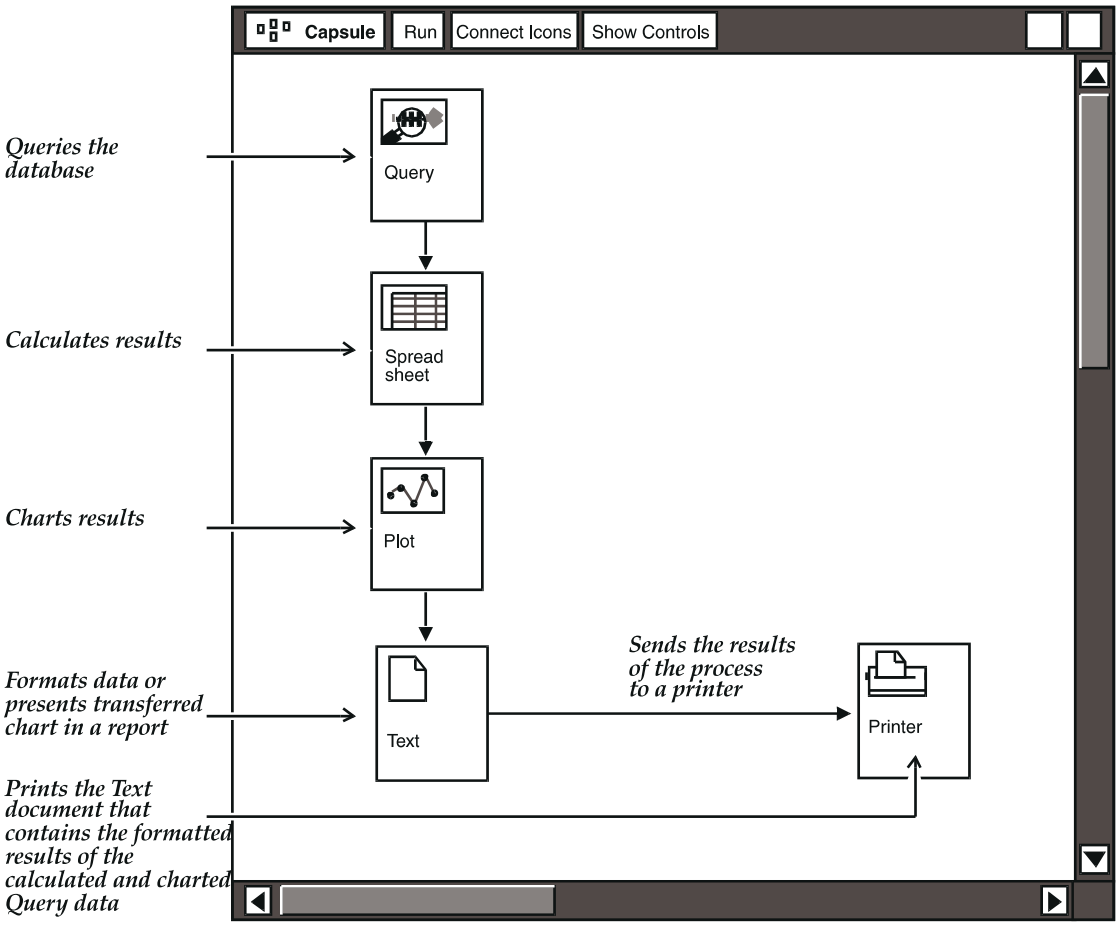


Figure 36. Processing sequence for a sample application

In this example, all icons to the left of the window are source and destination icons, meaning they can both send and receive data. The arrows between the icons actually control the processing.

Arrow settings for the Query icon are described in the next sections. For more detail about capsule application processing and arrow settings, see *Capsule User's Guide*.

Controlling the Data Flow

After you connect a Query icon to another icon in a capsule application window, you need to specify where the Query data will go. The Query icon is typically used to provide source data for Text, Spreadsheet, Layout, Plot and Folder icons, and can also be used to control nested capsule applications.

Stopping the Data Flow

The first section of the Arrow Options window displays options for stopping the flow of data and choosing whether to transfer data attributes. Figure 37 shows the fields and default settings that appear when data is being transferred from a Query icon.

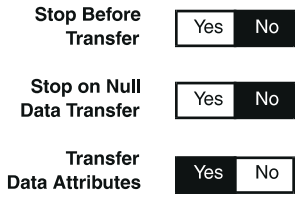


Figure 37. Fields and default settings for data transfer

Before Transfer to the Next Icon

If you set the **Stop Before Transfer** option to **Yes**, processing stops at the arrow leading from the Query icon (before data is transferred to the next icon). When the capsule application stops, you can open the Query icon, check the contents, close the icon, and restart the capsule application process.

If the Source is Null

With **Stop on Null Data Transfer** set to **No**, an attempt to transfer an empty data stream between icons will not cause the capsule application's processing to stop.

However, when the field is set to **Yes**, the capsule application's process halts if the source icon is sending null data. This setting is useful if you have a Query icon that fails to retrieve data. You can debug the problem at the source before the entire application runs.

Transferring Data Attributes

If the destination icon can receive data attributes from a source icon, the **Transfer Data Attributes** field appears in the Arrow Options window between the two icons. The default setting for this field is **Yes**, so you must set the field to **No** if you do not wish to transfer attributes. Query icons can transfer decimal precision to destination icons such as Text and Spreadsheet. If the destination icon cannot receive attributes, the field does not appear when you display the options for the arrow between the two icons.

Specifying a Destination for Query Data

A Query icon cannot transfer portions of data, but the destination icon can be set to receive that data in different ways. A Plot icon receives all of the data, while a Spreadsheet or Text icon can place the transferred data into a specific region. Figure 38 on page 66 shows how the Arrow Options window looks when you transfer Query results to a named Text region.

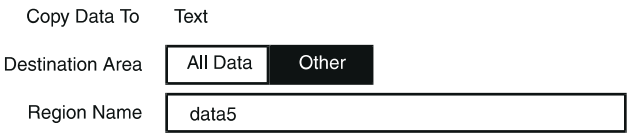


Figure 38. Arrow Options window

Selecting a Function for the Source Data

The fourth section of the Arrow Options window contains options that are visible only when the destination icon is a spreadsheet and when the **Destination Area** option is set to **Other**. Figure 39 shows that part of the Arrow Options window.

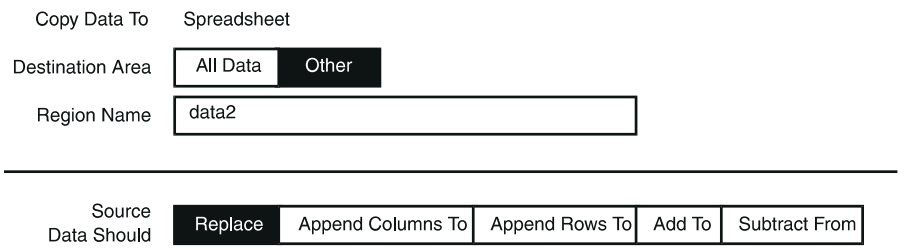


Figure 39. Other options for sending Query data to a spreadsheet

The **Source Data Should** option allows you to choose how you want to incorporate the Query data into the specified spreadsheet region.

Controlling Nested Capsule Applications

You can create nested capsule application routines that are dynamically controlled by the results of a query.

Using Queries to Set Controls in Capsule Applications

The Query tool can be used in a capsule application to set @-variables in the User Input Control window, and to control nested capsule applications. The sections that follow describe these procedures.

Getting @-Values from the Database

You can use a combination of the Query and Spreadsheet icons to automatically retrieve certain values from your database. This is useful if you are running applications and want to take advantage of frequent database updates.

Setting Up a Query

To use a Query tool to control a nested application, set up the results area so it looks like Figure 40 on page 67 when the data is retrieved. Notice that the Query tool's column headings must be used to indicate the @-value designation.

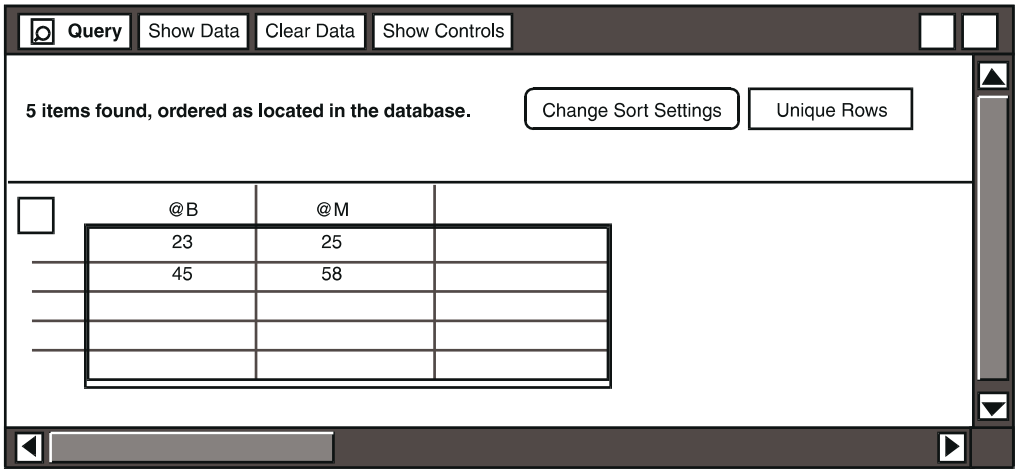


Figure 40. Query tool's iteration setup

Connecting a Query to a Nested Capsule Application

To connect a Query icon to a nested capsule application:

1. Connect the query to the nested capsule application by selecting **Connect Icons**.
2. In the Arrow Options window, type controls as the name of the destination area.

For more information about using the query tool in capsule applications, refer to *Capsule User's Guide*.

Chapter 11. Modifying the Database

This chapter provides some techniques for the advanced Query user or database administrator. The Query tool can be used to modify the database if you have the access keys to perform such tasks. Although the Data Entry tool is commonly used for updating and entering data into your database, the Query tool also lets you perform some modifications quite easily. This chapter explains how to:

Insert or update data

Delete data

It is important to remember that any modifications you make to the database can affect the work of others. This capability should be used carefully.

Each record in the database has a set of fields that contain unique identification keys. This set of fields allows a record to be individually selected, deleted, or updated. Keys are assigned by the database administrator. When the data is copied back into the Query icon, the old records in the table are replaced by the new records. If the record has a blank field, that field is assigned a default value (the N/A value if it exists, or a zero, or an empty string). The copy process is stopped if a key field is blank.

If the record includes a change that is in a key field, that change is not implemented. Instead, the record is appended to the database as a completely new record. You must then delete the old record from the database. In other words, you cannot change any values that are in a key field.

You cannot delete rows in the Query tool. When data is copied into the table, all records are appended to the existing table, regardless of whether it is revised or newly entered data.

Editing the Database

To update database records using Query, you use a spreadsheet icon to contain the data tables. You select the records you want to update from the Query's user catalog, retrieve them so that they appear in the results area, then copy them from the Query window to the spreadsheet. You make the modifications to the records while the data is in the spreadsheet, then copy the new data back into the user catalog of the Query Control window. There are some rules to remember before you begin.

Data types for the records you insert must match the data types defined in your database.

The spreadsheet must have all the columns in the table in the correct order as the title line. One way to ensure this is to copy the title line (the row of column headings) from the Query window results area.

You must fill in all the fields, even those that you do not want to change. If you simply want to replace one column name in a table, you must duplicate the table

© Copyright IBM Corp. 1996

structure when you edit the database. Therefore, to change the table in Figure 41 on page 69:

Territory Information
TERRITORY ID
Territory

Figure 41. Sample spreadsheet column

to one where **TERRITORY ID** reads **TERRITORY ID**, **Territory #**, and **Territory**, you would have to select all three items.

If data keys already exist in the table, records are updated, but if the keys do not exist, your edit inserts data.

If you try to edit data using the procedures provided here and receive an error message, you have not been granted editing privileges by the database administrator. This privilege is not available to everyone because of the potential danger for accidentally deleting the database tables.

To edit the database using the Query tool, you must first copy the data to a spreadsheet. You can then edit data on a row-by-row basis. Select **Disable Edits** in the **Special** menu to end database editing and return to normal functions, or close the Query tool. (Whenever you open the Query icon, editing is disabled by default.)

To modify one or more rows of data:

1. Select **Enable Edits** in the **Special** menu of the Query window. This menu item allows your actions to be interpreted as database edit functions to update the source data on the database.
2. Select a table name from the user catalog and copy it to the Query window.
3. Apply the necessary constraints to limit the number of records that you desire.
4. Click on the **Show Data** button.
5. Click on the **Select All** button and copy the query information into a blank spreadsheet. You can also select a range of rows if you do not want to edit all of the data.

   If you move the data instead of copying it, that data is then deleted from the Query tool and from the database server.
6. Select the information in the spreadsheet that you want to edit and make the appropriate changes.
7. Select the edited records in the spreadsheet and copy or move them to the table name in the user catalog of the Query Control window. You are moving the edited records back to the table from which they originated.

   If the database table you are editing consists entirely of character columns, do not include the first row (that is, the title) of the spreadsheet when you are selecting

records to be copied back into the Query window. If you do so, your titles are copied into the table as an additional row of data.

8. In the Query window, select **Show Data** to see the updated data.

The database replaces the original records with the edited ones. If any modified record is considered illegal by the database, the process stops and the message area indicates the record number and field of the invalid record, as well as a database server error code. The number of records successfully updated is posted in the message area.

Deleting Rows of Data

To delete one or more rows of data:

1. Select **Enable Edits** in the **Special** menu.
2. Select the desired table from the user catalog and copy it to the Query window.
3. Click on the **Show Data** button.
4. Select the rows in the Query window and delete them.

There are some restrictions on which data can be deleted. You cannot delete a record that contains data from more than one table. If you attempt to do so, you will receive an error message.

Inserting Rows of Data

Data is inserted into the copied data on the spreadsheet. To insert one or more rows of data:

1. In the Query window, click on the **Show Data** button to display the table you want to modify.
2. In the spreadsheet window, copy a blank row to the location where you want to insert the new data.
3. Select the new records in the spreadsheet.
4. Move or copy the new records to the table name in the user catalog. That is, move the records on top of the table name from which the data originated.
5. Click on the **Show Data** button to see the updated data in the Query window.

Chapter 12. Troubleshooting

In some instances, sending a database query can produce unexpected results. This chapter lists some of the more common problems you might encounter, and provides recommendations. It also tells you how to use the Query tool for maximum performance and efficiency.

If the problem you are experiencing is not listed here, or if a recommendation does not seem to work, contact your database administrator. This chapter does not cover all possible situations that could affect your database query. For more detailed information about system error messages, see *Messages and Codes*, or ask your system administrator.

This chapter provides troubleshooting help for the following tasks:

Retrieving data

Viewing retrieved data

Verifying retrieved data

Opening a Query icon

Working with views

Enhancing performance of database queries

Retrieving Data

This group of circumstances can occur after you have clicked on the **Show Data** button in the Query window header.

Query returns no data

This can occur if the wrong case is used in a value, if contradictory constraint values are entered, or if incorrect logic is detected in date arithmetic expressions. Verify your query by checking the following:

Be sure you match the case used in the database when you type the value. To verify the case, show all the choices from the Comparison Options window.

Check the query depiction for contradictory constraints. If you use @-variables in constraints, open the Comparison Options window and check each constraint.

Correct the constraint. Date constraints that use the **Between** operator must always go forward in time.

Check the logic in your date arithmetic expression to ensure that it specifies a formula without contradictions. For example, to retrieve data for a three month period ending on a given month, you might insert a constraint that uses the **Between** operator:

@F-2 months @F

© Copyright IBM Corp. 1996

If you accidentally transpose the variables as shown in this example, no data will be returned:

@F @F-2 months

Query returns no data and an important message appears

A date constraint might have been entered incorrectly. The message appears when you attempt to show data. Usually, the message explains that the date format is unrecognizable. Use one of the following methods to find and correct the incorrect constraint:

Look in the depiction area for obvious mistakes such as missing commas between the month and year. Then display the constraint's options window for the incorrect constraint and correct the entry.

Display the Constraint Options window and show all the available choices. Then delete the constraint and copy the entry from the Choices window.

Refer to the date format table for the correct date entry format.

Check the desktop preferences window settings to see if the date format has been changed.

Table 8. Acceptable date entries

Date	Correct formats
Day	January 15, 1995; Jan 15, 1995; Jan 15, 9;5 Jan 15 95; 15 Jan 95; 15 January, 1995
Month	January, 1995; Jan, 1995; January, 95; Jan, 95; Jan 95; 95 January
Year	1995; 95
Odd-Bimonth	DJ, 1995; DJ 95; 95 DJ
Even-Bimonth	JF, 1995; JF 95; 95 JF
Quarter	1Q95

Query retrieves too much data

The Query might take a long time to retrieve data, or the message in the Query window indicates that too many items have been retrieved. Look at the Query to see if any of the following problems exist:

The query can be set up to retrieve duplicate rows of data.

Check the depiction area to see if the constraints are restrictive enough to limit the data, then check for duplicate columns of data in the results area of the Query window. If you find duplicate rows, click on the Query window to ensure that only unique rows are returned.

Check to ensure that you constrained tables as needed.

You need to stop a database query

A complex query will take a long time to return results. If you have clicked on the **Show Data** button and need to stop the Query:

Select **Stop Query** in the **Special** menu of the Query window header if you know the query is complex and will take a long time.

Press the Stop key if you believe the query can return data quickly. This is the fastest way to stop a database query.

Remember, either of these methods will cause the database query to be incomplete. You must click on the **Show Data** button again to return results.

Viewing Retrieved Data

You cannot see all of the items in the user catalog

The window might be too small or there might be many items in the user catalog. Do one of the following:

Resize the window.

Scroll the window to see all of the items.

You cannot see all the tables in the query depiction

The window might be too small or there might be many items in the depiction. Do one of the following:

Resize the window.

Scroll the window to see all of the items.

Click on the **Show Query** button to rearrange the query depiction.

You cannot see all the constraints in the query depiction

Resize the window.

Scroll the window to see all of the items.

Click on the **Show Query** button to rearrange the query depiction.

Default joins are missing from the query depiction

This omission can occur when redundant joins have been created using the DBA Tool. When you are joining tables using the DBA Tool, and you have numerous joins, some of which are redundant, the Query tool will often display just those joins that it deems essential to perform the query. Occasionally, the wrong joins might be selected. There are two options:

If you have access privileges to the DBA Tool, you can remove the redundant joins.

Use the Query tool instead of the DBA tool to create manual joins that reflect the query you wish to perform. Chapter 3, “Building a Query” on page 15 tells you more about creating manual joins.

The user catalog area contains strange pieces of text

If you create a view in Query and give it a very long description, such as 75 characters, when you switch back to display Column Names, a portion of the description you entered remains in the window.

Click on **Refresh** in the Special menu.

Verifying Retrieved Data

A query returns erroneous results

If you send a query that contains an embedded view, and both the view and the query contain the same constraint, you can retrieve incorrect results.

Remove the duplicate constraint from the view to fix this problem.

There can be a conflict between what is actually contained in the fields in the Workstation Tools Data Dictionary (WTDD), and the way the query is requesting that it be displayed. If this happens, you might see columns displaying a zero, or incorrect values for character data.

Consult your database administrator.

Performing a calculation using computed columns causes an error message to appear

If the query is a complex one containing many columns of data, you might have exceeded the Query tool's capability to perform this function.

Check your calculations. If everything looks correct, consult your database administrator.

Items in the user catalog, results area or query depiction have overstrikes through them

This occurrence is usually signaled by an important message indicating that phantoms exist in the WTDD. A phantom can occur when the database administrator deletes or renames a category, table, or column, and needs to update your Query tool about the changes. If this happens, the Query tool leaves each item that has been deleted or renamed in the DBA catalog, with an overstrike through its name.

If you want to retain the query setup, copy the Query icon before you try to delete the phantoms.

To delete phantoms:

1. Copy the Query icon containing the phantoms.
2. In the original icon, select and delete the phantom tables from the query depiction.
3. Select and delete the phantom tables in the user catalog.

You can now use the copy you made of your query to help recreate the original setup.

A query returns incorrect matches when you constrain using a wild card character (%)

This can happen if you are using a SQL/DS Gateway. Because of the way the SQL/DS Gateway handles this pattern-matching character, trailing blanks in the characters you are asking Query to retrieve can cause undesirable matches to be returned.

Opening a Query Icon

Query icon does not open and an error message appears.

Ensure that you used the correct user ID and password. Then check your system messages. The file server could be offline or the database server might be unavailable. Consult your system administrator.

The icon might have been secured by another user. It is possible to secure a database access tool icon so that it can be opened only by an authorized user. You might receive one of these icons in the mail, but find that you cannot open it.

Check the Icon Options window to see if the icon is secured, then contact the icon's owner, or consult your database administrator.

Working with Views

A query depiction disappears while you are trying to copy new tables into the depiction area

If you are using multiple views in a database query, perhaps to test data retrieval, or to create different versions of a similar query, be sure to save each iteration as you make changes. If you forget to do so, copying tables into the depiction area while you are working with views can cause an in-process view to disappear.

You are unable to display the Choices window

SQL prohibits a query view to reference other tables in the database. A query view groups items from the database according to your specifications. Those groupings do not really exist in the database. Therefore, when you are working with query views, the **Show Choices** and **Show Constrained Choices** buttons do not appear in the Comparison Options window.

Enhancing Performance

The database query takes a long time to retrieve data

Consider rebuilding the query so that it selects from smaller tables rather than larger ones.

The Query returns more data than you expected

Check to ensure that you have constrained the data properly:

Have you added constraints to return just the data you need?

If you are running the Query in a capsule application and using @-variables to constrain data, make sure your constraints match the values in the User Input Control window. If you add an @-variable constraint and the corresponding value field in the capsule tool's control window is left blank, the Query tool will retrieve all data just as if you left the table unconstrained.

A capsule application containing several Query icons takes a long time to finish processing

If appropriate, construct an iterating application instead of using multiple Query icons.

Glossary

access key. A code number that restricts access to an IDS icon. Only users who have been assigned this code number can use the icon.

aggregate value. A set of values stored in the database as one value.

Ascending. The choice in the Sort Level and Sort By fields of a controls window that arranges the data in order from smallest to largest (A to Z, 0 to 9).

@date. A system-defined @-keyword that returns the date.

@-keyword. A system-supplied variable that returns current values within a Capsule application or within the headers and footers of a printable icon.

attribute. A column in a database table (or relation).

@user. A system-defined @-keyword that returns the user's user ID.

@-value. The value assigned to an @-variable in the Capsule's User Input Controls window.

@-variable. A control variable used to represent a value in any tool that can be run inside a Capsule application. These values are frequently used as constraints and variable titles in the Query, Plot, Spreadsheet, Layout, and other IDS tools.

average. An operator that adds all the returned values together and then divides the result by the number of returned values.

Beginning value. The field where the user enters a value that constrains the returned data to that which is greater than or equal to the entered value.

Between. An operator that constrains returned values to those that are greater than or equal to the Beginning Value entered and less than or equal to the Ending Value entered.

***Blank Icons file drawer**. A file drawer icon that contains one of each icon that an IDS user is likely to need. An asterisk at the beginning of a file drawer name indicates the file drawer is a standard system file drawer distributed on the software installation media.

button. An object, in a window or window header, that contains graphics or text and is used to open other windows, open a menu, perform an action, or set a condition.

Capsule. A tool for building an automated processing sequence using other icons.

Capture SQL. The choice in the Special menu that opens the Query Log window in which SQL statements are recorded.

capture window. A special Text window used to record the SQL statements sent to the database. In the Query tool, this window is called the Query Log.

insertion point. The symbol (usually a blinking vertical bar)that shows where mouse or keyboard input will appear in a document or field.

category. The highest-level division of a logical database. A category contains tables, and tables contain columns and rows. Logical joins between tables are defined at the category level.

Choices. The window that displays a one-column list of all acceptable values that can be used in a particular field. The user can select values from the list instead of typing them into the field. The user can open a Choices window by clicking on a button labeled Show All Choices or Show Constrained Choices.

choices window. A window that displays a list of all unique values in a particular database column. The user can select values from the list instead of typing them into a field. Contrast with browse list.

click. (1) (1) To press and release a mouse button without moving the pointer. (2) To press and release the left mouse button while the pointer is on a desired object. For example, “Click on Show Controls,”“Click anywhere in the window header.” (2) On a left-handed mouse, the right mouse button is used instead of the left one.

column. A vertical field in a table. Columns may be identified by letters or text strings at the top of the display area. Columns are identified by letters at the top of the display area.

© Copyright IBM Corp. 1996

column names. Subdivisions of table data. These names are grouped after their respective table names in the user catalog.

Column Options. The options window in which the user can change a column heading.

Comparison Options. (1) An options window for column names in which the user can specify a label, an operator, and a value, and can open a Choices window. (2) An option window for joins in which the user can specify two labels and an operator.

computed column. A column created to contain the results of calculations performed on one or more columns.

Computed Column Options. The window in which the user can create computed columns.

constraint. A limitation on the range or values of data retrieved from the database to reduce the amount of data that is returned. For example, the user might constrain data based on territory and period so that only data relevant to New York during March, 1992 is retrieved.

container. An icon, such as the Folder and File Drawer icons, that can hold other icons.

controls window. A window that displays settings that the user can view or modify. Usually, to open a controls window the user clicks on a button (often named Show Controls) or selects a choice from a menu.

Create Column. The button in the Query window that opens the Computed Column Options window or the Column Options window.

database administrator (DBA). The person who issues administrative instructions to a database service. The DBA defines the data tables used by data access tools, monitors database performance, and assigns user access.

date format. The internal storage representation of dates with a particular date resolution.

DBA catalog. The database listing containing all categories and tables available in a database on your system.

desktop. The screen space occupied by IDS.

IDS. The IBM Intelligent Decision Server.

double-click. (1) (1) To click a mouse button twice (transitive), as in “point to the icon and double-click the left mouse button. ”(2) To click the left mouse button twice (intransitive), as in “double-click on the Envelope icon.” (2) On a left-handed mouse, the right mouse button is used instead of the left one.

drag. To move an object on the screen by pointing to the object, pressing the left mouse button, and then moving the mouse while holding down the left mouse button.

On a left-handed mouse, the right mouse button is used instead of the left one.

Ending Value. The field where the user enters a value that constrains the returned data to that which is less than or equal to the entered value.

field. An area in a window where the user enters information.

Envelope. An icon used to create messages and to send messages and icons to other users.

expert function. An advanced capability that permits users to perform calculations in constraints.

field. (1) A specified area used for a particular class of data. (2) In a window, an identifiable area where the user can enter information (field), select a choice (field of choices), or view information.

file. A collection of information stored as a unit on a disk or other storage media.

file drawer. A container, created or renamed by the system administrator, that may be shared among desktops. A file drawer typically contains folders.

folder. An organizing container that resides on a single desktop and is used to store icons and other folders. Folders are represented on the IDS desktop by the Folder icon, on which a file folder is depicted.

From Data Access Controls. The choice in the Database Logon field of an Icon Options window specifying that the user ID and password are to be determined from the settings in the desktop Data Access Controls window.

General Qualifier. An expert function that permits users to constrain a query with an or, an and, or a percent, and enter @-variables.

icon. An electronic desktop representation of a query, report, plot, folder, or other object. Every workstation tool is represented by an icon.

icon graphic. The small picture that appears in the title block of a tool window header and in the tool icon. The icon graphic usually depicts a tool's function, so you can easily identify a tool window or icon at a glance.

Important Message. A window that displays a message that requires an acknowledgment from the user. An important message can be generated by a tool or service, or by the system administrator.

index. A list of pointers in a file used to locate records in a database.

inner join. A join between two tables in which equal or identical values from both tables are listed in the results.

internal name. In the IDS environment, the name by which a database item is stored in the Workstation Tools Data Dictionary (WTDD). Each database category, table, and column has an internal name based on naming conventions required by the database system. To make these names more useful to users, each internal name has a corresponding external name, called a print name, that the IDS tool user sees on the screen. For example, a table with the internal name DBA_EMP might be known to the user as the Employee table. Contrast with print name.

join. The method used to match tables of identical information in relational databases. A join between tables is usually represented by a line with an equal sign in the middle.

key fields. Certain fields within a database that are used to uniquely identify records. Key fields cannot be changed nor can blank key fields be inserted using the Query tool.

key holder. A user to whom the system administrator has assigned an access key.

Label. The field where the user can enter a name for a column or Query view.

Like. A constraint operator that enables the user to match string data. The user can use the percent-sign (%) and underscore (_) wild cards with the Like operator. For example, in a search for the names of various colors, entering a Like constraint of rose% might return roseblush, rosebud, and rosewood. Contrast with Not Like.

list separator. A character, such as a comma, used to break a text string into separate items.

manual join. A join between two table created from within the Query Control window. Manual joins are used to enhance Query performance.

menu. (1) A list of available choices that the user opens by pressing the left mouse button while the pointer is on certain buttons. (2)

message bar. The rectangular area along the bottom of the desktop, where advisory or error messages are displayed.

Minimize button. The window header button that reduces the size of a window so that only the window header is displayed on the desktop.

minimized window. An IDS window that is open but has been reduced to the size of its window header.

Name and Size. The choice in the Display field of an Icon Options window that displays the page in which the user can change the name and access privileges of an icon and can view other information about the icon such as modification dates, size, and ownership.

N/A value. A data value that is unavailable.

Not Between. A constraint operator that enables the user to select all values that are not contained within the identified range.

Not=. A constraint operator that enables the user to select all values that are not equal to the specified value.

Not Like. A constraint operator that enables the user to request all values that do not match the character string entered. This operator can be used with the percent-sign (%) and underscore (_) wild cards. Contrast with Like.

Not Null. A constraint operator that returns all values except those that are identified as null in the database. Contrast with Null.

Null. A constraint operator that returns all values identified as null in the database. Contrast with Not Null.

numeric date format. The representation of a date as the number of days before or after January 1, 1970. For example, 0 represents January 1, 1970, 1 represents January 2, 1970, and -1 represents December 31, 1969. Dates are typically stored in a database in numeric date format.

open. To perform an action that causes a window or menu to appear on the desktop. For example, the user opens a tool window by double-clicking on an icon.

Operators. The field of choices such as =, <, Average and Between that enable the user to constrain data relative to a specified value or to create calculations.

options window. Any window that the user can open by using the Optionskey or the options function on the pop-up function pad. Options windows enable the user to view and modify the properties of the selected object.

Other Data. The choice in the Display field of the Icon Options window that displays settings for configuring the icon to a specific database.

outer join. A join between two tables in which all values from the first, second, or both tables are listed in the results, whether or not table values match.

page. In a window, an alternate view that provides a different set of options for setting controls and defining the behavior of an icon.

password. A unique string of characters known to a computer system and to a user, who must specify the character string to gain access to a desktop, gateway, database, system, or service within a system.

phantom. A catalog, table, or column that has been deleted from or changed in the logical database, but whose name remains in the Query tool windows with an overstrike through it to alert users that the item no longer exists.

plot. A graphical representation of one row or column of numerical values.

Plot tool. A tool used to create line, bar, text, and pie graphs.

print name. The name of a database item (such as a column or table) that is used in IDS windows instead of the more cryptic name by which the data item is identified in the database. Print names are identified in the Workstation Tools Data Dictionary (WTDD) and are converted during data retrieval to internal names that can be understood by the database. Contrast with internal name.

query. A request for data from a database, based on specified conditions.

Query tool. A tool used to retrieve data from a database, based on specified conditions.

Query Catalog. A window that provides a catalog of column and table names from which users create database queries.

Query Control. A window where users create a depiction of a database query.

query depiction. The database tables shown in the Query Control window after column names are copied or selected from the Query Catalog window.

Query Log. The window that displays the SQL statements sent to the database while data is being retrieved.

Query Strategy. An expert function that only works with the DBS-250, Query Strategy permits you to set a priority list for constraints.

relational database. A database that is organized and accessed according to relationships between data items. These relationships are expressed as tables related by data values, rather than by pointers or memory locations.

row. (1) The horizontal component of a table. A row contains exactly one value for each column of the table. (2) A record, or single instance, in a database table.

scroll. To move the image in a window up, down, left, or right to view information that is not otherwise visible within the boundaries of the window.

scroll box. The part of a scroll bar that indicates the position and proportion of information currently visible in a window relative to the total amount of information available. The user can drag the scroll box to a different location in the scroll bar to bring different information into view.

scroll button. The part of a scroll bar that contains an arrow. Clicking on a scroll button causes the window to scroll up, down, left, or right.

Select. The transformer that sends specified data columns from 1 input region to as many as 10 output regions.

Select All button. The button in a window header that causes everything in the window to be selected. The Select All button looks like this:

server. The computer on which a service runs.

Show Constrained Choices. A button in the options windows of dimension headings and dimension column names. This button opens a Choices window that displays a list of all acceptable constraint values based on constraints the user has already applied to the selected table.

Show Controls. A window header button, usually in the main window of an icon, that opens a controls window in which the user can specify settings that determine icon behavior, how a program is processed, or both.

Show Data. The button in the Query window header that causes the retrieved information from the database to be displayed.

Show Query. The button in the Query Control window header that causes the columns selected from the user catalog to be displayed in the query depiction and in the Query window.

Sort Controls. The window in which the user can change the sorting sequence of retrieved data.

Source. A field in many constraint windows that enables the user to choose between entering constraint values directly in that window or using values that have already been entered in a Text icon.

Special menu. A menu that appears when the user points to the Special button and presses the left mouse button (or, in the case of a left-handed mouse, the right mouse button). In tool windows, the menu choices are usually reserved for special functions or controls associated with the icon.

Spreadsheet. A tool used to perform calculations and analyses, schedule projects, and perform iteration in embedded Capsules.

SQL. Structured Query Language, a command language for use with relational databases. The language consists of statements to insert, update, delete, query, and protect data. IDS tools use SQL to communicate with database servers.

SQL Entry. A tool used to run Structured Query Language (SQL)programs.

SQL ID. The field in the Icon Options window in which the user specifies the name of the creator of the database.

Capture SQL. The choice in the Special menu that opens the Query Log window. A time stamp recording the date and time when the window was opened is added to the icon name.

stopwatch pointer. A graphic representation of a stopwatch that replaces the pointer to inform the user to wait until a process has been completed.

string. A sequence of characters.

Structured Query Language. **See** SQL.

Sum. An operator in the Formula page of the Computed Options window that enables the user to add together the values from two columns or a value from a column and a constant value.

system administrator. In IDS, the person who is responsible for maintaining and configuring IDS, including its workstations, servers, and network.

table. In a database, a single file, consisting of data organized in columns and rows.

table name. The title or label for a database table.

Text. A tool used to write and format text documents, format the results of data access tools, and receive data and images from other tools and PC applications.

title block. A window header button that contains the icon graphic and the window title. The user can click on the title block to close the window.

tool. An IDS workstation program, such as Text, Reporter, or Query. Every tool is represented by an icon.

tool window. A window that opens when the user double-clicks on an icon or points to an icon and presses the [Size] key. A tool window is the user's view of a tool.

user catalog. The page of the Query Catalog window that contains the index of table information from which column names are selected.

user ID. A name used to access a desktop, system, gateway, or database service.

User Input Control. The window in which the user can set or view @-variables, their descriptions, and their type designations.

User ID. In the desktop data access windows (Add Entry, Change Mainframe Password, and Entry Options)and in the Icon Options windows for data access tools, a field where the user types a user ID that

the database can identify as valid. This may or may not be the same as the user's IDS user ID.

Value. A character field in the Comparison Options window that contains the text parameter that the operator will act upon.

Value field. The choice in the Source field of constraint windows that the user selects to enter constraints into the Value field of that window. Contrast with Text Icon.

view. A copy of a query that can be named, stored, edited and redisplayed for use in comparison analysis and query computations.

wild card. A special character, such as a percent sign (%) or an underscore (_), that can be used to represent one or more characters.

window header. The area at the top of a window that contains the title block, command buttons, and other buttons.

workstation. A computer at which the user can use IDS to perform work. Depending upon the model, workstation components include the monitor, keyboard, mouse, and system unit, which may house a disk drive.

Workstation Tools Data Dictionary (WTDD). A logical data dictionary that standardizes the IDS workstation view of data across many relational database management systems. The WTDD stores the print names of database items, defines how the items are to be displayed, and includes other logical attributes of data. The WTDD is implemented as a set of standard relational tables that can be managed through the DBA tool.

WTDD. See Workstation Tools Data Dictionary.

WTDD Owner. The field in the Icon Options window in which the user specifies the name of the owner of the WTDD catalog.

Year Format. In the Country page of the Set Preferences window, the field where the user specifies the format of a year in a date entry. The choices are yy (two digits, as in 93)or yyyy (four digits, as in 1993).

IDS Publications

This section lists IDS publications, including order numbers. To order copies of the books listed here, or to get more information about a book, see your IBM representative.

General Publications

Getting Started with the Intelligent Decision Server (SC26-8892)

Working with the Intelligent Decision Server (SC26-8893)

Installation, Configuration, and Administration Publications

Installing IDS LAN Components (GC26-8877)

System Administration Guide and Reference (SC26-8890)

Messages and Codes (GC26-8878)

Database Gateway Services Publications

Installing Database Gateway Services for UNIX Environments (GC26-8884)

Configuring and Administering Database Gateway Services (GC26-8885)

Administering Databases with IDS (SC26-8891)

Host Services Publications

Planning MVS Host-to-LAN Connectivity (GC26-8880)

Installing Host Services and Configuring Host-to-LAN Communications for MVS (GC26-8881)

Administering Host Services for MVS: DB2 and Cooperative Application Services (GC26-8882)

Developing Applications with Cooperative Application Services (GC26-8883)

Configuring and Administering Mail Gateway Services (GC26-8923)

Open Data Access Service Publications

Developing Applications with Open Data Access Service (SC26-8888)

Data Access Tools Publications

Browser User's Guide (SC26-8900)

Data Entry User's Guide (SC26-8897)

DBA Tool User's Guide (SC26-8894)

SQL Entry User's Guide (SC26-8898)

Data Analysis Tools Publications

BASIC Tool User's Guide (SC26-8908)

Query User's Guide (SC26-8901)

Spreadsheet User's Guide (SC26-8904)

Presentation Tools Publications

Layout User's Guide (SC26-8902)

Plot User's Guide (SC26-8905)

Reporter User's Guide (SC26-8903)

Text User's Guide (SC26-8896)

Application Development Tools Publications

Transformers Guide (SC26-8907)

Capsule User's Guide (SC26-8906)

Forms Design User's Guide (SC26-8899)

Interacting with PC Applications (SC26-8895)

Installing and Configuring IDS Open Clients (GC26-8889)

© Copyright IBM Corp. 1996

Index

Special Characters

@-variable values 27
@-variables 26, 27
  specifying 27
\ 30
= sign 30
> 30
>= 30

A

and operator 62
  specifying 60
appending values 33
  by typing 33

B

Beginning Value 31
Between 30
buttons
  Create Column 54
  Show Data 44
  Show Query 44

C

calculated data
  retrieving 56
calculating results
  whole numbers 56
capsule tool 27
capture process
  stopping 46
Capture SQL 45
capturing SQL 45
Category 5
Change Sort Settings 39
changing
  column heading 41
  constraints 35
  Display option 49
character limits
  Label field 30
character strings
character strings *(continued)*
  using wild cards 25
checkmarked selections 16, 17
choices
  displaying 32, 34
Choices window 32
  closing 32
Clear Data button 47, 54
clearing Query data 47
Column field 39
column heading
  changing 41
  renaming 41
Column Names 9
Column Options window 41
  closing 41
columns
  computed
    in views 52
  inserting 18
  specifying order 18
commas 31
  used in the Value field 32
  used to separate constraints 24
comments
  adding to Description field 49
Comparison Options window 23, 29, 32
  for joined columns 37
Computed Column Options window 54
  fields
    Formula 55
    Grouped By 55
    Label 55
  for decimal arithmetic 57
  options
    Operator 55
  settings
    Internal Name 55
    Type/Description 55
computed columns 52
  creating 54, 56
configuration instructions 5, 7
constraining
  to a single value 24
  to multiple values 24
  with expert functions 60

© Copyright IBM Corp. 1996

constraints
applying 27, 35
definition 23
deleting 35
in views 74
introduction 23
operators 23
retrieving a range of data 25
types 23, 27
date arithmetic 26
expert functions 60
multiple 24
single 24
copying columns
one at a time 18
together 18
copying values
from the Choices window 33
Create Column button 54
creating computed columns 54
when visible 12
creating manual joins 20
between views 52

D

database
deleting data, procedures 70
inserting data, procedures 70
table descriptions 9
database catalog
displaying 42
database category 41
changing 41
selecting a new category 42
database error code 70
Database Name field 5
database tables
order of 62
date arithmetic, expressions table 26
date entries
day 72
even-bimonth 72
month 72
odd-bimonth 72
quarter 72
year 72
DBA catalog 41
DBA Catalog page
Descriptions 9, 42
DBA Catalog page *(continued)*
Table Names 9
DBA Tool 20, 73
DBS–250 60
using Query Strategy 62
decimal precision
calculating 57
deleting
phantoms 74
tables 21
depiction
updating 44
description
view 9
Disable Edits 69
displaying
choices 34
duplicate rows
eliminating 40, 41
caution 40

E

embedded blanks 62
embedded capsule applications
controlled by queries 66
Ending Value 31
equal sign 20, 30
expert functions
General Qualifier 60
Query Strategy 60
removing 60
using 60

F

fields
Column 39
Database Name 5
Description 49
Formula 56
Label 50
Operator 56
Server 5
SQL ID 5
file drawer
* Query Icons 8
Formula field 54, 56
calculating decimal arithmetic 57

functions for source data 66

G

General Qualifier field
adding 60
character limitation 61
removing 60
sample entry 62
greater than 30
greater than or equal to 30
Grouped By field 54

I

icon options
changing 5
Icon Options window
description 5, 7
fields 5
Other Data page 5, 7
identification keys
for database records 68
improving Query performance 20
incomplete results, reason for 44
Internal Name 29
internal names 56
used as column names 61

J

joined columns 37
joins 19
constrained 38
default joins
creating with DBA tool 20
line 37
manual
creating 20
redundant 73
setting options 37

K

key fields 68
keys
\ 34, 44

L

Label field 27, 30, 54
character limitation 30, 41
less than 30
less than or equal to 30
Like 30

M

manual joins 20, 21
moving the insertion point 54
multiple constraints 24

N

New View Options window 49, 50
Not = 30
Not Between 30
Not Like 30
Not Null 30
Null 30

O

Operator field 30, 56
operators
{\ 37
\ 38
Between 31, 36, 37
used with wild card 25, 36
changing 30
equal sign 30
formula
using for calculations 57
in constraints
Between 25
in joins
changing 38
not = 30
operators, Between 31
option windows, Computed Column Options 54
or operator 62
specifying 60
Other Data page
input required 5, 7
overstrikes 74

P

Password field 5
pattern-matching, see also wild card 75
percent sign
  used as wild card 25
performance
  improving 40
Printer icon 48

Q

Qualifier, see General Qualifier 60
Query
  combining with other tools 1
  data 1
  description 1
Query Catalog window 9
Query Control window 8
  buttons
    Clear Query 8
    Show Query 8
  closing 9
  expert functions 60
Query data
  arranging 39
query depiction 73
  changing the arrangement 19
  constraints 73
  deleting 48
  joins 73
  layout 60
  rearranging 73
  tables 73
Query functions, associated windows 13
Query icon
  capsule application arrow settings 67
  connecting to embedded capsule application 67
  opening 8
Query Log
  header 45
  icon 46
  see also Capture SQL 45
query performance
  improving 62
Query setup
  to control capsule application processing 66
Query Strategy field
  adding 60
Query tool
Query tool *(continued)*
  column headings 66
  securing 7
  using in a capsule application 66
Query window
  buttons
    Change Sort Settings 11
    Clear Data 12
    Show Data 12
    sorting 39
    Unique Rows 11
  default configuration 8

R

recording a Query session 45
removing query data 47
Reporter tool, when to use 51
Reset 42
results area
  copying from user catalog to 17, 19
  deleting headings in 21
retrieving
  current date 26
retrieving data 17
Return to User Catalog button 42

S

scrolling
  user catalog 49
secured icon
  opening 75
securing icons 7
Select All button 11
  defined 80
selecting
  columns 15
  values from the Choices window 32
Server field 5
setting options
  in joins 37
Show All Choices button 30, 32, 34
Show Constrained Choices button 34
Show Controls button 9, 12
Show Data button 73
Show DBA Catalog button 42
Show View button 50
single right quotes in SQL strings 62

Sort Controls window
closing 40
opening 39
sorting levels
removing 40
sorting order
changing 39
displaying 40
setting 39
Special menu 45
Add Expert Functions 60
specifying
@-values 27
multiple values 31
Spreadsheet
using to modify database records 69
SQL ID field 5
SQL name 39
SQL text strings 61
copying 61
Stop Query 73
stopping
the Choices display 34
stopping a Query 44, 73
methods
Stop Query 44
stopping the capture process 46
Structured Query Language (SQL) 60
system area
buttons 39
system messages 11, 32, 49

T

table numbers 10
testing
a view 49
Text tool
printing 1
window 46
Type/Description field 29

U

underlined selections 16, 17
unique rows 41
Unique Rows button 40
user catalog 9
changing display 10
page
Column Names 9
user catalog *(continued)*
page *(continued)*
Descriptions 9
selecting from 15
showing saved views 51, 52
using commas
for multiple values 31
when appending values 33

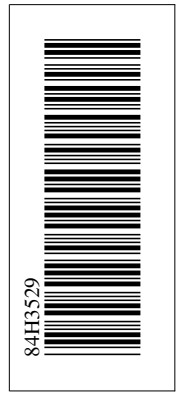
V

Value field 30
value range, in constraints 36
values
specifying a range 30
variables 26
view description 73
adding 49
view names
character limitation 73
views
comparing 52
creating 49, 50, 51
data retrieved 50
deleting 51
naming 49
shown in Query depiction 52
testing 49
using for comparison 52
using to perform calculations 52

W

windows
Choices 32
closing 35
for constrained choices 34
closing 9
Column Options 41
Comparison Options 23
for joins 38
New View Options 49
opening 9
Query 8, 11
Query Catalog
DBA catalog 42
Query Control 10, 44
Query Log 45
closing 46
Sort Controls 39, 40

IBM®

Part Number: 84H3529

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.