

# DB2 for i Index Advice *Trick or Treat?*



*Mike Cain*  
*DB2 for i Center of Excellence*  
*IBM Rochester, MN USA*  
*mcaain@us.ibm.com*

Hosted by:  
Centerfield Technology  
Rochester, MN USA  
[www.centerfieldtechnology.com](http://www.centerfieldtechnology.com)





Is DB2's index advice a trick or treat?



## DB2 for i – Index Technology Review

- Two types of indexing technologies are supported
  - *Radix* Index
  - *Encoded Vector* Index
- Each type of index has specific uses and advantages
- Respective indexing technologies compliment each other
- Indexes can be used for statistics and/or implementation
- Indexes can provide RRNs and/or data
- Indexes are probed and/or scanned
  - Probe can only occur on the leading, contiguous key columns
  - Scan (test) can occur on any key column
  - Probe and scan can be used together

# DB2 for i – Index Technology Review

cardinality The number of elements in a set.

- High cardinality = large distinct number of values
- Low cardinality = small distinct number of values



In general...

- A [radix index](#) is best when accessing a small set of rows and the key cardinality is high
- An [encoded vector index](#) is best when accessing a set of rows and the key cardinality is low
- Understanding the data and query are keys to success

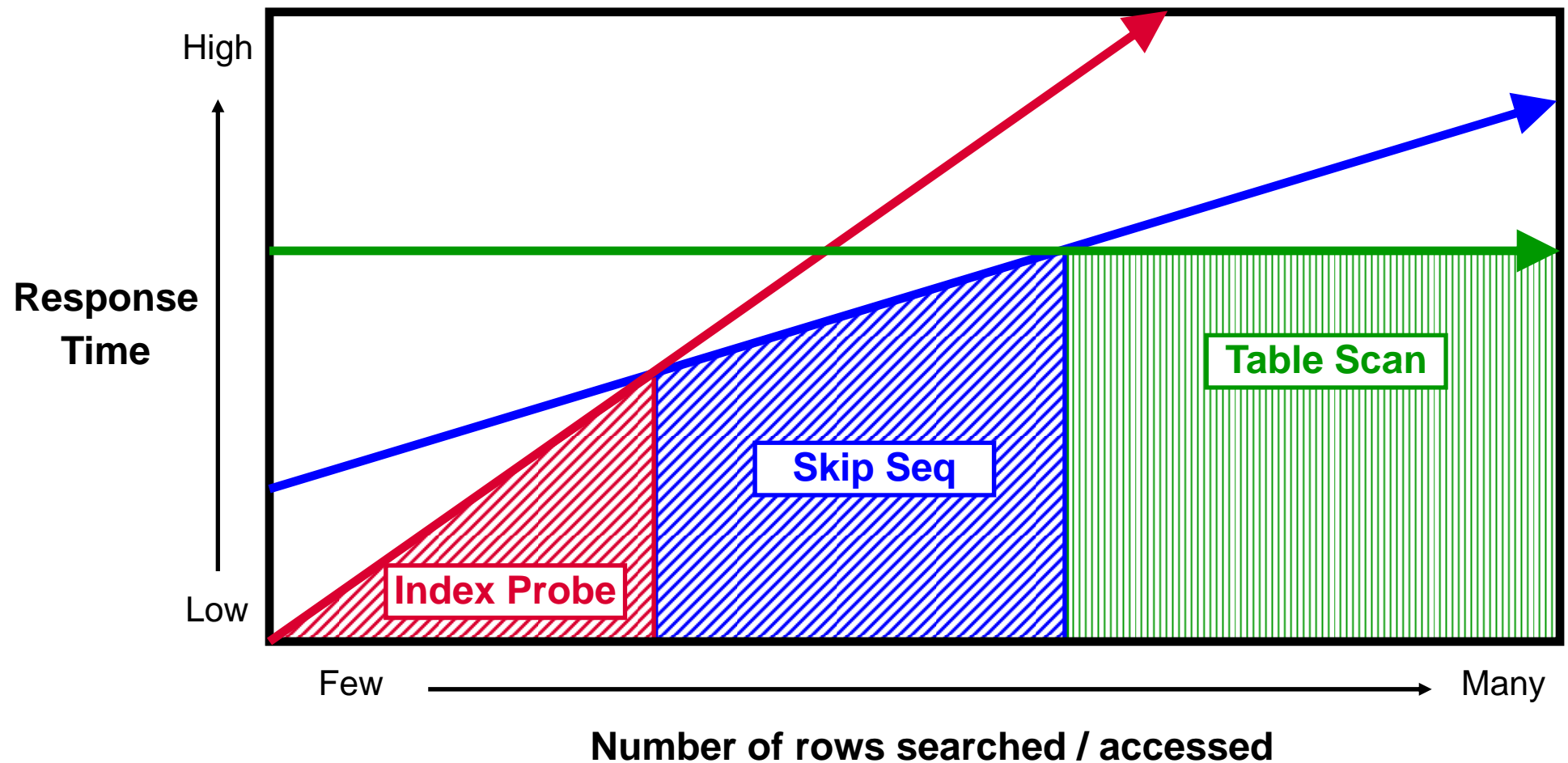
# DB2 for i – Index Technology Review

- CREATE INDEX SQL statement  
**CREATE INDEX MY\_IX ON MY\_TABLE (KEY1, KEY2)**
- CREATE ENCODED VECTOR INDEX SQL statement  
**CREATE ENCODED VECTOR INDEX MY\_EVI ON MY\_TABLE (KEY1)**
- Navigator for i – Database graphical interface
- CRTPF and CRTLF CL commands
  - Keyed access path within the physical file, or logical file or join logical file
- Primary Key, Foreign Key and Unique Key Constraints
  - CREATE TABLE
  - ALTER TABLE
  - ADDPFCST

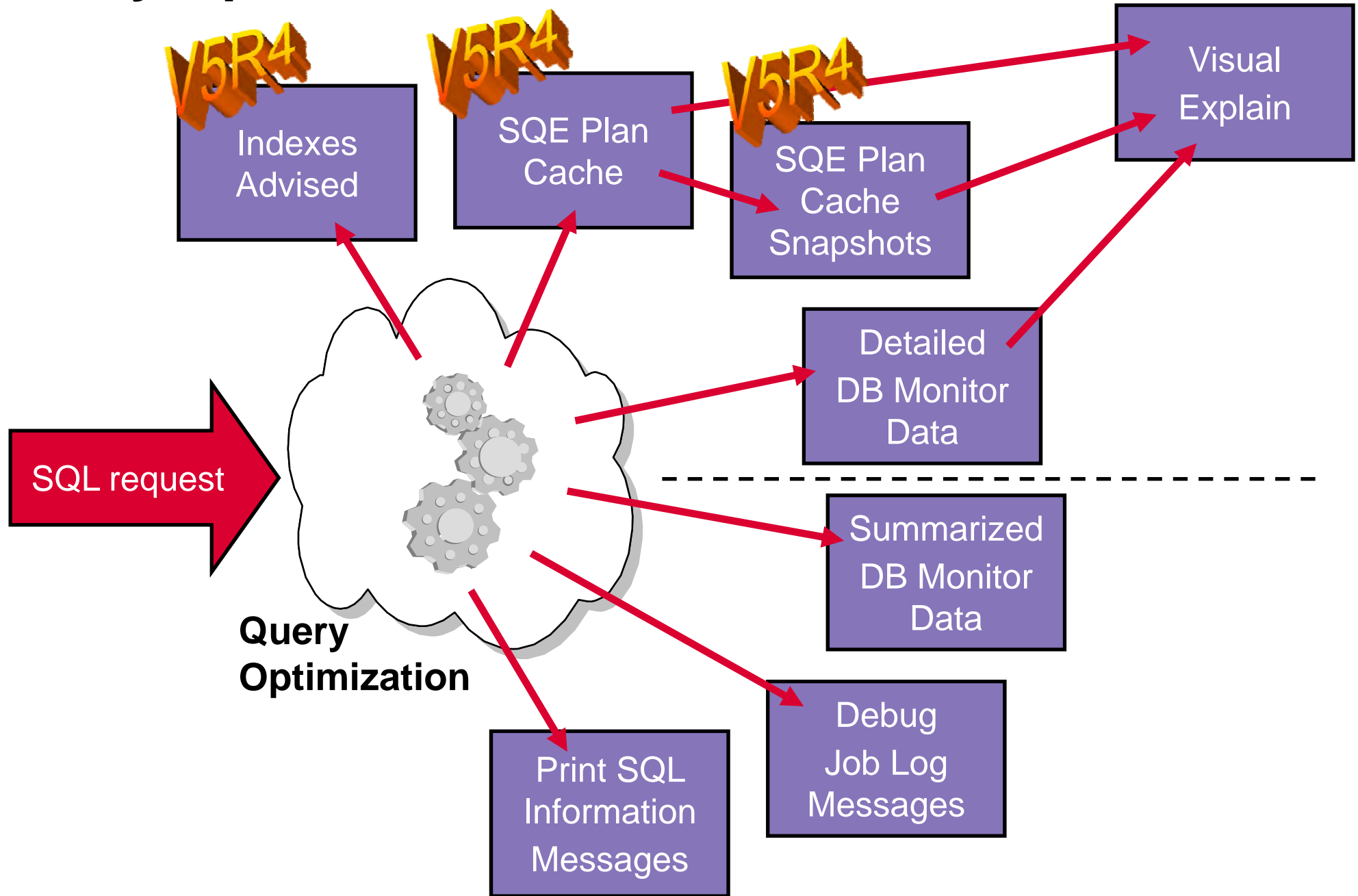
# Data Access Methods – Technology Review

Cost based optimization dictates that the fastest access method for a given table will vary based upon:

- Number of rows in the table
- Selectivity of the query



# Query Optimization and Runtime Information



# Indexing Advice from the Optimizer

- Index advice is provided to assist the user with tuning
- The advice should be used with other monitoring and analysis strategies
- Taking the advice is not required, but is usually a good idea
- The indexes created from the advice can help with:
  - Statistics
  - Implementation
- The optimizer and database engine do not necessarily have to use the index
  - After the index is created, query optimization begins anew
  - Trick or Treat!



# Indexing Advice from the Optimizer

- Both CQE and SQE provide index creation advice
- CQE
  - Basic advice
  - Radix index only
  - Based on table or index scan and local selection columns
  - Temporary index creation information also provides insight
  - CQE Visual Explain will try and tie pieces together for a more complete index
- SQE
  - Robust advice
  - Radix and EVI indexes
  - Based on all parts of the query
  - Multiple indexes can be advised for the same tables
  - Some limitations
    - 6.1 is better than V5R4
    - V5R4 is better than V5R3

## Indexing Advice from SQE 6.1

- There are only three main areas of advice
  - Row selection (includes joining)
  - Grouping / Ordering
  - Row selection plus Grouping / Ordering
- Remember that the query optimizer has the ability and freedom to rewrite the query
  - Tables and columns move around, appear and disappear
  - Trick or Treat!

# Indexing Advice from SQE 6.1

- Advice is based on:
  - Local equal predicates as leading keys
    - Including IS NULL
    - This is done regardless of the rest of the criteria below
  - *Additional* columns advised, in order of precedence:
    - equal join predicates  
OR
    - IN column predicates  
OR
    - a single non-equal join predicate
  
    - a single non-equal local predicate
      - This is always appended to the end of the advised list
      - Note: non-equal includes LIKE in 6.1
  - No more than 20 columns will be advised for a single index
  - Column order is: lowest cardinality to highest cardinality
    - For order independent columns

# Indexing Advice - Examples

-- Query 1

```
SELECT      A.CUSTOMER_NO, A.ORDER_DATE, A.QUANTITY
FROM        ORDERS A
WHERE       A.CUSTOMER_NO = 0112358;
```

```
CREATE INDEX ORDERS_IX1 ON ORDERS (CUSTOMER_NO);
```

# Indexing Advice - Examples

-- Query 2

```
SELECT      A.CUSTOMER_NO, A.ORDER_DATE, A.QUANTITY
FROM        ORDERS A
WHERE       A.CUSTOMER_NO = 0112358
AND         A.ITEM_ID LIKE 'ABC123YXZ%';
```

```
CREATE INDEX ORDERS_IX2 ON ORDERS (CUSTOMER_NO, ITEM_ID);
```

# Indexing Advice - Examples

-- Query 3

```
SELECT      A.CUSTOMER_NO, A.CUSTOMER, A.ORDER_DATE
FROM        ORDERS A
WHERE       A.CUSTOMER_NO IN (0112358, 1321345, 5891442)
AND        A.ORDER_DATE > '2005/06/30'
ORDER BY   A.ORDER_DATE;
```

```
CREATE INDEX ORDERS_IX3 ON ORDERS (CUSTOMER_NO, ORDER_DATE);
```

# Indexing Advice - Examples

-- Query 4

```
SELECT      A.CUSTOMER_NO, A.CUSTOMER, A.ORDER_DATE
FROM        ORDERS A
WHERE       A.CUSTOMER_NO = 0112358
OR          A.ORDER_DATE = '2005/06/30';
```

```
CREATE INDEX ORDERS_IX4 ON ORDERS (CUSTOMER_NO);
CREATE ENCODED VECTOR INDEX ORDERS_EVI4
ON ORDERS (ORDER_DATE);
```

**No Advice Provided**

# Indexing Advice - Examples

-- Query 5

```
SELECT      A.CUSTOMER_NO, B.CUSTOMER, A.ORDER_DATE, A.QUANTITY
FROM        ORDERS A,
            CUSTOMERS B,
            ITEMS C

WHERE       A.CUSTKEY = B.CUSTKEY
AND        A.ITEMKEY = C.ITEMKEY
AND        A.CUSTOMER_NO = 0112358;
```

```
CREATE INDEX ORDERS_IX5 ON ORDERS (CUSTKEY, ITEMKEY);
CREATE INDEX CUSTOMERS_IX5 ON CUSTOMERS (CUSTKEY);
CREATE INDEX ITEMS_IX5 ON ITEMS (ITEMKEY);
CREATE ENCODED VECTOR INDEX ITEMS_EVI5 ON ITEMS (ITEMKEY);
```



# Indexing Advice - Examples

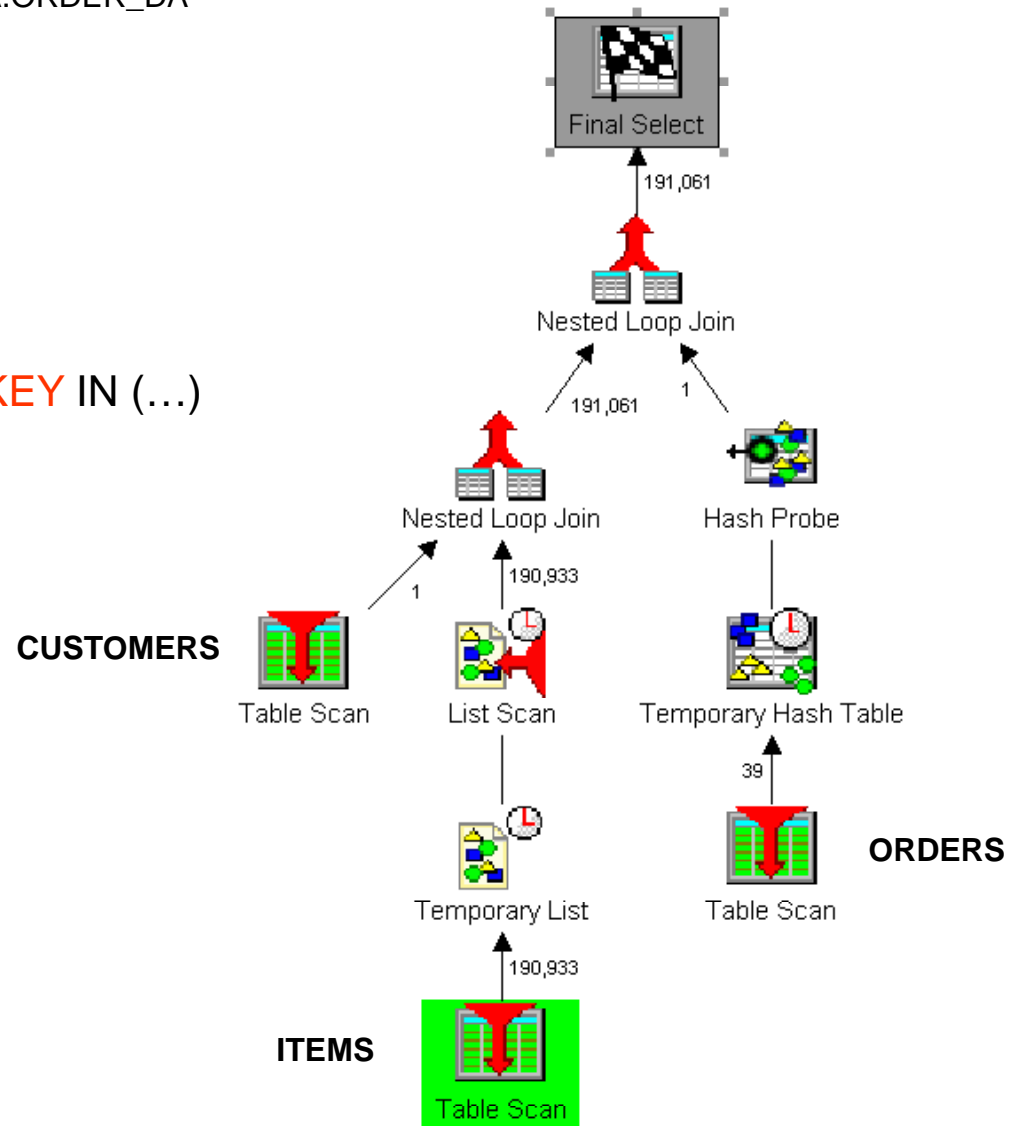
-- Query 5

```

SELECT      A.CUSTOMER_NO, B.CUSTOMER, A.ORDER_DA
FROM        ORDERS A,
           CUSTOMERS B,
           ITEMS C

WHERE       A.CUSTKEY = B.CUSTKEY
AND         A.ITEMKEY = C.ITEMKEY
AND         A.CUSTOMER_NO = 0112358;
    
```

LPG query rewrite adds: AND C.ITEMKEY IN (...)



# Index Advised – System wide

- New V5R4 feature, enhanced in V6R1
- System wide index advice
  - Data is placed into a DB2 table (QSYS2.SYSIXADV)
  - Autonomic
  - No overhead
- CQE and SQE support
  - CQE only provides basic advice based on local selection predicates
  - SQE provides complex advice based on all parts of the query
    - Not complete, but much better
- GUI interface via Navigator for i
  - Advice for System, or Schema, or Table
  - Report column list and column sorting is available
- System only adds (summary) rows, user must manage the data
  - Options to condense, clear or prune
- Can create single or multiple indexes directly from GUI
  - *Additional indexing analysis might be required to determine the optimal index*

# Index Advised – System wide

The screenshot shows the System i Navigator interface. On the left, a tree view shows the navigation structure under 'Databases' > 'Tpxe5'. A context menu is open over 'Tpxe5', with 'Index Advisor' selected. A sub-menu is also open, showing options like 'Index Advisor', 'Clear All Advised Indexes...', 'Condense Advised Indexes', and 'Prune Advised Indexes'. The 'My Connections' table on the right lists several system connections.

Name	Signed On User	Release	Description
Tpxe1.rchland.ibm.com		v5r4m0	Manage this system.
Tpxe2.rchland.ibm.com			Manage this system.
Tpxe3.rchland.ibm.com			Manage this system.
Tpxe4.rchland.ibm.com			Manage this system.
Tpxe5.rchland.ibm.com		v6r1m0	Manage this system.

Work with Index Advisor.

# Index Advised – System wide

Index Advisor - Tplxe5.rchland.ibm.com

File Edit View Help

Database: Tplxe5 - Advised Indexes for Tplxe5

Table for Which Index was Advised	Schema	Short Name	Partition	Keys Advised	Leading Keys Order Independent	Index Type Advised	Last Advised Use
STORE_SALES	TPCDS1GB	STORE00002		SS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:
STORE_SALES	TPCDS1GB	STORE00002		SS_STORE_SK		Encoded vector ...	12/12/07 3:
CATALOG_SALES	TPCDS1GB	CATAL00003		CS_CATALOG_PAGE...		Encoded vector ...	12/12/07 3:
CATALOG_SALES	TPCDS1GB	CATAL00003		CS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:
WEB_SALES	TPCDS1GB	WEB_SALES		WS_WEB_SITE_SK		Encoded vector ...	12/12/07 3:
WEB_SALES	TPCDS1GB	WEB_SALES		WS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:
STORE_RETURNS	TPCDS1GB	STORE00001		SR_STORE_SK		Encoded vector ...	12/12/07 3:

Scroll over...



# Index Advised – System wide

Index Advisor - Tplxe5.rchland.ibm.com

File Edit View Help

Database: Tplxe5 Advised Indexes for Tplxe5

Seq	Last Advised for Query Use	Times Advised for Query Use	Estimated Index Creation Time	Reason Advised	Logical Page Size Advised (KB)	Most Expensive Query Estimate	Average of Query Estimates	Rows in Table when Advised	Index Name
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	2880000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	2880000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	1440000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	1440000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	720000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	720000	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	287905	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	287905	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	144043	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	144043	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	73049	*HE
...	12/12/07 3:53:39 PM	4	00:00:01	Row selection	64	16	.0000	73049	*HE

# Index Advised – System wide and Condensed

The screenshot shows the System i Navigator interface. The left pane displays a tree view of system components, with 'Databases' expanded to show 'Tp1xe5'. A context menu is open over the 'Tp1xe5' database, and a sub-menu is open over the 'Index Advisor' option. The sub-menu has 'Condense Advised Indexes' selected.

The main pane displays a table of tables in the 'Tp1xe5.rchland.ibm.com' database, 'Schema: QGPL'. The table has columns: SQL Name, Partitioned, Owner, Last Changed, and Short Name.

SQL Name	Partitioned	Owner	Last Changed	Short Name
QCMSRC	No	QPGMR	11/15/07 2:51:08 AM	QCMSRC
QCPPSRC	No	QPGMR	11/16/07 9:34:44 AM	QCPPSRC
QCSRC	No	QPGMR	11/16/07 9:34:34 AM	QCSRC
QDDSSRC	No	QPGMR	11/15/07 2:51:08 AM	QDDSSRC
QEWCKBDMAP	No	QSYS	11/15/07 2:51:09 AM	QEWCKBDMAP
QEWCSNFMT	No	QSYS	11/15/07 2:51:09 AM	QEWCSNFMT
QEWCSRC	No	QSYS	11/15/07 2:51:10 AM	QEWCSRC
QEWLSRC	No	QSYS	11/15/07 2:51:10 AM	QEWLSRC
QFMTSRC	No	QPGMR	11/15/07 2:51:11 AM	QFMTSRC
QINVREC	No	QPGMR	11/16/07 9:34:05 AM	QINVREC
QLBLSRC	No	QPGMR	11/16/07 9:34:26 AM	QLBLSRC
QLBLSRC1	No	QPGMR	11/16/07 8:39:28 AM	QLBLSRC1
QMAPSRC1	No	QPGMR	11/16/07 8:39:28 AM	QMAPSRC1
QMNUSRC	No	QPGMR	11/15/07 2:51:11 AM	QMNUSRC
QORDDL	No	QPGMR	11/16/07 9:34:08 AM	QORDDL
QORDHDR	No	QPGMR	11/16/07 9:34:08 AM	QORDHDR
QPNLSRC	No	QPGMR	11/15/07 2:51:12 AM	QPNLSRC

At the bottom of the window, a status bar reads: "Condense the advised indexes for the selected object."

# Index Advised – System wide and Condensed

## Queries:

```
...WHERE YEAR = 2008 AND QUARTER = 1 AND COLOR = 'BLUE;  
...WHERE YEAR = 2008 AND QUARTER = 1;  
...WHERE YEAR = 2008;
```

## Index advice - by query:

```
YEAR, QUARTER, COLOR  
YEAR, QUARTER  
YEAR
```

## Condensed advice:

```
YEAR, QUARTER, COLOR
```

# Index Advised – System wide – Show Statements

Index Advisor - Tplxe5.rchland.ibm.com

File Edit View Help

Database: Tplxe5 Advised Indexes for Tplxe5

Table for Which Index was Advised	Schema	Short Name	Partition	Keys Advised	Leading Keys Order Independent	Index Type Advised	Last Advised for Query Use	Times Advised for Query Use	Estimated Index Creation Time
STORE_SALES	TPCDS1GB	STORE00002		SS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:53:39 PM	4	00:00:01
STORE_SALES	TPCDS1GB	STORE00002		SS_STORE_SK		Encoded vector ...	12/12/07 3:53:39 PM		
CATALOG_SALES	TPCDS1GB	CATAL00003		CS_CATALOG_PAGE...		Encoded vector ...	12/12/07 3:53:39 PM		
CATALOG_SALES	TPCDS1GB	CATAL00003		CS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:53:39 PM		
WEB_SALES	TPCDS1GB	WEB_SALES		WS_WEB_SITE_SK		Encoded vector ...	12/12/07 3:53:39 PM		
WEB_SALES	TPCDS1GB	WEB_SALES		WS_SOLD_DATE_SK		Encoded vector ...	12/12/07 3:53:39 PM		
STORE_RETURNS	TPCDS1GB	STORE00001		SR_STORE_SK		Encoded vector ...	12/12/07 3:53:39 PM		
STORE_RETURNS	TPCDS1GB	STORE00001		SR_RETURNED_DAT		Encoded vector ...	12/12/07 3:53:39 PM	4	00:00:01

Context menu for the first row:

- Create Index
- Remove from List
- Show SQL
- Show Statements...
- Table

SQL Plan Cache Statements - Tplxe5.rchland.ibm.com(Tplxe5)

Filters to apply:

- Minimum runtime for the longest execution of the statement:  Seconds
- Statements that ran on or after this date and time:
- Top 'n' most frequently run statements:
- Top 'n' statements with the largest total accumulated runtime:

Statements:

Last Time Run	Most Expensive Time (sec)	Total Processing Time (sec)	Total Times Run	Average Processing Time (sec)	Statement
12/12/07 5:31:06 PM	6.6073	6.6073	1	6.6073	with ssr as (s...
12/12/07 3:53:38 PM	0.0000	0.0000	0	0.0001	with ssr as (s...
12/12/07 3:53:38 PM	0.0000	0.0000	0	0.0001	with ssr as (s...
12/12/07 3:53:39 PM	0.0000	0.0000	0	0.0001	with ssr as (s...

Status: Complete - filters

Context menu for the first row:

- Visual Explain
- Show Longest Runs
- Show Active Jobs
- Show User History
- Work with SQL Statement
- Save to New...
- Plan

Buttons: Columns..., Refresh, Close, Help



# Visual Explain

- Enhanced in V5R4 and again in V6R1
- Graphical representation of query plan
  - Representation of the DB objects and data structures
  - Representation of the methods and strategy
  - Associated environmental information
  - ***Advice on indexes and column statistics***
  - Highlighting of specific query rewrites
  - Highlighting of expensive methods
- CQE and SQE support
- GUI interface via Navigator for i
  - Explain only
  - Run and Explain
  - Explain while running (SQE only)
  - Explain while active (SQE only)
- Based on detailed optimizer information
  - SQE Plan Cache
  - SQE Plan Cache Snapshots
  - Detailed Database Monitor Data

# Visual Explain

## Run SQL Scripts



## SQL Performance Monitor or Plan Cache Snapshot Analysis

Statements - Test Thursday Feb 21 2008 - Tplxe5.rchland.ibm.com(Tplxe5)

Filters to apply:

- Minimum runtime for the longest execution of the statement:  Seconds
- Statements that ran on or after this date and time:
- Statements that reference the following objects:

Start Time	Most Expensive Time	Total Processing Time	Total Time
12/12/07 5:31:06 PM	6.6073	6.6070	
2/21/08 11:42:25 AM			0.0320
2/21/08 11:00:28 AM			0.0590
12/3/07 11:15:16 AM			0.0170
12/3/07 11:15:17 AM			0.0160
2/21/08 12:00:07 AM	0.0158	0.2760	

Status: Complete

Test Thursday Feb 21 2008 - SQL Statements - Statements - Tplxe5.rchland.ibm.com(Tplxe5)

File View Actions Help

Start Time	End Time	Runtime	Statement Outco...	SQLS...	SQLCODE	Operation	Statement Text
2007-12-12 17:31:06.243647	2007-12-12 17:31:06.243647	6.607374	Successful	00000		0 OPEN	with sssr as (select
2008-02-21 11:42:25.687733	2008-02-21 11:42:25.687733					0 OPEN	select * from qsys
2007-12-03 11:15:16.829773	2007-12-03 11:15:16.829773					0 OPEN	SELECT * FROM
2007-12-03 11:15:17.884792	2007-12-03 11:15:17.884792					0 OPEN	SELECT * FROM
2007-12-04 14:45:52.613002	2007-12-04 14:45:52.613002					0 OPEN	SELECTOrderYes
2007-12-03 11:15:14.197516	2007-12-03 11:15:14.197516					0 OPEN	SELECT CREATE
2007-12-03 11:15:16.829773	2007-12-03 11:15:16.829773					0 OPEN	SELECT * FROM
2007-12-03 11:15:17.355715	2007-12-03 11:15:17.355715					0 OPEN	SELECT * FROM
2007-12-03 11:15:18.415505	2007-12-03 11:15:18.415505					0 OPEN	SELECT SYSTEM
2007-12-12 17:31:06.243647	2007-12-12 17:31:06.243647					0 OPEN	SELECT SYSSUM
2008-01-30 00:00:01.570655	2008-01-30 00:00:01.570655					0 SELECT INTO	SELECT WEDCN

# Visual Explain – Index Advisor

Attribute	Value
<b>Time Information</b>	
Timestamp for Creation of Monitor Entry	2008-02-21-12.11.44
Statement Start Timestamp	2008-02-21-12.11.44
Statement End Timestamp	2008-02-21-12.11.44
Total Estimated Run Time (ms)	50
<b>Actual Runtime Information</b>	
Optimization Time (ms)	4
Run Time (ms)	13
Statement Open Time (ms)	13
Statement Fetch Time (ms)	Not Available
Statement Close Time (ms)	Not Available
Rows Fetched	1
Total Times Query Was Run	1
Total Time For All Runs (ms)	14
Synchronous Database Reads	Not Available
Asynchronous Database Reads	Not Available
Page Faults	Not Available
<b>Information about SQL statement exe...</b>	
Statement Number	24
Statement Function	Select
Statement Operation	Open
Statement Type	Dynamic
Statement Name	STMT0044
Statement Outcome	Unsuccessful
SQL Return Code	-666

select \* from qsys2.sysixadv where table\_schema = 'MCAIN'

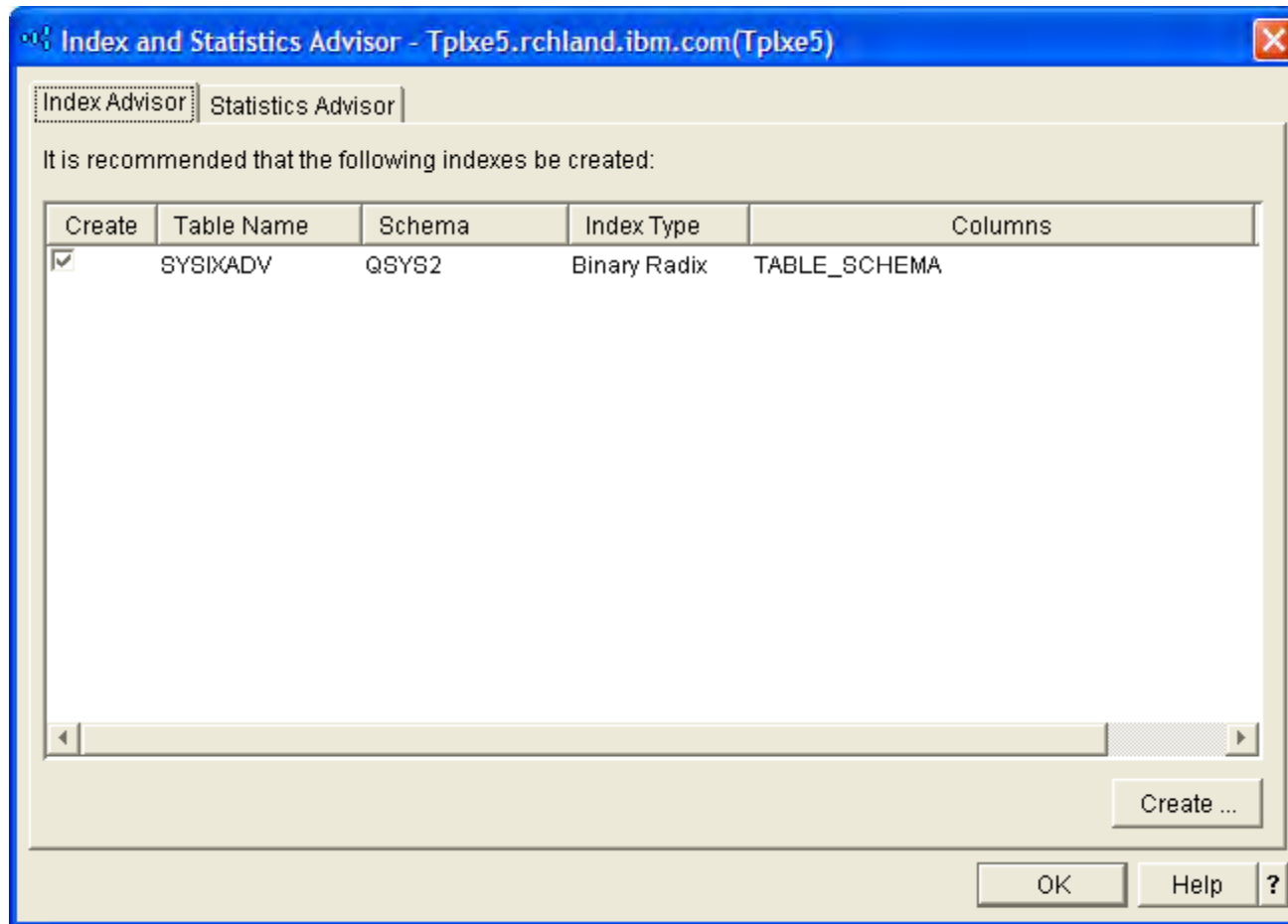
View Actions Options Help

- Zoom
- Overview
- Icon Spacing
- Orientation
- Arrow Labels
- Icon Labels
- Highlight Expensive Icons
- Highlight Index Advised**
- Highlight LPG
- Highlight Materialized Query Tables
- Refresh

Actions Options Help

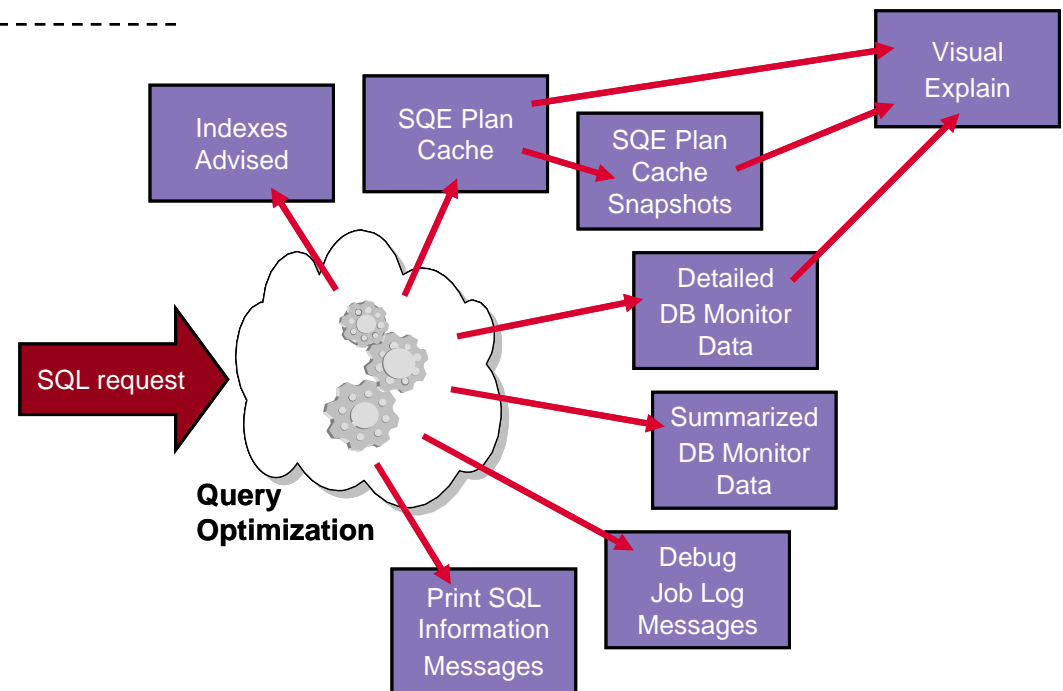
- Table Description
- Statistic Data
- Table Definition
- Show Indexes
- Show Materialized Query Tables
- Show Related
- Index Description
- Index Definition
- Create Index
- Explain SQL
- Function Properties
- Advisor**
- Display Query Environment

# Visual Explain – Index Advisor



# Indexed Advised from other Mechanisms

- **SQE Plan Cache (V5R4)**
    - No direct index advice
    - Index advice via Snapshot data or Visual Explain
  - **SQE Plan Cache Snapshot (V5R4)**
    - Enhanced SQE index advised
    - "3020" records to show multiple indexes for same table
    - Temporary index created
  - **Detailed Database Monitor (V5R4)**
    - Enhanced SQE index advised
    - "3020" records to show multiple indexes for same table
    - Temporary index created
- 
- **Summary Database Monitor**
    - No enhanced SQE index advised
    - Basic index advice
    - Temporary index created
  - **Debug Messages in Job Log**
    - No enhanced SQE index advised
    - Basic index advice
    - Temporary index created
  - **Print SQL Information**
    - No index advice
    - Temporary index created



## 3020 Row - Index Advised Analysis (V5R4)

- What indexes are advised, and how often?

```
SELECT COUNT(*) "Times Advised",  
       qvptbl "Table Name",  
       qvplib "Schema",  
       qq1000L "Keys Advised"  
FROM   monitor file  
WHERE  qqrid = 3020  
GROUP BY qvplib, qvptbl, qq1000L  
ORDER BY 1 desc, 3, 2, 4
```

No. Times Advised	Schema	Table Name	KEYS_ADVISED
28	DBITSODATA	CUSTOMERS	CUSTOMER, CUSTKEY ...
24	DBITSODATA	ORDERS	RETURNFLAG, CUSTKEY ...
22	DBITSODATA	SUPPLIERS	SUPPKEY ...
18	DBITSODATA	PARTS	PARTKEY ...
15	DBITSODATA	DATES	YEAR, DATEKEY ...
12	DBITSODATA	ORDERS	SHIPDATE ...
12	DBITSODATA	ORDERS	YEAR, QUANTITY ...
10	DBITSODATA	DATES	DATEKEY ...
10	DBITSODATA	SUPPLIERS	SUPPLIER, SUPPKEY ...

*Remember, system wide index advice is ALWAYS available in QSYS2/SYSIXADV*

**What if you do not take the  
advice provided?**

**Cue the scary music!**

# Autonomic Index Creation

- Optimizer can have the DB Engine create a temporary index
- Both full and sparse indexes can be created
- Temporary indexes are not used for statistics
- Temporary indexes are *maintained*
- CQE
  - Temporary indexes are not reused and not shared
  - Usually a bottleneck in query performance
  - Can impact overall system performance
  - Can increase the amount of temporary storage used
- SQE
  - New feature in V5R4
  - Temporary indexes are reused and shared across jobs and queries
  - Creation is based on “watching” the query requests over time
  - Creation is based on optimizer’s own index advice
  - Temporary index maintenance is delayed when all associated cursors closed

The word "Scary!" is written in a large, 3D, yellow-to-orange gradient font, slanted upwards to the right. The letters have a shadow effect, giving them a three-dimensional appearance.



**“I see indexes”**

# Index Evaluator (Show Indexes)

Environment: My Connections

Tplxe5.rchland.ibm.com: Tables Database: Tplxe5 Schema: TPCDS1GB

SQL Name	Partitioned	Owner	Last Changed	Short Name	Text
CATALOG_RETURNS	No	QDFTOWN	12/12/07 2:39:35 PM	CATAL00002	
CATALOG_SALES	No	QDFTOWN	12/12/07 2:39:35 PM	CATAL00003	
CUSTOMER			2/07 2:39:36 PM	CUSTOMER	
CUSTOMER_ADDRESS			2/07 2:39:36 PM	CUSTO00001	
CUSTOMER_DEMOGRAPHICS			2/07 2:39:36 PM	CUSTO00002	
DATE_DIM			2/07 2:39:36 PM	DATE_DIM	
DBGEN_VERSION			2/07 2:39:36 PM	DBGEN00001	
HOUSEHOLD_DEMOGRAPHIC			2/07 2:39:37 PM	HOUSE00001	
INCOME_BAND			2/07 2:39:37 PM	INCOM00001	
INVENTORY			2/07 2:39:37 PM	INVENTORY	
ITEM			2/07 2:39:37 PM	ITEM	
PROMOTION			2/07 2:39:37 PM	PROMOTION	
REASON			2/07 2:39:37 PM	REASON	
SHIP_MODE			2/07 2:39:37 PM	SHIP_MODE	
STORE			2/07 2:39:37 PM	STORE	
STORE_RETURNS			2/07 2:39:38 PM	STORE00001	
STORE_SALES			2/07 2:39:38 PM	STORE00002	

My Tasks - Tplxe5.rchland.ibm.com

- Add a connection

Databases tasks

- Select schemas to display
- Run an SQL script
- Map your database
- Create a new SQL performan...

Display a list of the indexes for the selected table(s).

# Index Evaluator (Show Indexes)

SQL Name	Type	Schema	Owner	Short Name	Text
"Temporary index"	Temporary index				2 MAINTAINED TEMPORARY INDEXES
CATALOG_SALES_INDEX_SHIP_MODE	Index	TPCDS1GB	MCAIN	CATAL00004	
CATALOG_SALES_QUANTITY	Index	TPCDS1GB	MCAIN	CATAL00005	
CS_B_A	Foreign Key Constraint	TPCDS1GB			
CS_B_C	Foreign Key Constraint	TPCDS1GB			
CS_B_CD	Foreign Key Constraint	TPCDS1GB			
CS_B_HD	Foreign Key Constraint	TPCDS1GB			
CS_CC	Foreign Key Constraint	TPCDS1GB			
CS_CP	Foreign Key Constraint	TPCDS1GB			
CS_D1	Foreign Key Constraint	TPCDS1GB			
CS_D2	Foreign Key Constraint	TPCDS1GB			
CS_I	Foreign Key Constraint	TPCDS1GB			
CS_P	Foreign Key Constraint	TPCDS1GB			
CS_S_A	Foreign Key Constraint	TPCDS1GB			
CS_S_C	Foreign Key Constraint	TPCDS1GB			
CS_S_CD	Foreign Key Constraint	TPCDS1GB			
CS_S_HD	Foreign Key Constraint	TPCDS1GB			
CS_SM	Foreign Key Constraint	TPCDS1GB			
CS_T	Foreign Key Constraint	TPCDS1GB			
CS_W	Foreign Key Constraint	TPCDS1GB			
Q_TPCDS1GB_CATAL00003_CS_ITEM_SK_00001	Primary Key Constraint	TPCDS1GB			

Who ya gonna call?

Aha!

Scroll right for more treats...



# Index Evaluator (Show Indexes)

Indexes for TPCDS1GB.CATALOG\_SALES - Tplxe1.rchland.ibm.com

File Edit View Help

4 minutes old

Database: Tplxe1 Indexes for TPCDS1GB.CATALOG\_SALES

Valid	Creation Date	Last Build	Last Query Use	Last Query Statistics Use	Query Use Count	Query Statistics Use Count	Last Used Date	Days Used Count	Date Rese Used Cou
Yes	2/21/08 5:00:03 PM	2/21/08 5:00:05 PM	2/21/08 5:01:46 PM	2/21/08 5:01:46 PM	2	2	2/21/08	1	
Yes	2/21/08 5:00:58 PM	2/21/08 5:01:00 PM	2/21/08 5:34:46 PM	2/21/08 5:36:18 PM	5	19	2/21/08	1	
Yes	3/1/06 2:57:25 PM	3/1/06 2:57:43 PM			0	0	2/21/08	2	
Yes	3/1/06 2:58:00 PM	3/1/06 2:58:15 PM			0	0	2/21/08	2	
Yes	3/1/06 2:57:44 PM	3/1/06 2:57:58 PM			0	0	2/21/08	2	
Yes	3/1/06 2:58:17 PM	3/1/06 2:58:31 PM			0	0	2/21/08	2	
Yes	3/1/06 2:58:32 PM	3/1/06 2:58:43 PM			0	0	2/21/08	2	
Yes	3/1/06 2:58:44 PM	3/1/06 2:58:57 PM			0	0	2/21/08	2	
Yes	3/1/06 3:00:14 PM	3/1/06 3:00:26 PM			0	0	2/21/08	2	
Yes	3/1/06 3:00:57 PM	3/1/06 3:01:10 PM		3/27/07 9:03:27 AM	0	1	2/21/08	2	
Yes	3/1/06 2:58:59 PM	3/1/06 2:59:12 PM	3/27/07 9:03:27 AM	2/21/08 5:36:18 PM	1	14	2/21/08	2	
Yes	3/1/06 2:59:13 PM	3/1/06 2:59:25 PM			0	0	2/21/08	2	
Yes	3/1/06 2:59:26 PM	3/1/06 2:59:40 PM			0	0	2/21/08	2	
Yes	3/1/06 2:59:57 PM	3/1/06 3:00:12 PM		2/21/08 5:36:18 PM	0	13	2/21/08	2	
Yes	3/1/06 2:59:41 PM	3/1/06 2:59:54 PM			0	0	2/21/08	2	
Yes	3/1/06 3:00:28 PM	3/1/06 3:00:42 PM			0	0	2/21/08	2	
Yes	3/1/06 3:00:43 PM	3/1/06 3:00:57 PM	2/21/08 4:58:28 PM	2/21/08 5:34:54 PM	1	22	2/21/08	2	
Yes	3/1/06 3:01:11 PM	3/1/06 3:01:25 PM			0	0	2/21/08	2	
Yes	3/1/06 3:01:26 PM	3/1/06 3:01:38 PM			0	0	2/21/08	2	
Yes	3/1/06 12:04:04 PM	3/1/06 12:04:05 PM			0	0	2/21/08	2	

1 - 21 of 21 objects

# Indexing Strategy - Basic Proactive Approach

## Radix Indexes

- Equal local selection columns
  - Equal join columns
  - Local selection columns + join columns
  - Local selection columns + grouping columns
  - Local selection columns + ordering columns
  - Ordering columns + local selection columns
  - One non-equal column at the end
- } Minimum

## Encoded Vector Indexes

- Equal local selection column (single key)
- Join columns (data warehouse - star or snowflake schema)
- Grouping columns
  - COUNT and COUNT(DISTINCT)

# Indexing Strategy – Basic Reactive Approach

If the optimization information indicates the following, and no suitable index exists:

**Full table scan** → Create an index on local selection columns

**Temporary index** → Create an index on join columns  
→ Create an index on grouping columns  
→ Create an index on ordering columns

**Hash table** → Create an index on join columns  
→ Create an index on grouping columns

“Perfect”, multiple key column radix indexes are usually best

Balance individual query performance with overall maintenance

# Questions & Answers



Thank You



**Halloween**, or **Hallowe'en**, is an international holiday celebrated on October 31. Halloween activities include trick-or-treating, ghost tours, bonfires, costume parties, visiting haunted attractions, carving jack-o'-lanterns, reading scary stories and watching horror movies. Irish immigrants carried versions of the tradition to North America in the nineteenth century. Other western countries embraced the holiday in the late twentieth century. Halloween is celebrated in several countries of the Western world, most commonly in the United States, Canada, Ireland, Puerto Rico, Japan, New Zealand, United Kingdom and occasionally in parts of Australia. In Sweden the All Saints' official holiday takes place on the first Saturday of November.

# Trademarks and Disclaimers

© IBM Corporation 1994-2008. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Trademarks of International Business Machines Corporation in the United States, other countries, or both can be found on the World Wide Web at <http://www.ibm.com/legal/copytrade.shtml>.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

The customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Photographs shown may be engineering prototypes. Changes may be incorporated in production models.