

BL03 - Mastering SQL Performance using Visual Explain Basic Workshop



International Technical Technical Organization

Contents

Contents	iii
Lab 1. Operations Navigator setup and introduction	v
Task 1: Getting connected	v
Task 2: Database options	vii
Task 3: Familiarization with the database	viii
Task 4: SQL environment options	x
Lab 2. An introduction to Visual Explain	xiii
Task 1: Query Optimizer debug messages	xiii
Task 2: Accessing Visual Explain	xv
Task 3: Visual Explain with Performance Monitor Data	xvii
Lab 3. Non-keyed access methods	xxi
Task 1: Table scan	xxii
Task 2: Parallel table scan	xxiii
Task 3: Skip sequential table scan	xxiv
Task 4: Comparing the job log with Visual Explain (Optional)	xxvi
Task 5: SQL statements	xxx
Lab 4. Keyed access methods	xxxiii
Task 1: Using Visual Explain to show keyed access methods	xxxiii
Task 2: Additional keyed access methods explained	xxxv
Task 3: Temporary Index	xxxvi
Task 4: Comparison of Visual Explain and job log messages (Optional) ..	xxxviii

Lab 1. Operations Navigator setup and introduction

Introduction

To use Visual Explain proficiently, you must have a knowledge of Operations Navigator. This lab provides that knowledge. During the lab, you connect your workstation to the iSeries server and establish an Operations Navigator session. From this session, you will use some of the database options and database functions within Operations Navigator on the iSeries server.

Note: To run SQL or system commands, XX in library names, profile names, and so on refers to your team number, for example, VETEAMXX, TEAMXX and VEJOBXX. Refer to your lab worksheet for details.

Objectives

Upon completing this lab, you will:

1. Establish a connection to the iSeries server.
2. Learn basic navigational skills using Operations Navigator.
3. Access the database using Operations Navigator.
4. Select libraries using Operations Navigator.
5. Configure the ODBC Data Source.
6. Find information about tables using Operations Navigator.

Lab prerequisites

Before you begin this lab, ensure that you meet the following requirements:

- IBM iSeries server with V4R5M0 or higher
 - 5769-SS1 Host Servers
 - The latest PTFs should be installed
- Client Access Express V4R5M0
 - The latest Service Pack should be installed.

Time required

The approximate time required to complete this lab is 20 minutes.

Task 1: Getting connected

This task explains general navigation using Operations Navigator. The Operations Navigator icon (Figure 1) should appear on your desktop. If the Operations Navigator is not visible on your desktop, contact your network administrator to install Client Access.



Figure 1. The Operations Navigator icon

- ___ 1. Click the **Operations Navigator** icon on your desktop. If this is the first time you are running Operations Navigator, a message box appears stating that there are no connections to the iSeries server. Click **Yes** if you receive this box. Otherwise, proceed to step 5.
- ___ 2. The Add AS/400 Connection dialog appears. Obtain the system name from your network administrator to complete the next step. In the AS/400 System text box, enter the name of the iSeries server provided by the network administrator. Click **Next**.
- ___ 3. The AS/400 Sign on Information dialog appears. Select the **Use default User ID, prompt as needed** radio button. In the text box below, enter VETEAMXX. Click **Next**. The Verify Connection dialog appears.
- ___ 4. Click the **Verify Connection** button, and make sure that the host servers on the iSeries server are up and running. Click **Finish**.
- ___ 5. In the Operations Navigator window click the plus sign (+) next to My AS/400 Connections to show the iSeries servers that are available for use as shown in Figure 2. Click the iSeries server that you plan to use for this lab.

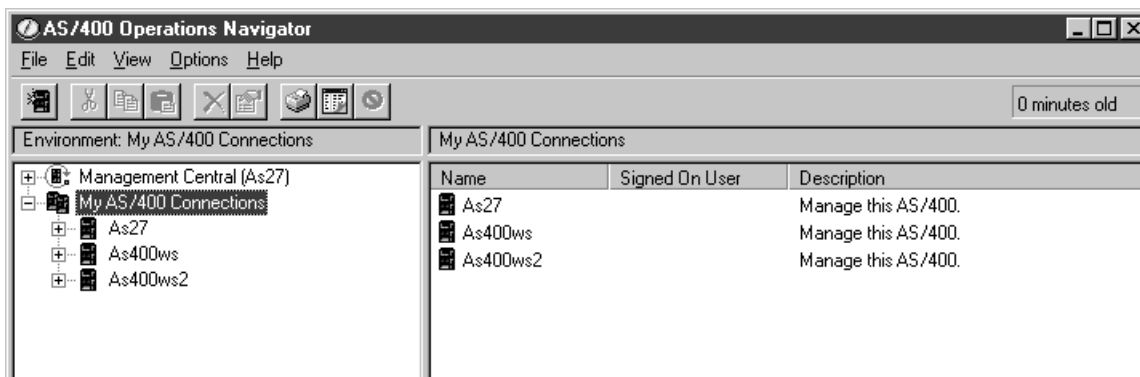


Figure 2. AS/400 connections

- ___ 6. A window prompts you to enter a user name and password as shown in Figure 3. Consult your lab worksheet for your profile name and password. When finished, click **OK**. If this is the first time Operations Navigator is being used on this iSeries server, the system updates Operations Navigator with its capabilities.

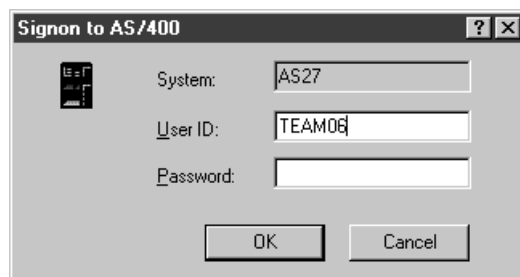


Figure 3. Signon to AS/400

- ___ 7. Click the plus sign (+) next to the Database object and click the **Database** object to expand the selection as shown in Figure 4. The display on the right shows the objects associated with the database object. The

associated objects include Libraries, Database Navigator and SQL Performance Monitors. Each object is described in these exercises.

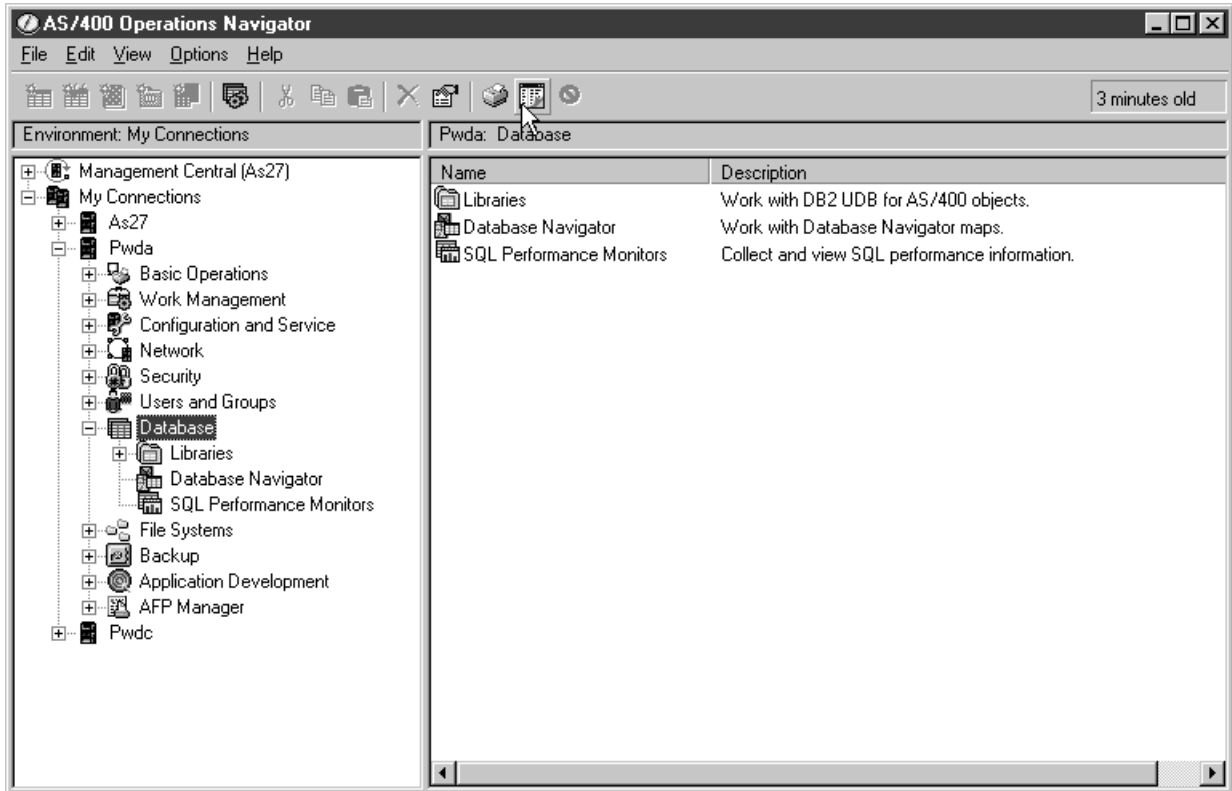


Figure 4. The Operations Navigator database selection

Note: In Operations Navigator, there are three places in which you can access functions for an object:

- The File drop-down menu
- An icon from the toolbar
- The right-click menu.

In the remaining labs, the right-click menu is used to access these functions.

Task 2: Database options

This task explains how to change the refresh option for the database objects. By default, the display is only refreshed when you click the Refresh icon on the toolbar or when an object is changed. The task also allows you to amend the list of libraries for the display.

1. Right-click the **Database** object as shown in Figure 5.

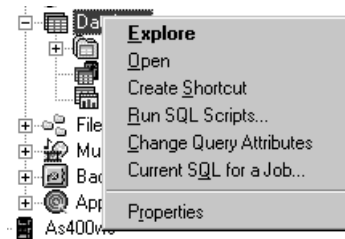


Figure 5. Right-click the database icon for the database options

- ___ 2. From the right-click menu, select **Properties**.
- ___ 3. Select the **Refresh contents every time list is displayed** check box.
- ___ 4. Click **OK**.

Note: Now, each time a list appears, the display automatically refreshes.

- ___ 5. Click the plus sign (+) next to the Libraries object.
- ___ 6. Right-click the **Libraries** object, and select **Select Libraries to Display** from the pull-down menu.
- ___ 7. In the library name box, type:

VETTEAMXX

- ___ 8. Click **Add**. Notice that the library is added to the list in the right window panel.
- ___ 9. Click **OK**. The display refreshes and the library is added to the display. Operations Navigator remembers this change, and the library will be included in this list on each subsequent execution.

Note: Changing the library here only changes the libraries that are displayed with Operations Navigator. It does not change the library list for the job.

Regular CL commands, such as ADDLIB and CHGCURLIB, are still used to achieve temporary library list changes. Regular or permanent changes involve altering job description or user profile objects. These tasks are normally carried out by the security officer or system administrator.

Task 3: Familiarization with the database

This exercise shows you how to find basic information on the tables that you use for the remainder of the labs. Understanding your data is a key requirement when you analyze database performance, regardless of the tool that you use for the analysis. The tables are all located in the library (collection) VETTEAMXX.

- ___ 1. Click the **Database** object under your AS/400 Connection.
- ___ 2. Click the plus sign (+) next to the **Libraries** object. You should see the libraries in your library list, and any libraries that were previously selected.
- ___ 3. Click the **VETTEAMXX** library for a list of tables on the right-hand side of the Operations Navigator window. If the library is not in the list, refer to Task 2 to include the library.
- ___ 4. Right-click any table. A pop-up menu appears that shows more options. The options are different between indexes, tables, and views as shown in Figure 6.

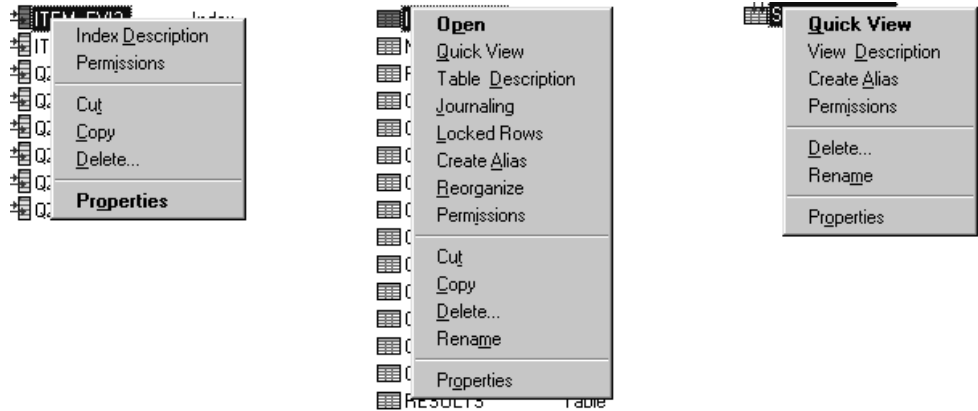


Figure 6. The pop-up menu for indices, tables, and views

___ 5. Right-click the **ITEM_FACT** table. Select the **Description**, **Permissions**, and **Properties** options to answer the questions in Table 1. The answers will not necessarily be on the first display. You will need to select the tabs to find all of the information.

Table 1. ITEM_FACT questions

Question	Answer
How many rows are in the table?	
How many deleted rows are in the table?	
Does the table re-use deleted rows?	
Who is the owner of the table?	
What level of authority does the public have?	
How many (if any) indexes does the table have?	

___ 6. Gather more information about your database to answer the questions in Table 2.

Table 2. Database information

Question	Answer
How many rows are in CUST_DIM?	
How many rows are in PART_DIM?	
How many rows are in SUPP_DIM?	

Task 4: SQL environment options

This exercise shows you how to update the SQL environment. The SQL environment is defined through a JDBC connection, which is created by the system. The first data source, QDSN_X (X is the iSeries server name), is used by Operations Navigator for communications to the iSeries server.

The second data source, QDS2_X (X is the iSeries server name), is used to communicate with the SQL scripting tool. This data source defines the SQL environment options.

1. Right-click the **Database** object under your iSeries server connection. Select **Run SQL Script**. The QDS2_X data source is created automatically. The Run SQL Script dialog appears as shown in Figure 7.

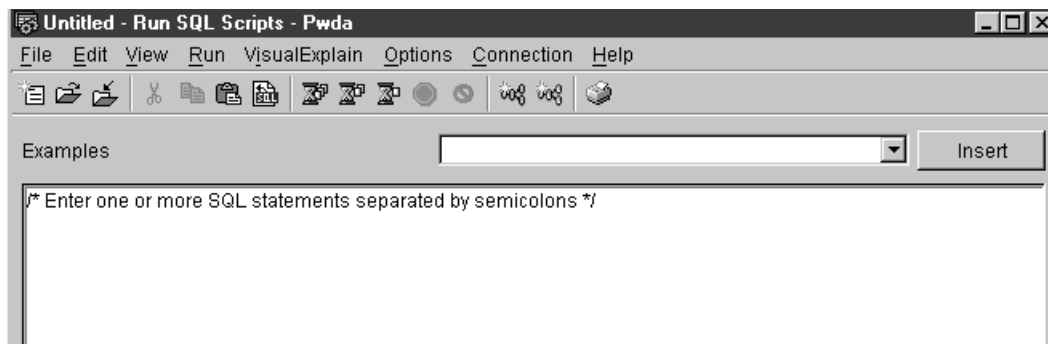


Figure 7. The Run SQL Script display

2. Select **Connections->JDBC Setup** as shown in Figure 8. The Client Access Express JDBC Setup dialog for the JDBC source QDS2_as400 appears.



Figure 8. Accessing the JDBC Setup options

3. Click the **Server** tab.
4. Verify that ***LIBL** is listed in the default libraries box. This allows you to access objects in all libraries in your user library list.
5. Change the naming convention for the SQL scripting to ***SYS**. Click the **Format** tab. To view the formatting options that apply to the JDBC data transfer.
6. Select **System naming convention (*SYS)** from the Naming Conventions drop-down menu as shown in Figure 9. This ensures that the list of libraries will be used and you will not have to enter qualified table names.

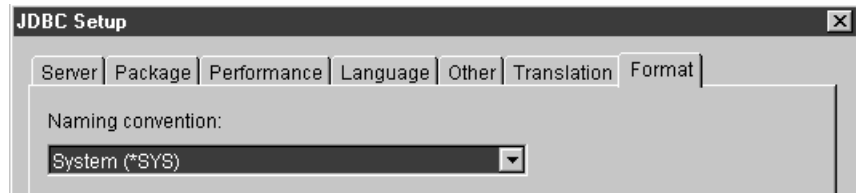


Figure 9. Selecting the System naming convention

__ 7. Click **OK**.

Since the VETEMXX library is on your user library list, and the naming convention is set to *SYS, you no longer need to specify qualified table names in your SQL statements when you use this ODBC data source. To verify this, in the statements box of the SQL scripting utility, type:

```
select * from cust_dim;
```

__ 8. From the toolbar, click the **Run All** icon as shown in Figure 10.

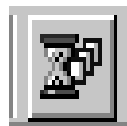


Figure 10. The Run All icon

__ 9. The results appear at the bottom of the window.

__ 10. Close the Run SQL Scripts window. Click **No** when you are asked to Save the changes.

Note: The SQL default name for a collection is the user name. This task can be omitted for applications where this is the case. The SQL naming convention is not often applied in an iSeries server environment.

Lab 2. An introduction to Visual Explain

You must complete all of the steps in Lab 1 before starting lab 2.

Introduction

The first lab introduced some general Operations Navigator tasks, the database functions within Operations Navigator, and the SQL Script window.

This lab introduces you to Visual Explain and compares its use with more traditional methods of database performance analysis. You should be familiar with the SQL Script Center within Operations Navigator before you attempt this lab.

Objectives

This lab teaches you how to:

1. Use debug to check the optimizer messages.
2. Use Visual Explain with SQL statements.
3. Use Visual Explain with performance data.

Lab prerequisites

Before you begin this lab, ensure that you have the following requirements:

- IBM iSeries server with V4R5M0 or higher
 - 5769-SS1 Host Servers
 - The latest PTFs should be installed.
- Client Access Express V4R5M0
 - The latest Service Pack should be installed.

Time Required

The time required to complete this lab is 30 minutes.

Task 1: Query Optimizer debug messages

Prior to the introduction of Visual Explain, the traditional way of determining the implementation of an SQL statement would be to analyze query optimizer debug messages. Before looking at the Visual Explain analysis, let's take a look at the debug messages so that you will be able to compare and contrast the two techniques.

This task shows you how to run an SQL script in debug mode and view the resulting messages in the job log. Debug mode messages may help to determine problems with SQL statements.

Note: Use the following path C:\VisualExplain\SQLScripts\Basic when asked to open an SQL script file.

- ___ 1. From Operations Navigator, right-click the **Database** icon, and select **Run SQL Scripts** to obtain the Run SQL Scripts window. Select the **Open** icon.
- ___ 2. Open the **Lab02 Task01.SQL** script to obtain the following query:

```

SELECT    Continent, Country, Count(*) as COUNT
FROM      Cust_Dim
GROUP BY  Continent, Country
ORDER BY  Continent, Country;

```

- ___ 3. Click the **Options** drop-down menu and select **Include Debug Messages in the Job Log** as shown in Figure 11 on page xiv. You may need authority (from the security officer) to access debug functions on the iSeries server. Also select **Run Statement on Double-Click**.

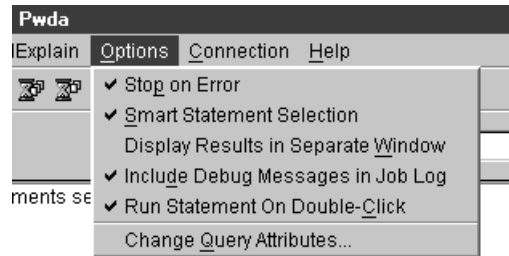


Figure 11. Run SQL Scripts showing debug options

- ___ 4. Click the **View** drop-down menu, and select **Job Log** as shown in Figure 12. A window prompts you to enter the user name and password. Use the user name and password that was assigned to you. The job log window appears. The most recent messages are at the top of the job log. The oldest message in the job log contains the job number.

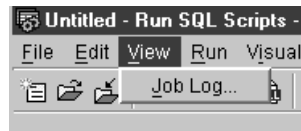


Figure 12. Run SQL Scripts Job Log options

- ___ 5. Run the SQL script by clicking the **Run All** icon.
- ___ 6. Close the Query Results viewer.
- ___ 7. Return to the job log window. Click the **Refresh** icon. Notice that a number of CPLxxxx (and sometimes, SQLxxxx) messages were logged during the execution of the SQL statement.
- ___ 8. What access method was chosen by the optimizer?_____
- ___ 9. Were any temporary objects used in this implementation?_____
- ___ 10.If so, what kind of objects?_____
- ___ 11.Close the job log window.
- ___ 12.Close the Run SQL Scripts window.

You have now completed this lab.

Task 2: Accessing Visual Explain

This task introduces Visual Explain. Visual Explain is an SQL performance analysis tool that assists you in the analysis of database performance on the iSeries server.

Visual Explain can be used with “live” SQL statements, as well as with SQL Performance Data that has been collected on the iSeries server. This exercise introduces both techniques.

This section of the lab helps you to navigate your way around Visual Explain and make the best use of its features. Subsequent labs deal with specific performance issues and interpreting the data.

___ 1. From Operations Navigator, right-click the **Database** icon, and select **Run SQL Scripts** to obtain the Run SQL Scripts window. Click the **Open** icon.

___ 2. Open the **Lab02 Task02.SQL** script to obtain the following query:

```
SELECT    Continent, Country, Count(*) as COUNT
FROM      Cust_Dim
GROUP BY  Continent, Country
ORDER BY  Continent, Country;
```

___ 3. Select the SQL statement (a single click is enough to select a statement; although, for clarity, a double-click highlights a statement).

___ 4. Select **Visual Explain** from the menu bar to locate the Visual Explain options as shown in Figure 13. The Explain option allows you to review an SQL statement, and the Run and Explain option allows you to run and review an SQL statement.

There are two icons for Visual Explain. The left icon (Figure 13) is to “Explain only”, and the right icon is to “Run and Explain”. These correspond to the menu options. Either method provides a Visual Explain graph.

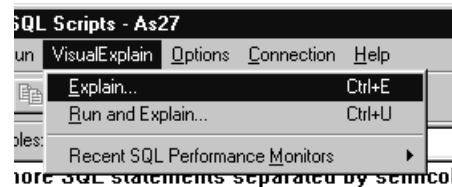


Figure 13. SQL Script Center Visual Explain options and icon

Choose the option to Run and Explain the SQL statement.

___ 5. There is a short delay. Visual Explain analyze the statement and presents the query results and its findings in new windows. The Visual Explain display has two sides: the Query Implementation Graph on the left and the Query Attributes and Values on the right. Analyze the graph. Does it make sense to you?

Note: The redpaper DB2 UDB for AS/400 Visual Explain - Illustrating the secrets that accompanies this workshop includes sample output from Visual Explain.

Click the **View** drop-down menu, and select **Overview** to obtain the overview shown in Figure 14 on page xvi. Move the cursor into the shaded

area of the Overview window representing the area on display. Drag the shaded area to the left side of the overview box.

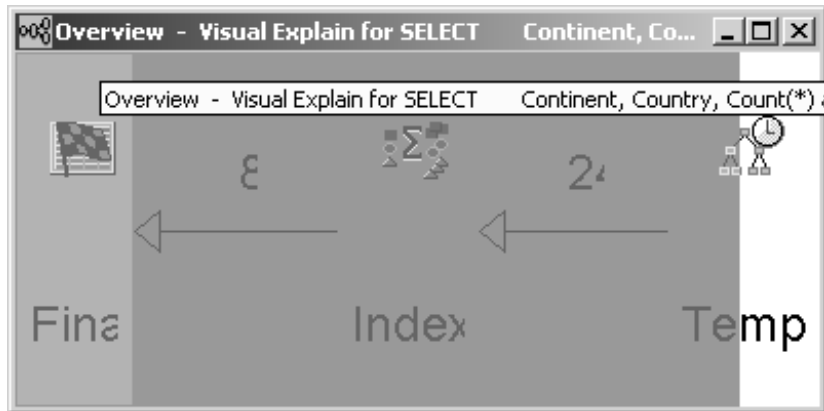


Figure 14. Visual Explain overview

- ___ 6. Close the Overview window.
- ___ 7. The finishing flag is now on the left of the Visual Explain Query Implementation Graph.

You may try the four rotational icons now (right-hand side of the tool bar) to see which style of presentation you prefer as shown in Figure 15. Each time you use Visual Explain, you will return to the default view.

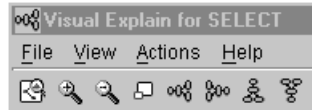


Figure 15. Visual Explain tool bar

- ___ 8. Position the cursor over the **Final Select** icon. A pop-up window appears. Each of the icons support a pop-up window.
- ___ 9. The Query Attributes and Values relate to the selected icon. Right-click the **Final Select** icon, and select **Environment Settings**. The work environment where the query was executed appears. This shows an example of how Visual Explain pulls everything together for query performance tuning.
- ___ 10. What access method was chosen by the optimizer? _____
- ___ 11. Were any temporary objects used in this implementation? _____
- ___ 12. If so, what kind of objects? _____
- ___ 13. Query optimizer debug messages and Visual Explain both allow you to determine the query implementation, but use different ways to represent the query implementation. Both tools continued to be used in the future. Close the Visual Explain window and the Query Results viewer.

Note: In V5R1 the optimizer messages are shown on the bottom of the Visual Explain window.

- ___ 14. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button.

Task 3: Visual Explain with Performance Monitor Data

Visual Explain can be used with “live” SQL statements, as well as with SQL Performance Data that has been collected on the iSeries server. This exercise illustrates how to use visual explain with Performance Monitor Data.

- ___ 1. Return to the SQL Scripts window..
- ___ 2. Open the **Lab02 Task03.SQL** script, and select the SQL statement:

```
SELECT    Custkey, phone, territory, salesperson
FROM      Cust_Dim
WHERE     country = 'ITALY' and region = 'CENTRAL' and
          MKTSEGMENT = 'BUILDING'
```

- ___ 3. Select **View->Job Log**. A window prompts you to enter the user name and password. Use the user name and password that was assigned to you. The job log for your current connection appears. Scroll down to the bottom of the job log, and find **CPF1124** message. Record the job data in Table 3.

Table 3. A record of the job number

Question	Answer
Job number	
User name	
Job name	

- ___ 4. Close the job log window.
- ___ 5. In the Operations Navigator window, select **SQL Performance Monitors** to display the monitors currently on the system.
- ___ 6. When you start the SQL Performance Monitor a new instance of the monitor is created on your iSeries server. Switch to the main Operations Navigator window. Start the SQL Performance Monitor. To do so, right-click **SQL Performance Monitors**. Select **New->Detailed** as shown in Figure 16.

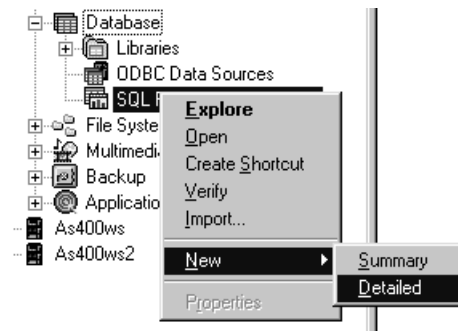


Figure 16. Starting the Memory Resident Database Monitor job

- ___ 7. The New SQL Performance Monitor dialog box appears. Click the **General** tab, and choose a name for the performance monitor that includes your VETTEAMXX profile. For the library name, select the **VETTEAMXX** library from the pull-down list.

- ___ 8. Now click the **Monitored Jobs** tab. Locate your job. The easiest way to find the job is to sort the list by Job Number. To do this, wait a few moments until the list has completed building and then click the **Number** column heading. Scroll down the list until you find your job. Highlight the job name, and click **Select** as shown in Figure 17.

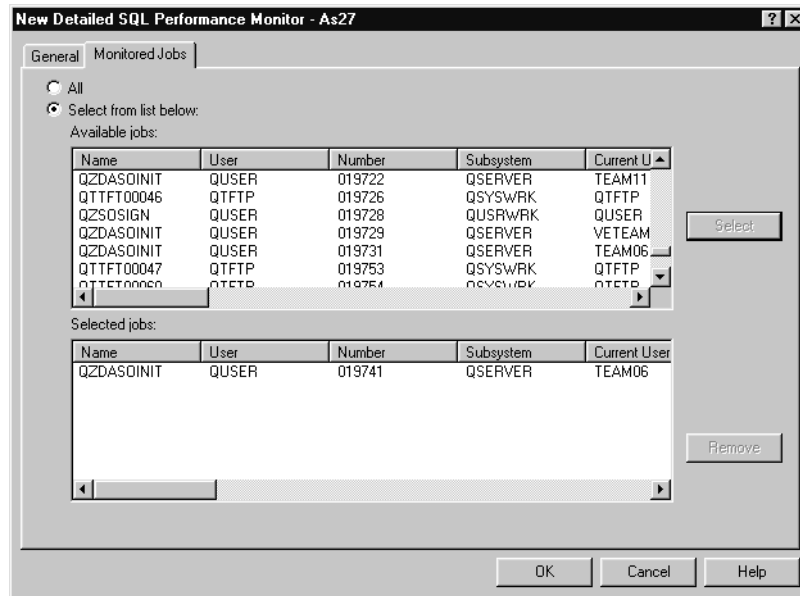


Figure 17. Select the job to monitor

- ___ 9. Click **OK** to start the SQL Performance Monitor. The monitor status should change to *Started* in the right panel of the Operations Navigator window.
- ___ 10. Return to the SQL Script window.
- ___ 11. Run the SQL statement.
- ___ 12. Unless there is a reason to continue collecting performance data, end the monitor once you collect the data that you need. Select **Operations Navigator->Database->SQL Performance Monitors** to obtain a list of the SQL Performance Monitors that are currently active.
- ___ 13. Right-click the SQL Performance Monitor that you created in the previous step. Select **End** as shown in Figure 18.

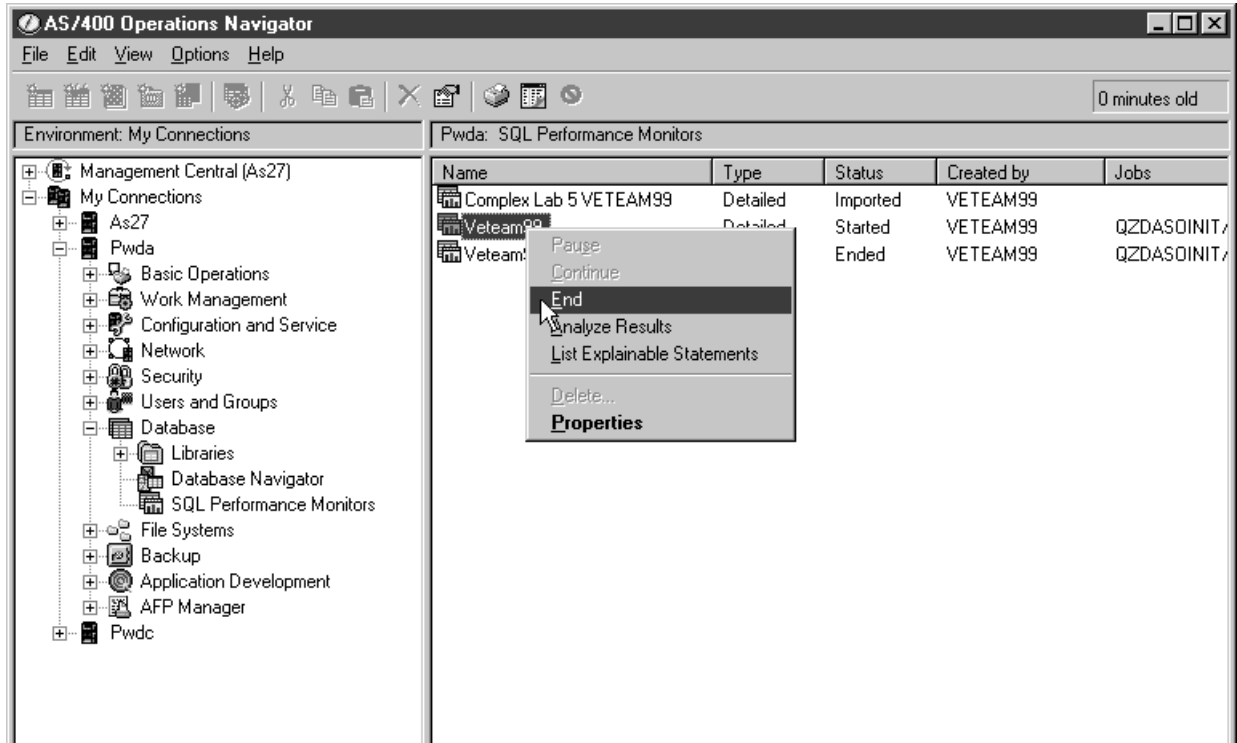


Figure 18. SQL Performance Monitors

- 14. Right-click the performance monitor again, and select **List Explainable Statements**. You should see the SQL statement from the SQL Script window as shown in Figure 19.

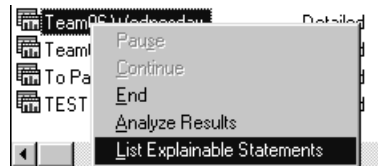


Figure 19. Selecting statements for Visual Explain

- 15. Select the SQL statement, and click the **Run Visual Explain** icon as shown in Figure 20.

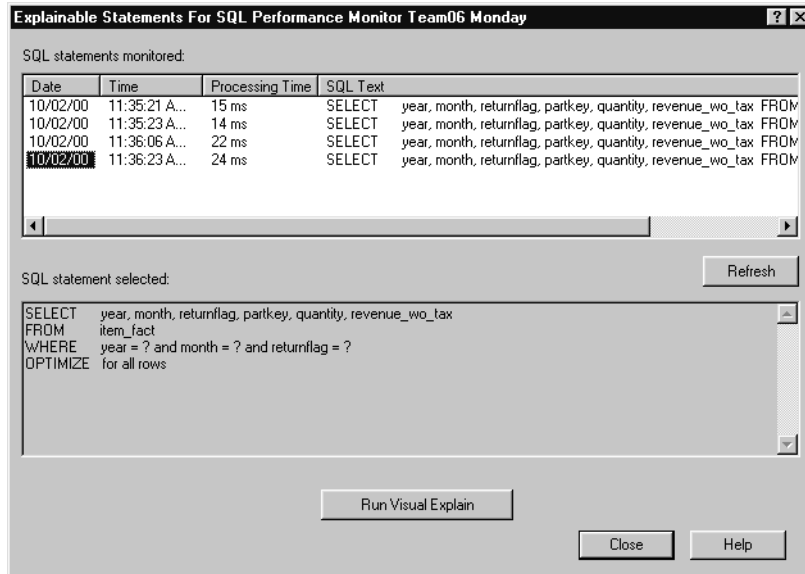


Figure 20. Visual Explain from the SQL Performance Monitor

___ 16. The Visual Explain Query Implementation Graph and the Query Attributes and Values appear. Answer the questions in Table 4 from this display.

Table 4. Table Scan questions

Question	Answer
How many rows were estimated by the Table Scan?	
What is the number of rows in the table?	

___ 17. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button

___ 18. Close the Run SQL Scripts window.

___ 19. Close the Explainable Statements for SQL Performance Monitor window.

___ 20. Unless you need to retain the performance data, remove it from the system. Select **Operations Navigator->Database->SQL Performance Monitors** to obtain the list of SQL Performance Monitors.

___ 21. Right-click the SQL Performance Monitor that you created earlier and select **Delete**.

___ 22. Verify that you have selected the correct monitor, and Confirm the delete.

You have completed this task.

Lab 3. Non-keyed access methods

Client Access Express V4R5M0 introduces a new graphical interface to analyze the performance of queries running on the iSeries server. Any SQL statement that is processed by the query optimizer on the iSeries server can be analyzed, either interactively or by using the performance monitoring tools.

Labs 1 and 2 provide the preparation for this lab. You must complete them before you start this lab.

Introduction

This lab explains how to use Visual Explain for queries that use the table scan access method. The table scan methods that are simulated are a full table scan, a parallel table scan, a skip sequential table scan, and a parallel skip sequential table scan. The IBM Learning Services course *S6140 DB2 UDB for AS/400 SQL & Query Performance Tuning and Monitoring Workshop* includes a full discussion of table scan access methods. Also refer to the manual *DB2 UDB for AS/400 Database Performance and Query Optimization*, which can be accessed online at: <http://publib.boulder.ibm.com/html/as400/v4r5/ic2924/info/index.htm>

At the site, in the left panel, click **Database and File Systems**. Click **DB2 Universal Database for AS/400**, and then click **Books (PDF format)**. Choose whether to view the HTML version online or download the PDF.

Objectives

Upon completing this lab, you will be able to use Visual Explain to:

1. Identify and analyze a table scan.
2. Identify and analyze a parallel table scan.
3. Identify and analyze a skip sequential table scan.
4. Identify and analyze a parallel skip sequential table scan.
5. Assist your table indexing strategy.

Lab prerequisites

Before you begin this lab, ensure that you have the following requirements:

- IBM iSeries server with V4R5M0 or higher
 - 5769-SS1 Host Servers
 - The latest PTFs should be installed.
- Client Access Express V4R5M0
 - The latest Service Pack should be installed.

Lab environment

This lab uses query optimization to perform parallel processing. Lab tests were conducted on an iSeries server with more than one processor and symmetric multiprocessing installed. Different results may be obtained in different environments.

Note: Query runtime depends on many factors, for example, the number of other jobs running on the system and their impact on CPU, DASD, and memory utilization.

Time required

The time required to efficiently complete this lab is 45 min.

Task 1: Table scan

Complete the task, which explains how to run a table scan query. This example will help you to complete the questions at the end of the task. The script that is provided sets the query attributes and runs an SQL statement designed to provide a table scan.

___ 1. From Operations Navigator, right-click the **Database** icon, and select **Run SQL Scripts** to view the Run SQL Scripts window. Select the **Open** icon.

___ 2. Open the **Lab03 Task01.SQL** script:

```
CL:      chgqrya degree(*none);
SELECT  year, month, returnflag, partkey, quantity,
        revenue_wo_tax
FROM    item_fact
WHERE   year = 2002 and month = 6 and returnflag = 'R';
```

___ 3. Click the **Run Selected** icon to run the CL command.

___ 4. Click the **Run and Explain** icon to run the SQL statement.

___ 5. Verify that the display that you received has the same icons as those shown in Figure 21. The query optimizer may provide a different access plan on some systems.

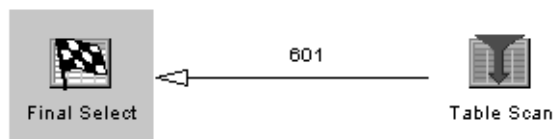


Figure 21. Table scan

___ 6. Using the information presented by Visual Explain, complete the questions in Table 5.

Table 5. Table scan questions

Question	Answer
What is the name of the table being queried?	
How many rows in the table?	
What is the number of estimated rows that are selected?	
Is an index advised?	
If so, what keys is the optimizer suggesting?	

- ___ 7. Review the remaining information, and close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button.

Task 2: Parallel table scan

The following task explains how to run a parallel table scan query. Complete this task, which will help you to complete the questions at the end of the task. The script that is provided sets the query attributes and runs an SQL statement designed to provide a parallel table scan.

- ___ 1. From the Run SQL Scripts window, select the **Open** icon.
- ___ 2. Open the **Lab03 Task02.SQL** script:

```
CL:      chgqrya degree(*optimize);
SELECT  year, month, returnflag, partkey, quantity,
        revenue_wo_tax
FROM     item_fact
WHERE    year = 2002 and month = 6 and returnflag = 'R'
OPTIMIZE for all rows;
```

- ___ 3. Click the **Run Selected** icon to run the CL command.
- ___ 4. Click the **Run and Explain** icon to run the SQL statement.
- ___ 5. Verify that the display that you received resembles the example shown in Figure 22. The query optimizer may provide a different access plan on some systems.

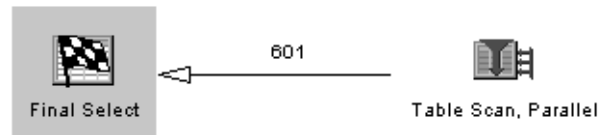


Figure 22. Parallel table scan

- ___ 6. After you select the icons presented by Visual Explain, complete the questions in Table 6.

Table 6. Parallel table scan questions

Question	Answer
What reason was given for selecting a table scan?	
Was parallel processing used?	
If so, what degree was requested?	
Was either parallel pre-fetch or parallel pre-load processing used?	
Which memory pool was used for the query?	
How much memory was available for the query?	

- ___ 7. Review the remaining information and close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button

This completes Task 2. You may continue with Task 3, or close the Operations Navigator window and resume later.

Task 3: Skip sequential table scan

This task shows you how to run a skip sequential table scan query so you can complete the questions at the end of the task. The script that is provided sets the query attributes and creates three encoded vector indexes prior to running an SQL statement designed to provide a skip sequential table scan.

- ___ 1. From the Run SQL Scripts window, select the **Open** icon.
- ___ 2. Open the **Lab03 Task03.SQL** script:

```
CL:      chgqrya degree(*none);
CREATE   encoded vector index item_evi1 on item_fact
         (year) with 3 distinct values;
CREATE   encoded vector index item_evi2 on item_fact
         (month) with 12 distinct values;
CREATE   encoded vector index item_evi3 on item_fact
         (returnflag) with 3 distinct values;
SELECT   year, month, returnflag, partkey, quantity,
         revenue_wo_tax
FROM     item_fact
WHERE    year = 2002 and month = 6 and returnflag = 'R';
```

- ___ 3. Click the **Run Selected** icon to run the CL command.
- ___ 4. Click the **Run Selected** icon again to run the first SQL command (to create the index item_evi1). This command may take awhile to complete.
- ___ 5. Click the **Run Selected** icon again to run the second SQL command (to create the index item_evi2). Again, this command may take awhile to complete.
- ___ 6. Click the **Run Selected** icon again to run the third SQL command (to create the index item_evi3). Once again, this command may take awhile to complete.
- ___ 7. Click the **Run and Explain** icon to run the SQL statement.
- ___ 8. Verify that the display that you received resembles the example shown in Figure 23. The query optimizer may provide a different access plan on some systems.

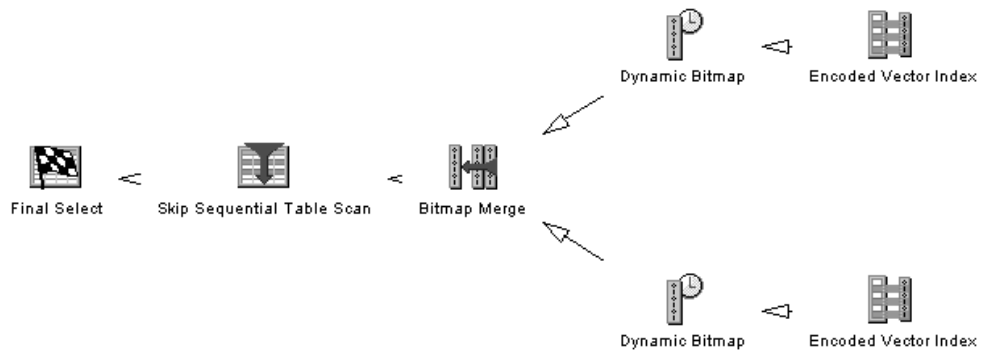


Figure 23. Skip sequential table scan

___ 9. Select the icons presented by Visual Explain, and complete the questions in Table 7. You need to use both Encoded Vector Index icons for the first question. Use either icon for the remaining questions.

Table 7. Skip sequential table scan questions

Question	Answer
Determine which indexes were used in the bit map creation by finding the indexes that are linked to the Dynamic Bitmap Icon	
Which indexes were considered by the optimizer?	
Which index was rejected?	
What was the reason code for rejection?	

___ 10. Select the **Skip Sequential Table Scan** icon and the **Final Results** icon, and answer the questions in Table 8.

Table 8. Further skip sequential table scan questions

Question	Answer
What reason was given for selecting a table scan?	
Will the table being scanned fit in main memory?	
Is an index advised?	
If so, what keys is the optimizer suggesting?	

___ 11. Review the remaining information, and close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button

This completes Task 3. You may continue with Task 4, or close the Operations Navigator window and resume later.

Task 4: Comparing the job log with Visual Explain (Optional)

In Task 4, you replicate the queries that were executed during the previous tasks to show table scan methods. You collect the SQL Performance data and job log debug information for these queries. You use Visual Explain on the performance data and compare the results with the job log messages.

The statements to do this task are contained in a single script. This script deletes the encoded vector indexes created during the previous labs to recreate the initial environment. It then runs each script statement in the same sequence and with the same query attributes.

If you use the Run All icon in the SQL Script window, it submits all of the tasks to the iSeries server to be executed. As the iSeries server multi-tasks, statement execution overlaps, and the statements compete with each other. For this reason, you need to run each statement individually.

- ___ 1. From Operations Navigator, right-click the **Database** icon and select **Run SQL Scripts** to obtain the Run SQL Scripts window.
- ___ 2. From the Run SQL Scripts window, select the **Open** icon.
- ___ 3. Open the **Lab03 Task04.SQL** script:

```
DROP          index item_evi1;
DROP          index item_evi2;
DROP          index item_evi3;

CL:           chgqrya degree(*none);

SELECT        year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM          item_fact
WHERE         year = 2002 and month = 6 and returnflag = 'R';

CL:           chgqrya degree(*optimize);

SELECT        year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM          item_fact
WHERE         year = 2002 and month = 6 and returnflag = 'R'
OPTIMIZE     for all rows;

CL:           chgqrya degree(*none);

CREATE        encoded vector index item_evi1 on item_fact (year) with 3
distinct values;
CREATE        encoded vector index item_evi2 on item_fact (month) with 12
distinct values;
CREATE        encoded vector index item_evi3 on item_fact (returnflag) with
3 distinct values;

SELECT        year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM          item_fact
WHERE         year = 2002 and month = 6 and returnflag = 'R';

CL:           chgqrya degree(*optimize);
```

```

CREATE      encoded vector index item_evi4 on item_fact (year, month,
returnflag);

SELECT      year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM        item_fact
WHERE       year = 2002 and month = 6 and returnflag = 'R'
OPTIMIZE   for all rows;

DROP        index item_evi1;
DROP        index item_evi2;
DROP        index item_evi3;
DROP        index item_evi4;

-- THIS IS THE END OF THE SCRIPT.

```

- ___ 4. From the Run SQL Scripts menu bar, select **Options->Include Debug Messages in Job Log**.
- ___ 5. Select **View->Job Log**.
- ___ 6. Click the **Run Selected** icon to run each SQL statement and CL command individually, until the final command in the script has completed.
- ___ 7. For each SQL select statement, a window containing the selected rows appears. Close each of these windows as they appear.
- ___ 8. Select the Job Log window that you opened earlier in this lab. Click the **Refresh** icon from the tool bar. You should find a set of four SQL7953 messages at the top of the job log. Scroll down until you find the next three SQL7953 messages toward the bottom of the job log. These correspond to the beginning of the SQL script.
- ___ 9. The PREPARE of statement QZ... message (SQL7967) prior to the SQL7953 drop of ITEM_EVI4 message is the first debug message for the first query.
- ___ 10. The messages that may be most interesting to you are CPI433A, CPI4329, CPI432F, SQL7912, SQL7913 and SQL7963. Double-click the **CPI4329** message for more information. This confirms that a table scan has taken place as shown in Figure 24.

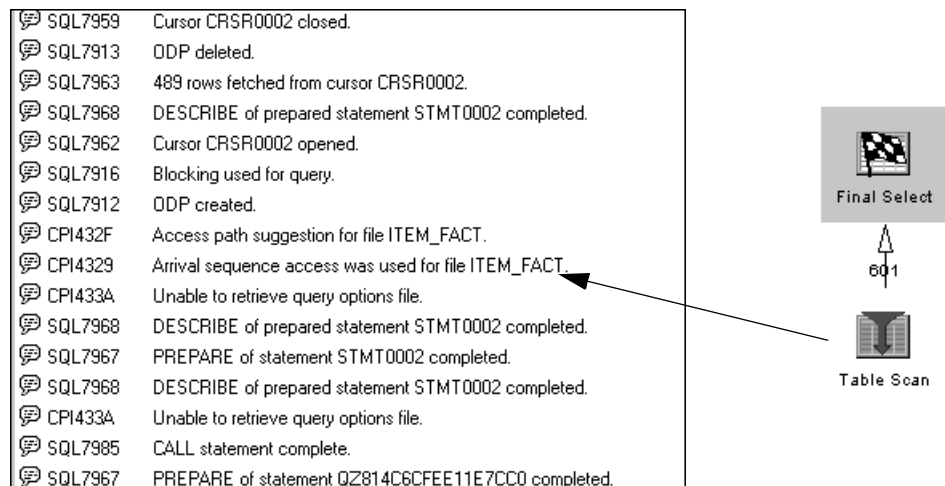


Figure 24. Table scan job log entries and the equivalent Visual Explain image

- ___ 11. Double-click the **CPI432F** message. This message advises the creation of an index. Locate the SQL7963 message in the job log for the number of rows selected.
- ___ 12. Scroll up the job log until you find the next SQL7967 PREPARE statement, which marks the beginning of the next query.
- ___ 13. The same messages are evident, with the addition of the CPI4330 message as shown in Figure 25.

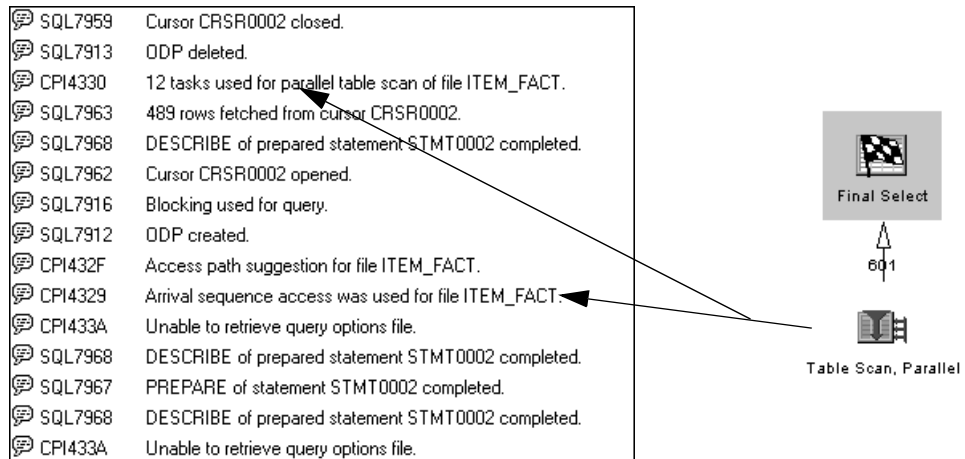


Figure 25. Parallel table scan and Visual Explain image

- ___ 14. Double-click the **CPI4330** message to find more information about the parallel table scan that has taken place.
- ___ 15. Scroll up the job log until you find the SQL7967 PREPARE statement, which denotes the beginning of the next query.

Note: The next prepare was completed before the encoded vector indexes were built. Scroll up one or two more lines so that you can see all of the messages for this query.
- ___ 16. The significant messages are: CPI4323, CPI433A, CPI4329, CPI432C, CPI432F, CPI4338, SQL7912, SQL7913, and SQL7963. Compare the job log and the Visual Explain graph in Figure 26.

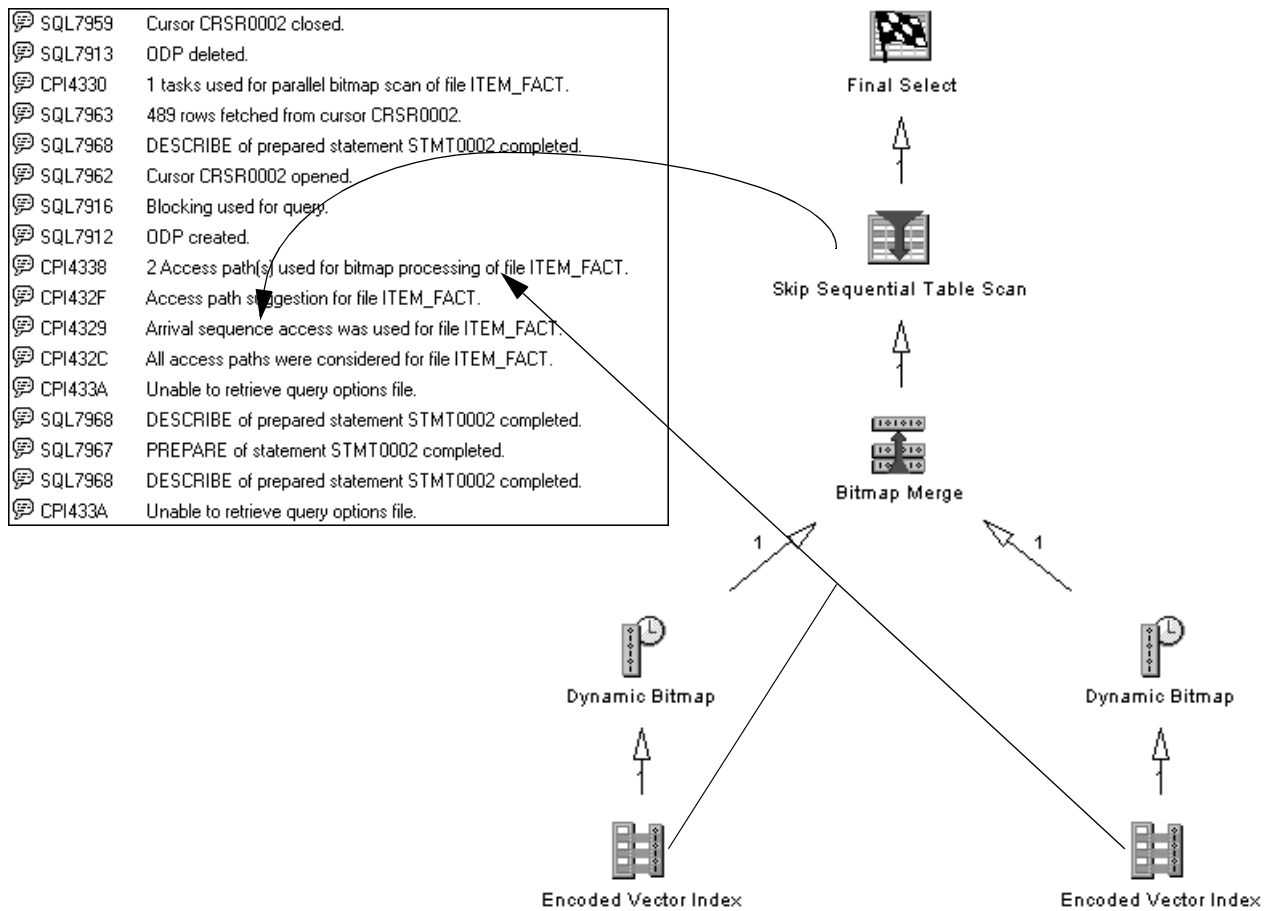


Figure 26. Skip sequential table scan

- ___ 17. Check messages CPI4323, CPI432F, and CPI4338.
- ___ 18. Scroll up the job log until you find the SQL7967 PREPARE statement, which marks the beginning of the final query.
- ___ 19. The significant messages are: CPI4323, CPI433A, CPI4329, CPI432C, CPI432F, CPI4330, CPI4338, SQL7912, SQL7913, and SQL7969. Compare the job log and the Visual Explain graph in Figure 27.

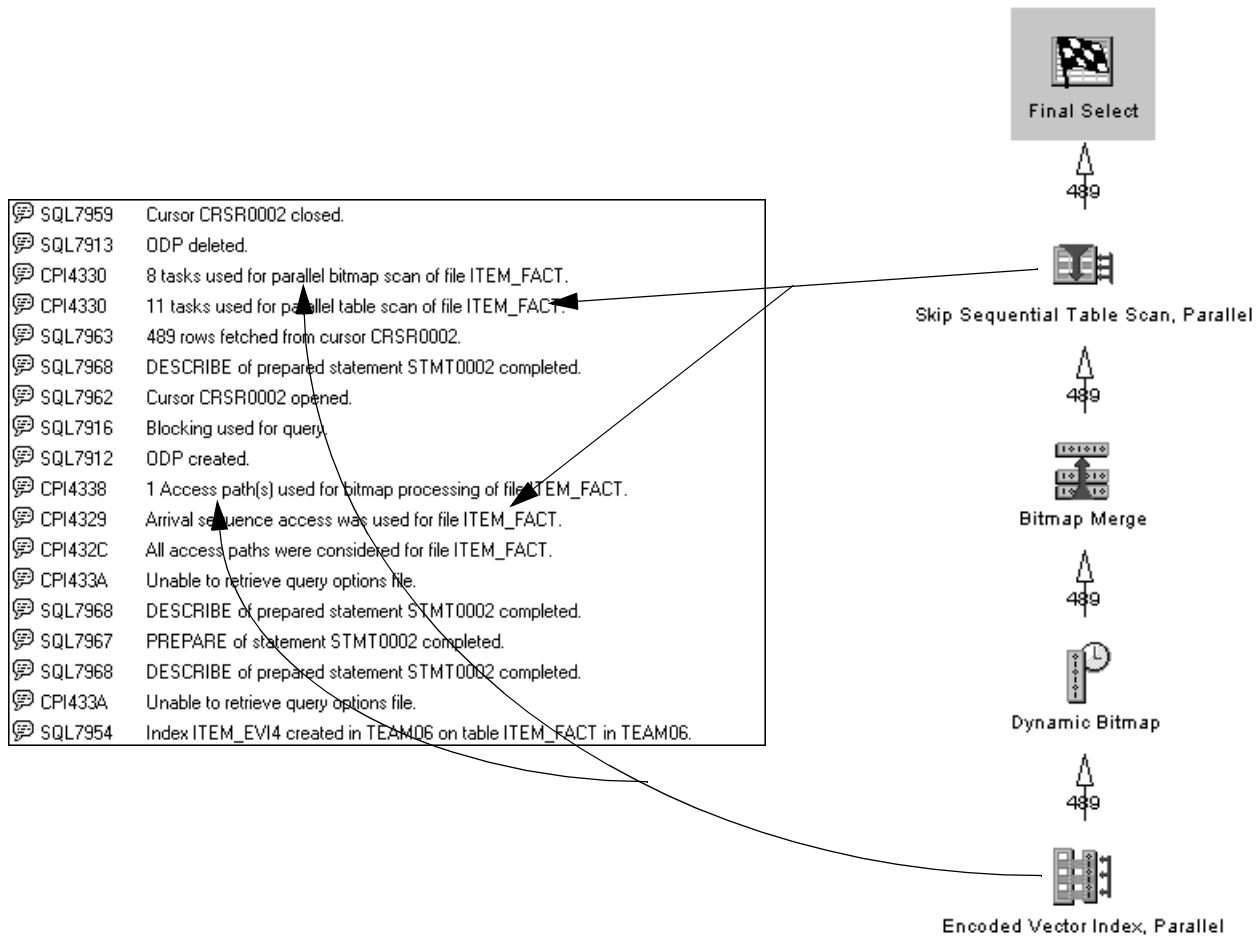


Figure 27. Parallel Skip Sequential job log and Visual Explain image

- ___ 20. Check messages CPI4323, CPI432F, CPI4330, and CPI4338.
- ___ 21. Close the Job Log window.
- ___ 22. Close the Run SQL Scripts window.

Task 5: SQL statements

The SQL statements that are used during Lab 4 are repeated here for convenience:

```

DROP          index item_evi1;
DROP          index item_evi2;
DROP          index item_evi3;
DROP          index item_evi4;

CL:          chgqrya degree(*none);

SELECT       year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM         item_fact
WHERE        year = 2002 and month = 6 and returnflag = 'R';

CL:          chgqrya degree(*optimize);

SELECT       year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM         item_fact
WHERE        year = 2002 and month = 6 and returnflag = 'R'

```

```

OPTIMIZE      for all rows;

CL:          chgqrya degree(*none);

CREATE        encoded vector index item_evi1 on item_fact (year) with 3
              distinct values;
CREATE        encoded vector index item_evi2 on item_fact (month) with 12
              distinct values;
CREATE        encoded vector index item_evi3 on item_fact (returnflag)
              with 3 distinct values;

SELECT        year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM          item_fact
WHERE         year = 2002 and month = 6 and returnflag = 'R';

CL:          chgqrya degree(*optimize);

CREATE        encoded vector index item_evi4 on item_fact (year, month,
              returnflag);

SELECT        year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM          item_fact
WHERE         year = 2002 and month = 6 and returnflag = 'R'
OPTIMIZE      for all rows;

DROP          index item_evi1;
DROP          index item_evi2;
DROP          index item_evi3;
DROP          index item_evi4;

```

Lab 4. Keyed access methods

Introduction

This lab reviews the use of Visual Explain to analyze keyed access methods.

Objectives

This lab teaches you how to:

1. Interpret Visual Explain graphs and attributes for keyed access methods
2. Create required indexes
3. Interpret job log messages to determine the access methods used

Lab prerequisites

Before you begin this lab, ensure that you have the following requirements:

- IBM iSeries server with V4R5M0 or higher
 - 5769-SS1 Host Servers
 - The latest PTFs should be installed.
- Client Access Express V4R5M0
 - The latest Service Pack should be installed.

Lab environment

Query runtime depends on several factors, for example, the number of other jobs running on the system and their impact on CPU, DASD, and memory utilization.

Disclaimer

This workshop describes query implementations observed in a laboratory environment. Depending on the hardware configuration of your machine and its current workload, your query may require a different implementation plan. While creating an access plan for a query, the Query Optimizer takes into account many factors, which are machine specific.

The results contained in this publication have not been submitted to any formal IBM test and are distributed as *is*.

Time required

The time required to efficiently complete this lab project is 25 minutes.

Task 1: Using Visual Explain to show keyed access methods

This exercise shows you the results of a keyed access method as viewed by Visual Explain.

- ___ 1. Before you start be sure to delete or drop the following indexes: item_evi1, item_evi2, item_evi3 and item_evi4.
- ___ 2. From Operations Navigator, right-click the **Database** icon and select **Run SQL Scripts** to obtain the Run SQL Scripts window. Select the **Open** icon.

- ___ 3. Open the **Lab04 Task01.SQL** script.
- ___ 4. Amend the CL command in the script to apply the *OPTIMIZE value for the query processing degree. Run the CL command.

```
CL: CHGQRYA DEGREE(*OPTIMIZE);
```
- ___ 5. Amend the SELECT statement in the script by modifying the local selection with the values highlighted in the following script:

```
SELECT  a.year, a.month, a.returnflag, a.partkey,
        a.quantity, a.revenue_wo_tax
FROM    item_fact a
WHERE   a.year = 2000 AND a.month = 10 AND
        a.returnflag = 'R'
OPTIMIZE FOR ALL ROWS;
```
- ___ 6. Click the **Run and Explain** icon to execute the revised SQL statement.
- ___ 7. Verify that the display that you receive has the same icons as those shown in Figure 28. The Query Optimizer may provide a different access plan on some systems.

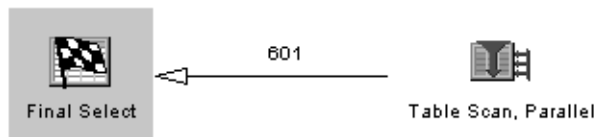


Figure 28. Parallel table scan

- ___ 8. Analyze the query implementation by reviewing the Visual Explain window, and complete Table 9.

Table 9. Query implementation questions

Question	Answer
Was the creation of an index advised?	
What is the list of key columns for the advised index?	
Over which table was the index advised?	

- ___ 9. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button
- ___ 10. Amend the CREATE INDEX statement in the SQL script window to build the advised index using the attributes you collected during the analysis:

```
CREATE  index item_idx1 ON item_fact
        (year, month, returnflag);
```
- ___ 11. **Run** the CREATE INDEX statement and verify that it completed successfully.
- ___ 12. **Run and Explain** the SELECT statement that now becomes highlighted.
- ___ 13. Verify that the display that you received has the same icons as those shown in Figure 29. It is possible that the Query Optimizer will provide a different access plan on some systems.

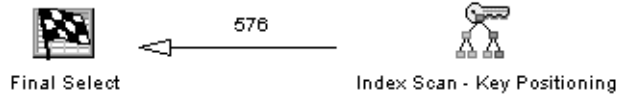


Figure 29. Key positioning

__ 14. Review the Visual Explain window, and complete Table 10.

Table 10. Keyed access questions

Question	Answer
Which index was used?	
What access method was used?	
What type of index was used?	
What is the index size (in bytes)?	
What is the index page size (in bytes)?	
What is the reason code given for using this index?	
Did the optimizer advise the creation of an index?	

__ 15. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button

Task 2: Additional keyed access methods explained

In this lab, you use Visual Explain to show the attributes of some additional keyed access methods.

__ 1. Amend the next CREATE SQL statement in the script to create a radix index (ITEM_IDX2) over the ITEM_FACT table for the *partkey* and *month* columns.

```
CREATE INDEX ITEM_IDX2 ON ITEM_FACT (partkey, month);
```

__ 2. Amend the CL command in the script to apply the *NONE value for the query processing degree. Run the CL command.

```
CL: CHGQRYA DEGREE(*NONE);
```

__ 3. Run the statement and verify that it completed successfully.

__ 4. Amend the next SELECT statement in the script with the values highlighted here:

```
SELECT partkey, month
FROM item_fact
WHERE partkey BETWEEN 99 AND 1234
AND month IN (03, 05, 07)
OPTIMIZE FOR ALL ROWS;
```

__ 5. Run and explain the statement.

- ___ 6. Verify that the display you receive has the same icons as those shown in Figure 30. The Query Optimizer may provide a different access plan on some systems.

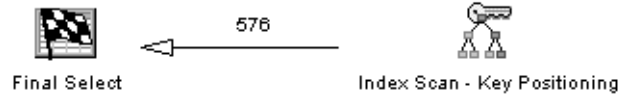


Figure 30. Index scan - Key Positioning

- ___ 7. Review the Visual Explain window, and complete Table 11.

Table 11. Key selection questions

Question	Answer
What access methods were used?	
What columns were used for key selection?	
Was index only access used?	

- ___ 8. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button.

Task 3: Temporary Index

The Query Optimizer occasionally creates a Temporary Index to assist with the access plan. This lab shows the Visual Explain icons where a Temporary Index is used to assist the implementation of the query.

- ___ 1. Amend the next SELECT statement in the script by inserting the ORDER BY statement highlighted here:

```
SELECT    year, month, returnflag, partkey, quantity,
          revenue_wo_tax
FROM      item_fact
ORDER BY  quantity
OPTIMIZE  FOR 1 ROW;
```

- ___ 2. Run and explain the statement.
- ___ 3. Verify that the display you receive has the same icons as those shown in Figure 31. The Query Optimizer may provide a different access plan on some systems.

Note: Figure 31 is shown without the full directional arrows and estimated records to allow the graphics to be represented here.

Note: When analyzing temporary index creation with Visual Explain, you probably want to ignore any table scan icons or index selection icons that are linked to the temporary index icon (if you're analyzing from left to right, these extra icons are to the right of the temporary index icon). These extra table and index methods icons can be confusing since these methods are NOT used in the actual implementation of the query - these methods are only used in the creation of the temporary index. Thus, the extra icons are primarily for analysis by the IBM development team. The same information

(regarding if the temporary index is created directly from a table or another index) is found in the Query Attributes window associated with the Temporary Index icon.

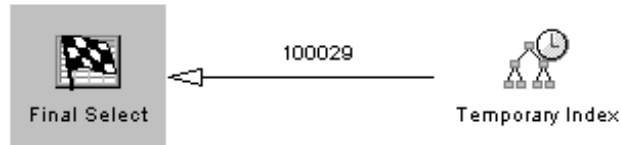


Figure 31. Temporary Index

- ___ 4. Review the Visual Explain window. Click the **Temporary Index** icon, and complete Table 12.

Table 12. Temporary Index questions

Question	Answer
Over which file was the temporary index built?	
How many entries are in the created index?	
Is this a sparse index?	
What type of index was created?	
Why was the index created?	
What columns were used for the index key?	

- ___ 5. Right-click the **Temporary Index** icon. Select the **Create Index** menu option.
- ___ 6. In the New Index on Table window that appears, the Quantity column should be selected for the key to a new index for this file.
- ___ 7. Enter the ITEM_IDX3 file name in the Index box, enter your library name and select Not Unique for index type then click **OK**. There is a delay while the index is created.
- ___ 8. Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button.
- ___ 9. Explain the same query again now that the index has been created.
- ___ 10. Will your new index be used by the query? Close the Visual Explain window. A window appears asking if you wish to save this data as a performance monitor. Click the **No** button.
- ___ 11. Close the SQL Script Center window. Choose **No** when prompted to save the changes.

Task 4: Comparison of Visual Explain and job log messages (Optional)

Task 4 replicates the queries processed so far in this lab and enables you to compare the Visual Explain results to the job log. The process to achieve this follows that of the previous lab.

Therefore, you need to monitor the detailed job log messages and start an SQL Performance Monitor for the job. Once this is done, you run the SQL statements again, one at a time, and end the performance monitor when completed. You then analyze the data and compare it to the job log messages.

- ___ 1. From Operations Navigator, right-click the **Database** icon, and select **Run SQL Scripts** to obtain the Run SQL Scripts window.
- ___ 2. From the Run SQL Scripts window, open the **Lab04 Task02.SQL** script.

```
-- Visual Explain Part 1 Lab 4.
```

```
DROP INDEX Item_idx1;
DROP INDEX item_idx2;
DROP INDEX item_idx3;
```

```
CL:CHGQRYA DEGREE(*optimize);
```

```
SELECT a.year, a.month, a.returnflag, a.partkey, a.quantity,
a.revenue_wo_tax
FROM item_fact a
WHERE a.year = 2000 AND a.month = 10 AND a.returnflag = 'R'
OPTIMIZE FOR ALL ROWS;
```

```
CREATE INDEX item_idx1 ON item_fact (year, month, returnflag);
```

```
SELECT a.year, a.month, a.returnflag, a.partkey, a.quantity,
a.revenue_wo_tax
FROM item_fact a
WHERE a.year = 2000 AND a.month = 10 AND a.returnflag = 'R'
OPTIMIZE FOR ALL ROWS;
```

```
CREATE INDEX item_idx2 ON item_fact (partkey, month);
```

```
SELECT partkey, month
FROM item_fact
WHERE partkey BETWEEN 99 AND 1234
AND month IN (03, 05, 07)
OPTIMIZE FOR ALL ROWS;
```


```
SELECT          year, month, returnflag, partkey, quantity, revenue_wo_tax
FROM            item_fact
ORDER BY      quantity
OPTIMIZE       FOR 1 ROW;
```

```
-- THIS IS THE END OF THE SCRIPT.
```

- ___ 3. From the Run SQL Scripts menu bar, select **Options->Include Debug Messages in Job Log**.
- ___ 4. Select **View->Job Log**. The job log for your current connection appears.

- ___ 5. Click the **Run Selected** icon to run each SQL statement and CL command individually until the final command in the script has completed.
- ___ 6. For each SQL select statement, a window containing the selected rows appears. Close each window as it appears.
- ___ 7. Select the Job Log window that you opened earlier. Click the **Refresh** icon from the toolbar. Scroll down until you find three SQL7953 messages near the bottom of the job log. This is the beginning of the SQL script.
- ___ 8. Scroll up the job log and find the messages that correspond to the first query, a table scan. Scroll up further and find the SQL7954 message that was issued when the CREATE INDEX... statement was executed. The messages that are those from the index scan.
- ___ 9. Select the Job Log window. You should see messages similar to those shown in Figure 32.

Message ID	Message
SQL7959	Cursor CRSR0002 closed.
SQL7913	ODP deleted.
SQL7963	576 rows fetched from cursor CRSR0002.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7962	Cursor CRSR0002 opened.
SQL7916	Blocking used for query.
SQL7912	ODP created.
CPI4328	Access path of file ITEM_IDX1 was used by query.
CPI433A	Unable to retrieve query options file.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7967	PREPARE of statement STMT0002 completed.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
CPI433A	Unable to retrieve query options file.
SQL7954	Index ITEM_IDX1 created in TEAM06 on table ITEM_FACT in TEAM06.




Final Select

↑

576

↓



Index Scan - Key Positioning




Figure 32. Key row positioning

- ___ 10. Review the job log messages.
- ___ 11. In the Job Log window, scroll up to find the SQL7954 message relating to the second index, ITEM_IDX2. The messages above this are for the final query.

Message ID	Message
SQL7959	Cursor CRSR0002 closed.
SQL7913	ODP deleted.
SQL7963	5461 rows fetched from cursor CRSR0002.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7962	Cursor CRSR0002 opened.
SQL7916	Blocking used for query.
SQL7912	ODP created.
CPI432F	Access path suggestion for file ITEM_FACT.
CPI4328	Access path of file ITEM_IDX2 was used by query.
CPI432D	Additional access path reason codes were used.
CPI432C	All access paths were considered for file ITEM_FACT.
CPI433A	Unable to retrieve query options file.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7967	PREPARE of statement STMT0002 completed.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
CPI433A	Unable to retrieve query options file.
SQL7954	Index ITEM_IDX2 created in TEAM06 on table ITEM_FACT in TEAM06.

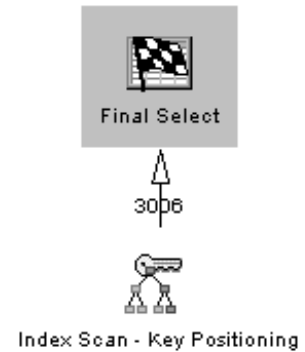


Figure 33. Key row positioning and selection

- ___ 12. Review the job log messages.
- ___ 13. In the Job Log window, scroll up to find the SQL7963 message that relates to the third query (1310 rows fetched...). A couple of messages above this message, you see the SQL7968 DESCRIBE messages, which are for the final query.

Message ID	Message
SQL7959	Cursor CRSR0002 closed.
SQL7913	ODP deleted.
SQL7963	1310 rows fetched from cursor CRSR0002.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7962	Cursor CRSR0002 opened.
SQL7916	Blocking used for query.
SQL7912	ODP created.
CPI4321	Access path built for file ITEM_FACT.
CPI432C	All access paths were considered for file ITEM_FACT.
CPI433A	Unable to retrieve query options file.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
SQL7967	PREPARE of statement STMT0002 completed.
SQL7968	DESCRIBE of prepared statement STMT0002 completed.
CPI433A	Unable to retrieve query options file.

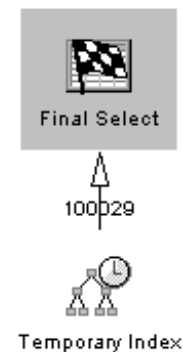


Figure 34. Temporary Index scan job log

- ___ 14. Review the job log messages for the final query.

__ 15. Close the Job Log window

__ 16. Close the SQL Script Center.

You have now completed this lab. You may close Operations Navigator.

Answers: Lab 1

The solutions given for the labs are based on our results using V4R5M0 on an iSeries server Model 270 in laboratory conditions. This system has two processors, 2 GB of main storage, 12 disk drives, and SMP installed.

Some answers shown in Table 13 and Table 14 will differ when Visual Explain is run in your own iSeries environment.

Task 3

Table 13. ITEM_FACT questions

Question	Answer
How many rows are in the table?	100,029
How many deleted rows are in the table?	0
Does the table re-use deleted rows?	Yes, default for SQL created tables
Who is the owner of the table?	HBEDOYA (May vary)
What level of authority does the public have?	*CHANGE
How many (if any) indexes does the table have?	None

Table 14. Database information

Question	Answer
How many rows are in CUST_DIM?	2,499
How many rows are in PART_DIM?	20,000
How many rows are in SUPP_DIM?	1,000

Answers: Lab 2

The solutions given for the labs are based on our results using V4R5M0 on an iSeries server Model 270 in laboratory conditions. Some answers will differ when you run Visual Explain in your own iSeries environment.

Task 1

- 8. What access method was chosen by the optimizer? Table Scan
- 9. Were any temporary objects used by the optimizer? Yes
- 10. If so, what kind of objects? Temporary Index

Task 2

- 10. What access method was chosen by the optimizer? Table Scan
- 11. Were any temporary objects used by the optimizer? Yes
- 12. If so, what kind of objects? Temporary Index

Task 3

Table 15. A record of the job number

Question	Answer
Job number	Will vary
User name	QUSER
Job name	QZDASOINIT

Table 16. Table scan questions

Question	Answer
How many rows were estimated by the table scan?	15
What is the number of rows in the table?	2499

Answers: Lab 3

The answers shown in Table 17 through Table 20 on page xlvii were obtained under laboratory conditions. Your answers may vary from the published answers.

Task 1

Table 17. Table scan questions

Question	Answer
From the table scan icon:	
What is the name of the table being queried?	VETEMxx.ITEM_FACT
How many are rows in the table?	100,029
What is the estimated number of rows that are selected?	100
Is an index advised?	Yes
If so, what keys is the optimizer suggesting?	Year, Month and Returnflag

Task 2

Table 18. Parallel table scan questions

Question	Answer
From the parallel table scan icon:	
What reason was given for selecting a table scan?	No indexes exist
Was parallel processing used?	Yes
If so, what degree was requested?	12. (May vary)
Was either parallel pre-fetch or parallel pre-load processing used?	No. (May vary)
From the final result icon:	
Which memory pool was used for the query?	2 (Normally *BASE)
How much memory was available for the query?	18,625,142 (May vary)

Task 3

Table 19. Skip sequential table scan questions

Question	Answer
Determine which indexes were used in the bit map creation by finding the indexes that are linked to the Dynamic Bitmap Icon	ITEM_EVI3, ITEM_EVI2
Which indexes were considered by the optimizer?	ITEM_EVI1, ITEM_EVI2, ITEM_EVI3
Which index was rejected?	ITEM_EVI1
What was the reason code for rejection?	4, cost to use this index, as determined by the query optimizer, was too high

Table 20. Further skip sequential table scan questions

Question	Answer
What reason was given for selecting a table scan?	Optimizer chose table scan
Will the table being scanned fit in main memory?	It varies.
Is an index advised?	Yes
If so, what keys is the optimizer suggesting?	Year, Month, Returnflag

Answers: Lab 4

The answers shown in Table 21 through Table 24 on page xlvii were obtained under laboratory conditions. Your answers may vary from the published answers.

Task 1

Table 21. Query implementation questions

Question	Answer
Was the creation of an index advised?	Yes
What is the list of key columns for the advised index?	Year, Month, Returnflag
Over which table was the index advised?	ITEM_FACT

Table 22. Keyed access questions

Question	Answer
Which index was used?	ITEM_IDX1
What access method was used?	Index Scan Key Positioning
What type of index was used?	Binary Radix
What is the index size (in bytes)?	13,853,248
What is the index page size (in bytes)?	4096
What is the reason code given for using this index?	I1 - Row selection
Did the optimizer advise the creation of an index?	No

Table 23. Key selection questions

Question	Answer
What access methods were used?	Key row positioning and key selection
What columns were used for key selection?	Month
Was index only access used?	Yes

Table 24. Temporary Index questions

Question	Answer
Over which file was the Temporary Index built?	ITEM_FACT
How many entries are in the created index?	100,029
Is this a sparse index?	No- Record selection has not been applied on this index
What type of index was created?	Binary Radix
Why was the index created?	I2 - Ordering/Grouping
What columns were used for the index key?	Quantity

