



DB2 Under the Hood: Managing Performance with Quest Central for DB2

Jim Wankowski

Agenda

- Defining Performance
- Monitoring Methods
- Identifying/Resolving Performance Issues
 - ✓ Memory Management
 - ✓ Space management
 - ✓ Application Design

What is Performance?

Performance: *noun*

“The fulfillment of a claim, promise, or request”

- How does your company define performance?
 - System availability
 - Transaction throughput
 - Minimum response times (SLA's)

Subsystem/Instance Monitoring

All aspects of the DB2 subsystem or Instance need to be monitored.

- Take a look at the big picture
 - Think of DB2 as an ecosystem

- Do not Tune for the sake of Tuning!
 - Where are your bottlenecks?

Monitoring Methods

z/OS

- **Instrumentation Facility Component (IFC)**
 - **Statistics**
 - Global statistical data
 - **Accounting**
 - Detail info for specific application
 - **Performance**
 - Most detailed \$\$\$
 - Only use for short periods

L,U,W

- **Snapshot Monitor**
 - Show status of database for an instant in time
 - Monitor Switches need to be turned on at the instance level to collect data
 - Low overhead (~5%)
- **Event Monitor**
 - Historical collection of data
 - More Overhead (~10-20%)
 - Main focus on application statistics

Memory Management

z/OS

- EDM Pool
- RID Pool
- Buffer Pool

L,U,W

- Catalog Cache
- Buffer Pool
- Lock List
- Package Cache
- Sort Heap

A key performance objective should be to minimize the amount of disk access

Memory Usage

z/OS

EDM Pool

- “System Bufferpool”
 - Minimizes I/O against catalog and directory
- Contains
 - DBD (Database Descriptor)
 - CT (Cursor Table)
 - PT (Package Table)
 - SKCT (Skeleton Cursor Table)
 - SKPT (Skeleton Package Table)
 - Plan/Package authorization Cache (CACHESIZE > 0)
 - Dynamic SQL skeletons (Dynamic SQL caching active)

L,U,W

Catalog Cache

- Minimizes I/O against catalog
- Contains
 - SYSTABLES information
 - Authorization information
 - SYSDBAUTH
 - Execute Privileges for routines

Possible Performance Implications

z/OS EDM Pool

- Increased I/O activity against DSNDB01
 - SCT02
 - SPT01
 - DBD01
- Increased response times due to loading the SKCTs, SKPTs, and DBDs
- Re-preparation of Dynamic SQL
- Fewer threads used concurrently, due to a lack of storage
- Resource unavailable “-904”

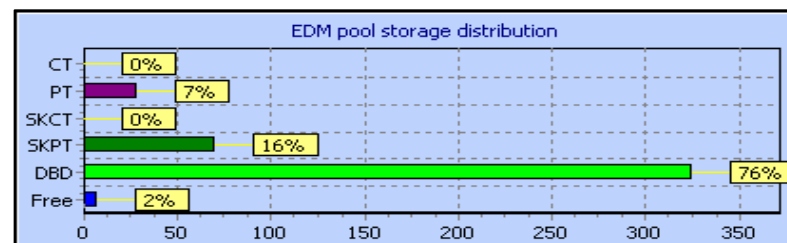
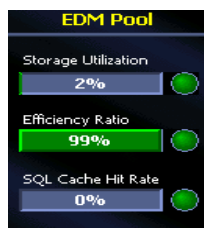
L,U,W Catalog Cache

- Increased bind times
- Increased compile times
- Increase time to check DB and execution privileges

What to monitor

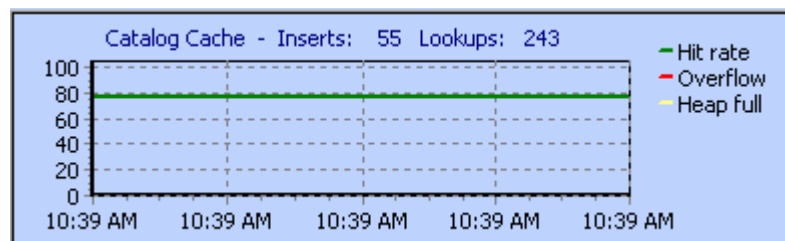
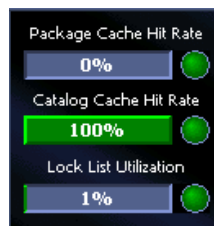
z/OS

- EDM Pool hit ratio should be at least 80%
 - PT/CT's
 - 80-90%
 - DBD requests
 - 100%
 - Pages used for Package/Cursor tables <50% of pool



L,U,W

- Catalog Cache hit ratio 80-90%
 - Catalog Cache lookups
 - *Cat_cache_lookups*
 - Catalog Cache Inserts
 - *Cat_cache_inserts*
 - Catalog Cache Overflows
 - *Cat_cache_overflows*



Managing EDM Pool Size

- Reduce size of DBD's
 - ✓ Try to minimize # of objects in database
 - ✓ Use 32K Pieces for large databases
 - ✓ Run MODIFY utility regularly to remove old recovery info.
 - ✓ Dropped objects
 - ✓ REORG tablespaces when tables dropped/recreated
- Dataspaces for Dynamic Caching
 - ✓ Specify a portion of EDM pool in a dataspace
 - ✓ EDMPOOL DATA SPACE SIZE > 0
- Avoid large Plans
 - Use Packages (avoid DBRM's)
 - Break applications into separate Plans
- DEGREE(ANY) maintains 2 access paths
 - One each for Parallelism YES/NO
- Bind w/AQUIRE(USE) vs. AQUIRE(ALLOCATE) when possible.
- Use RELEASE(COMMIT) for infrequently used Plans/Pkg's
- Use RELEASE(DEALLOCATE) cautiously
 - Can cause EDM pool to grow enormously.
 - Only change a few programs at a time.

RID Pool – z/OS

“Row Identifier” Pool

- Enforces unique keys during multi-row updates
- Used for Storing and Sorting RID's for:
 - List Prefetch
 - Multiple index access
 - Hybrid Joins
- Performance Implication
 - If RID pool is too small above access paths revert to TS scans

What to Monitor

- Insufficient pool size
 - RID pool too small
 - Recalculate size

- RDS Limit
 - RID list > 25% #rows in table
 - Prefetch turned off and TS scan results
 - Determined at Bind time
 - Have tables grown since BIND?
 - Make sure stats are accurate
 - RUNSTATS/REBIND

- DM Limit
 - RIDs req'd to satisfy query > 16 million
 - TS scan results
 - Is TS Scan best access?
 - Re-evaluate indexes
 - Add additional filtering

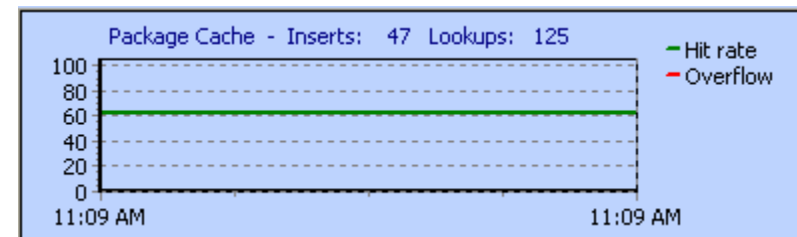
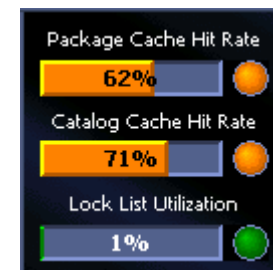
SQL statistics		
Event	Count	Per sec
Commits	18,836,894	49.71
Rollbacks	4,792	0.01
Incremental binds	70,285	0.19
Runtime reoptimizes	0	0.00
Direct row success	0	0.00
Direct row, index use	0	0.00
Direct row, tbs scan	0	0.00
RID list success	520,010,971	1,372.21
RID failure, storage	0	0.00
RID failure, RDS limit	926	0.00
RID failure, DM limit	0	0.00
RID failure, size limit	1	0.00

Package Cache – L,U,W

- Minimizes I/O against catalog
 - Loading Packages
 - Having to prepare Dynamic SQL
- Possible Performance Implications
 - Slower response time with Dynamic SQL

What to Monitor

- Package cache Hit Ratio
- Package cache overflows
 - *Pkg_cache_num_overflows*
- Package cache lookups
 - *Pkg_cache_lookups*
- Package cache inserts
 - *Pkg_cache_inserts*
- Package cache high water mark
 - *pkg_cache_size_top*



Sort Heap-L,U,W

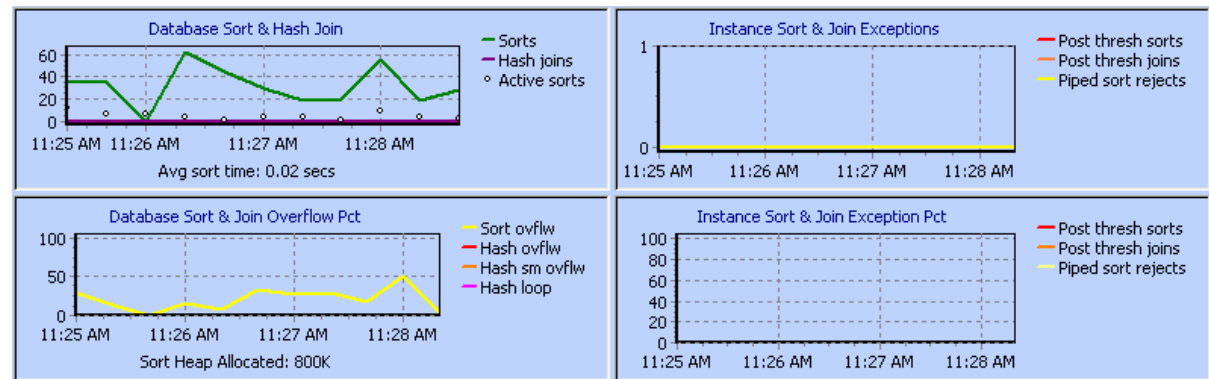
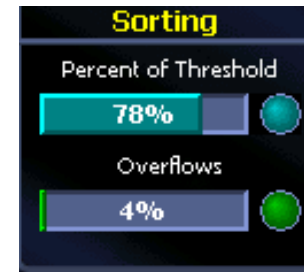
Number of pages available for private or shared sorts

- Used by Optimizer for determining access paths
 - Sorting
 - Hash Joins
 - Index ANDing
 - Dynamic bitmaps

- Possible Performance implications
 - Frequent large sorts
 - Non-piped vs. Piped

What to Monitor

- Sort Heap
 - *Sortheap*
- Sort Heap Threshold
 - *Sheapthres*



- Proper indexing can minimize sorting

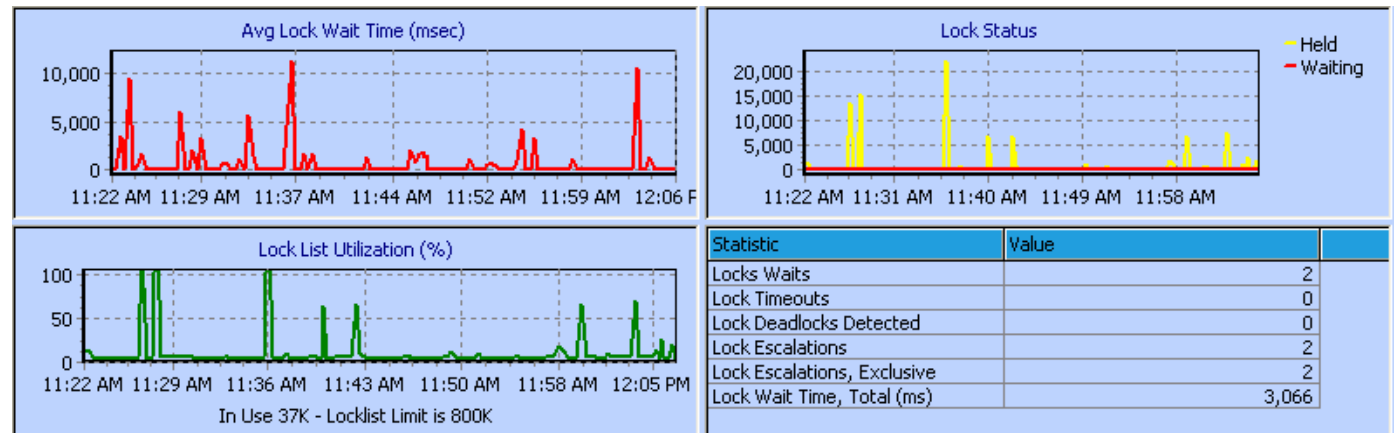
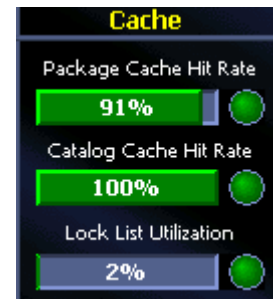
Locklist – L,U,W

Amount of storage allocated to a database for locking

- Possible Performance Implications
 - Lock Escalations
 - Decrease in concurrency
 - Degradation of performance due to lock waits
 - Deadlocking
 - SQLCODE -912
 - Maximum # locks reached in database
- How to avoid
 - Frequent COMMITs
 - LOCK TABLE for heavy updates

What to Monitor

- Lock List
 - *Locklist*
- Maximum Locks
 - *Maxlocks*
- Lock Escalations
 - *lock_escals*



Quest Central Performance Diagnostics

The screenshot displays two windows from the Quest Central Performance Diagnostics application. The top window, titled 'Spotlight on DB2-05/390 - DZS7 V7 QSZDB2', shows a tree view of DB2 subsystems (DZS7, DZ7A, DZ7B, DZ7C) and a 'Users' table with columns for Total, DDF, TSO, CICS, IMS, Call Attach, Utility, and RRSAP. The bottom window, titled 'Spotlight on DB2 - MONKEEE (0) (Not Cataloged)', provides a detailed performance dashboard for the MONKEEE instance. It includes sections for Client Apps (Connected: 12, Active: 5, Percent Active: 18%, Percent of Max: 2%), Agents (Registered: 86, Non-Pooled: 100%, Pending Locks: 0%, Pending Token: 0%), Buffer Pool (Overall Hit Rate: 99%, Index Hit Rate: 100%), Cache (Package Cache Hit Rate: 83%, Catalog Cache Hit Rate: 100%), I/O Agents (Page Cleaners: 3, Victim Cleans: 0%, Thresh Cleans: 0%, Prefetchers: 5), External Storage (DMS and SMS utilization and space), and Database Information (Max Connections: 36, Deadlocks Detected: 0, Timeouts: 0, Last Backup: 1 days 20 hrs 13 mins). A yellow callout box points to the DZS7 subsystem in the top window and contains the text: 'Homepage provides complete Overview of DB2 Subsystem or Instance activity Including SYSPLEX and EEE/ESE clustered environments'. The Quest Software logo and 'Not calibrated' text are visible at the bottom right of the dashboard.

Users

Category	Value
Total	11
DDF	3
TSO	0
CICS	0
IMS	0
Call Attach	0
Utility	0
RRSAP	8

Requests

Category	Value
DDF	3.55 DML/s
DDL	0.00 DDL/s
DCL	0.33 DCL/s
Call	0.00 Call/s
Xact	0.85 Xact/s

Client Apps

Category	Value
Connected	12
Active	5
Percent Active	18 %
Percent of Max	2 %

Agents

Category	Value
Registered	86
Non-Pooled	100 %
Pending Locks	0 %
Pending Token	0 %

Buffer Pool

Category	Value
Overall Hit Rate	99 %
Index Hit Rate	100 %

Cache

Category	Value
Package Cache Hit Rate	83 %
Catalog Cache Hit Rate	100 %
Lock List Utilization	2 %

Sorting

Category	Value
Percent of Threshold	46 %
Overflows	0 %

External Storage

Category	Value
DMS Utilization	60 %
Used Space	114.0 MB
Free Space	79.1 MB
SMS Utilization	0 %
Used DB2	20.6 MB
Used Other	0.0 KB
Free Space	0.0 KB

Database Information

Category	Value
Max Connections	36
Deadlocks Detected	0
Timeouts	0
Statistics Reset	
Last Backup	
Total Uptime	1 days 20 hrs 13 mins

Active Log

Category	Value
Utilization	0 %
Secondaries	0

Locking

Category	Value
Escalation rate	0 %
Suspend	0 %

Database Information

Category	Value
Max Connections	36
Deadlocks Detected	0
Timeouts	0
Statistics Reset	
Last Backup	
Total Uptime	1 days 20 hrs 13 mins

Active Log

Category	Value
Utilization	0 %
Secondaries	0

Locking

Category	Value
Escalation rate	0 %
Suspend	0 %

Not calibrated

QUEST SOFTWARE

Complete System statistics Overview

CPU, EDM pool, Log, Locking statistics
Catalog cache, package cache, lock list....

The screenshot displays the Quest Spotlight interface with two main windows open.

Spotlight on DB2-05/390 - DZ57 V7 QSZDB2

System summary - INTERVAL statistics

Interval: **Lifetime** | Period

System Overview | System Statistics | Applications

CPU utilization

Addr space	A5ID	TCB time	SRB time
DBM1		0.00	
DIST			
IRLM		0.00	
MSTR		0.06	
Totals		0.06	

Log manager sta

Description	Count
Reads from output buffer	
Reads from active log	
Reads from archive log	
Archive log read allocations	
Archive log write allocations	
Archive log CIs moved from active log	
Archive log read delays - tape vol conter	
Archive log read delays - resource unav	
Archive log look ahead mounts attempt	
Archive log look ahead mounts failed	

Spotlight on DB2 - MONKEEE (0) (Not Cataloged)

File | Connections | Drilldown | Navigator | Help

11:48:01 AM | MONKEEE (0)

Active database analysis - INTERVAL statistics

Interval: **Lifetime** | Period

Databases | Partitions

Database	Status	Clients Active (%)	Locks Waiting	Bufferpool HR(%)	Pkg Cache HR(%)	Ctlg Cache
MONKEEE	Active	65.22	0	98.86	97.42	

Database Name: MONKEEE | Location: Remote | Server: NT | Status: Active | Spotlight State: Active

Database Sort & Hash Join

Sorts | Hash joins | Active sorts

Avg sort time: 0.02 secs

Instance Sort & Join Exceptions

Post thresh sorts | Post thresh joins | Piped sort rejects

Database Sort & Join Overflow Pct

Sort ovflw | Hash ovflw | Hash sm ovflw | Hash loop

Sort Heap Allocated: 1000K

Instance Sort & Join Exception Pct

Post thresh sorts | Post thresh joins | Piped sort rejects

Application Monitoring

RID pool, SQL, Plan/Package, Parallelism, and authorizations

Spotlight on DB2-OS/390 - DZS7 V7 QS2DB2

File Connections Drilldown Navigator Help

SQL IO

DZS7

System summary - INTERVAL statistics

Interval Lifetime Period

System Overview System Statistics Application Overview Application Statistics Configuration

SQL statistics				
	Event	Count	Per sec	
DML	Commits	32	1.14	
	Rollbacks	0	0.00	
DDL	Incremental binds	0	0.00	
	Runtime reoptimizes	0	0.00	
DCL	Direct row success	0	0.00	
	Direct row, index use	0	0.00	
SQL cache	Direct row, tbs scan L	0	0.00	
	RID list success	0	0.00	
Routines	RID failure, storage	0	0.00	
	RID failure, RDS limit	0	0.00	
Miscellaneous				

Plan and package statistics			
Description	Plans	Packages	
Binds completed	0	0	
BIND ADD cmds	0	0	
BIND REPLACE cmds	0	0	
BIND without plan id (test)	0	0	
Rebind cmds	0	0	
Rebinds attempted	0	0	
Rebinds completed	0	0	
Autobinds attempted	0	0	
Autobinds completed	0	0	
Autobinds failed, resource ID	0	0	

Parallel query statistics	
Description	Count
Max degree of parallelism achieved	0
Parallel groups executed	0
Parallel groups executed as planned	0
Parallel groups executed with reduced degree	0
Revert to seq mode, UPD/DEL cursor	0
Revert to seq mode, no ESA sort	0
Revert to seq mode, buffer shortage	0
Revert to seq mode, enclave unavailable	0
Parallel group plan reformed, sysplex config change	0
Parallel group plan reformed, buffers depleted	0

Authorization management statistics			
Description	Count	Per sec	
Plan auth attempts	32	1.14	
Plan auth successes	32	1.14	
Plan auth successes w/o catalog	0	0.00	
Plan auth successes - granted public	0	0.00	
Package auth successes	13	0.46	
Package auth successes w/o catalog	4	0.14	
Package auth successes - granted put	4	0.14	
Package authid overwrites	0	0.00	
Package entry overwrites	0	0.00	
Routine auth successes	4	0.14	

Thread Activity Monitoring

Detailed display of all thread activity

File Connections Drilldown Navigator Help

12:04:23 PM MONKEE (0)

Client Application analysis - INTERVAL statistics

Interval Lifetime Period Criteria

Database	Application ID	Auth ID	Application Name	Agent ID	Application Status	Status (min)
MONKEEE	0A041454.950D.030311165619	KENNY	Agent.exe	665	Low executing	0.00
Dynamic SQL stmt - SELECT BUTTERS.SUMMARY.ORDERID, BUTTERS.SUMMARY.PRODUCTID, BUTTERS.SUMMARY.QUANTITY FROM BUTTERS.SUMMAI						
MONKEEE	0A041454.930D.030311165612	KENNY	BFactory.exe	663	DB connect completed	63.33
MONKEEE	0A041454.990D.030311165623	KENNY	Agent.exe	669	Low executing	0.00
Dynamic SQL stmt - SELECT PRODUCTID, PRICE FROM QA2.TBT_PRODUCTS WHERE DESCRIPTION = 'WAGON' ORDER BY PRODUCTID ASC						
MONKEEE	0A041454.980D.030311165622	KENNY	Agent.exe	668	Low executing	0.00
Dynamic SQL stmt - SELECT BUTTERS.SUMMARY.ORDERID, BUTTERS.SUMMARY.PRODUCTID, BUTTERS.SUMMARY.QUANTITY FROM BUTTERS.SUMMAI						
MONKEEE	0A041454.970D.030311165621	KENNY	Agent.exe	667	Low executing	0.00
Dynamic SQL stmt - SELECT ORDERID, QUANTITY FROM QA3.TBT_ORDERDETAILS ORDER BY PRODUCTID ASC						

Details Caching Sorting I/O Activity Locking Unit of Work Authorities Statistics

Auth ID: KENNY App Name: Agent.exe Agent ID: 667 Spotlight State: Paused

```
SELECT ORDERID, QUANTITY FROM QA3.TBT_ORDERDETAILS ORDER BY PRODUCTID ASC
```

Statement Details	Value	Statement Activity	Interval	Life of Statement
Statement Type	Dynamic	Sorts, Statement	0	0
Statement Operation	SQL Fetch	Sort Overflows	0	0

Tablespace Detail for L,U,W

Spotlight on DB2 - SODADB (SODADB)

File Connections Drilldown Navigator Help

SQL IO

SODADB

Tablespace analysis - INTERVAL statistics

Interval **Lifetime** Period

Tablespace Name	State	Managed	Size (Mb)	Utilized (%)	Other (%)	Free (%)	Containers	Logical Reads	Avg Pool F
QCTEMPTS	NORMAL	SMS	0	0.00	6.60	93.40	1	0	
SODA_DATA	NORMAL	DMS	7	100.00		0.00	1	64	
SODA_INDX	NORMAL	DMS	4	84.00		16.00	1	64	
SODA_TEMP	NORMAL	SMS	1	0.00	0.00	0.00	1	0	
SODA_UTILITY	NORMAL	DMS	14	52.46		47.54	1	64	
SYSCATSPACE	NORMAL	SMS	19	0.00	6.60	93.40	1	11	
TEMPSPACE1	NORMAL	SMS	0	0.00	6.60	93.40	1	0	
USERSPACE1	NORMAL	SMS	2	0.00	6.60	93.40	1	0	

Details **I/O Activity** Containers Tables **Statistics**

Tablespace Name: SODA_DATA **Spotlight State: Active**

Table Schema	Table Name	Index Tablespace	Columns	Rows Read	Rows Written	Overflow Accesses
DB2ADMIN	BEVERAGE_CLASS	SODA_INDX	3	0	0	0
DB2ADMIN	BEVERAGE_DISTRI...	SODA_INDX	7	0	0	0
DB2ADMIN	DELIVERY_LOCATI...	SODA_INDX	7	0	0	0
DB2ADMIN	DELIVERY_TRUCK	SODA_INDX	7	0	0	0
DB2ADMIN	SODA	SODA_INDX	6	0	0	0
DB2ADMIN	SODA_MANUFACT...	SODA_INDX	7	0	0	0
DB2ADMIN	SODA_NUTRITION...	SODA_INDX	11	0	0	0
DB2ADMIN	SODA_SHIPMENTS	SODA_INDX	3	0	0	0
DB2ADMIN	SODA_SHIPMENT_...	SODA_INDX	3	0	0	0

Bufferpools

Virtual storage for temporarily holding data and IX pages

z/OS

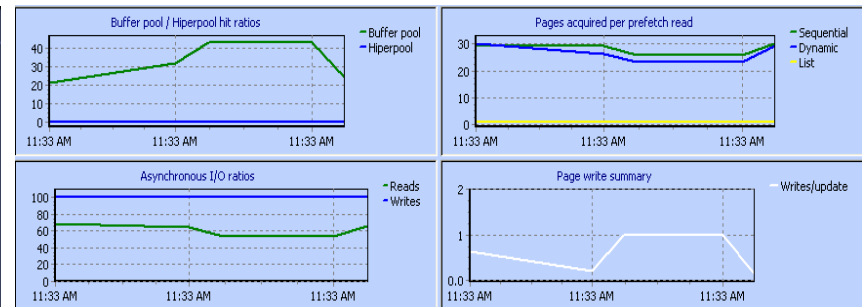
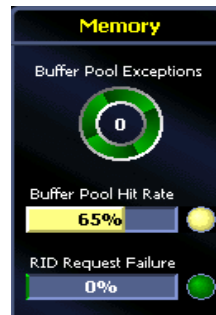
L,U,W

- **Use DBM1 address space**
 - Virtual bufferpools
 - Can define up to 80 BP's
 - Limited to 1.6GB total
- **DBM1 + Hiperpace**
 - Hiperpools
 - Additional 8GB of extended storage
 - “Holding tank” for infrequently updated data
- **MVS Dataspace**
 - Support up to 8M buffers
 - Allows for direct I/O from extended storage
- **Can be used by multiple objects**
- **IBMDEFAULTBP** automatically created with database
 - Additional pools created with DDL
- Uses memory from database shared memory (*database_memory*)
- 4 GB size limit
 - 32-bit machines can use extended storage cache

What to Monitor

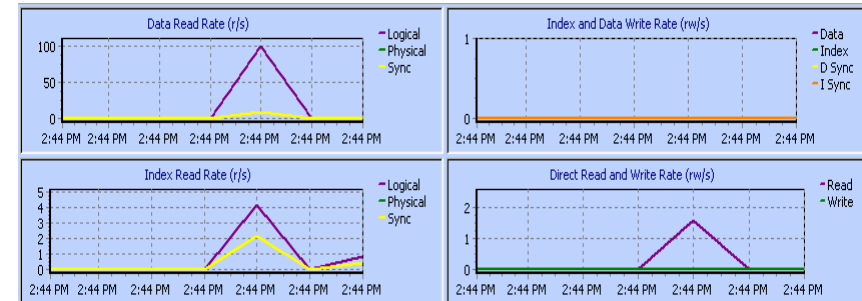
z/OS

- Buffer Pool hit ratio
(GETPAGES – pages read)/GETPAGES
- Hiperpool Hit Ratio
 - #pages read from hiperpool/pages written
- Page Externalization
 - Excessive writes to disk



L,U,W

- Overall Hit ratio
 - Total # data/IX reads by BP
- Data Hit Rate
- Index Hit Rate
- Asynchronous Page Cleaners
 - *Num_iocleaners*



Effective Use of Bufferpools – z/OS

The single biggest performance mistake is to stick everything in BP0!

- DB2 has 80 bufferpools available for a reason!
- Separate bufferpools for:
 - ✓ Catalog and Directory (BP0)
 - ✓ DSNDB07
 - ✓ Tablespaces
 - ✓ Indexes
 - Large VPSIZE
 - More than one
 - ✓ Small, Read-only tables
 - ✓ Large tablespaces w/random processing
 - ✓ Small frequently updated tables and IX's
 - ✓ Test environment for isolating test cases

Maintenance

- Proper maintenance is critical for optimal performance
 - Reorganization
 - Statistics Collection

Reorg

- What causes fragmentation?
 - Insert/Update
 - Check PCTFREE and FREEPAGE
 - VARCHAR fields being updated

Monitoring for TS and Table REORGs

z/OS

- CLUSTER RATIO < 95%
 - SYSINDEXES
 - CLUSTERATIO < 95%
 - SYSINDEXPART
 - FAROFFPOS > 10% of CARD
- Excessive row relocation
 - SYSTABLEPART
 - NEARINDREF+FARINDREF > 10% of CARD
- Excessive extents (>50)
- Excessive drop space
 - Simple TS only
 - PERCDROP > 10%
- LOB tablespaces
 - SYSLOBSTATS
 - ORGRATIO > 2

L,U,W

- Cluster Ratio < 90%
 - SYSCAT.INDEXES
 - CLUSTERRATIO
- Overflow of Rows
 - SYSSTAT.TABLES
 - OVERFLOW
- Fetch Statistics
 - SYSCAT.INDEXES
 - Small # of
 - AVERAGE_SEQUENCE_FETCH_PAGES
 - Growth of AVERAGE_RANDOM_FETCH_PAGES
- Empty Pages
 - SYSCAT.TABLES
 - FPAGES-NPAGES

Monitoring for Index REORGs

z/OS

- Excessive distance between LEAF pages
 - SYSINDEXPART
 - LEAFDIST > 200
 - Can cause Pre-fetch to be disabled
 - Should be monitored for growth
 - LEAFFAR > 10% of NLEAF (SYSINDEXES)
 - Excessive extents (>50)

L,U,W

- # of LEAF pages
 - SYSCAT.INDEXES
 - NLEAF
- Low Cluster Ratio
 - SYSCAT.INDEXES
 - CLUSTERRATIO

Statistics

Accurate statistics are a critical factor for performance monitoring and tuning

RUNSTATS provides statistical information for:

1. Optimization of SQL
2. Monitoring status of objects

RUNSTATS

- When to run RUNSTATS:
 - ✓ After LOAD, REORG, and REBUILD IX
 - ✓ After creating new index
 - ✓ After heavy insert, update, delete activity

SPACE MANAGEMENT with Quest Central

- Improve overall response time
 - Detect table and index fragmentation
 - Advice on what needs to be done
 - Provides immediate resolution
- Reduce risk of database outage
 - Detect out of space conditions
 - Forecast space usage
 - Calculator to predict space usage
- Utility Generation

Quest Central Space Exceptions

Quest Central for DB2

File Edit View Tools Window Help

Exploring Databases on qsc1-DSC7 (jwankow)

Database Name Database Type Storage Group Tablespace Buff... Index Bufferpool Database Identifier Database Cre

Storage Exceptions on qsc1-SUBSYS-DSC7

Space Usage Tablespace Reorg Index Space Usage Index Reorg Index Cardinality

Description: Space Usage is an exception report identifying tablespaces with space problems. Tablespaces identified here can be altered using Space Management's DDL features.

Criteria:

Severity	PQTY Used %	Extents
Severe	90	100
Warning	80	15

Edit...

Severity	TS Name	DB Name	Part	Card	PQTY Used %	Extents	SQTY Used %	Priqty Pg	Secqty Pg	Spa
Severe	BASEBALL	QCDEMODB	0	207740	100	3	1	1800	1800	409
Severe	DSNRFUNC	QCDEMODB	0	-1	100	1	2	3	3	12
Severe	DSNRSTAT	QCDEMODB	0	-1	100	1	2	3	3	12
	PLANRTAB	QCDEMODB	0	-1						12
	QC20S71D	QCDEMODB	0	14						12
	QC20S71E	QCDEMODB	1	33						12
	QC20S71E	QCDEMODB	2	0						12
	QC20S71E	QCDEMODB	3	11						12
	QC20S71E	QCDEMODB	4	0						12
	QC20S71R	QCDEMODB	0	-1	100	1	1	5	5	12
	QC20S71S	QCDEMODB	0	0	100	1	1	5	5	12

Advice

Quest Central ADVICE

Alter tablespace to reduce Priqty used to 80%.

Tablespace QCDEMODB.BASEBALL has Priqty used = 100% (> criteria 80%). Alter tablespace to Priqty 20490 Kb to achieve Priqty utilization of 80%.

Identify tablespaces and IX's low on freespace, excessive extents, or in need of reorg

Quest Central Space Exceptions

Storage Exceptions on 10.4.23.108-SUBSYS-DZS7

Space Usage | Tablespace Reorg | Index Space Usage | Index Reorg | Index Cardinality

Description: Index Reorg is an exception report identifying indexes that need to be reorganized. Indexes identified here can be altered using Space Management's DDL features.

Criteria:

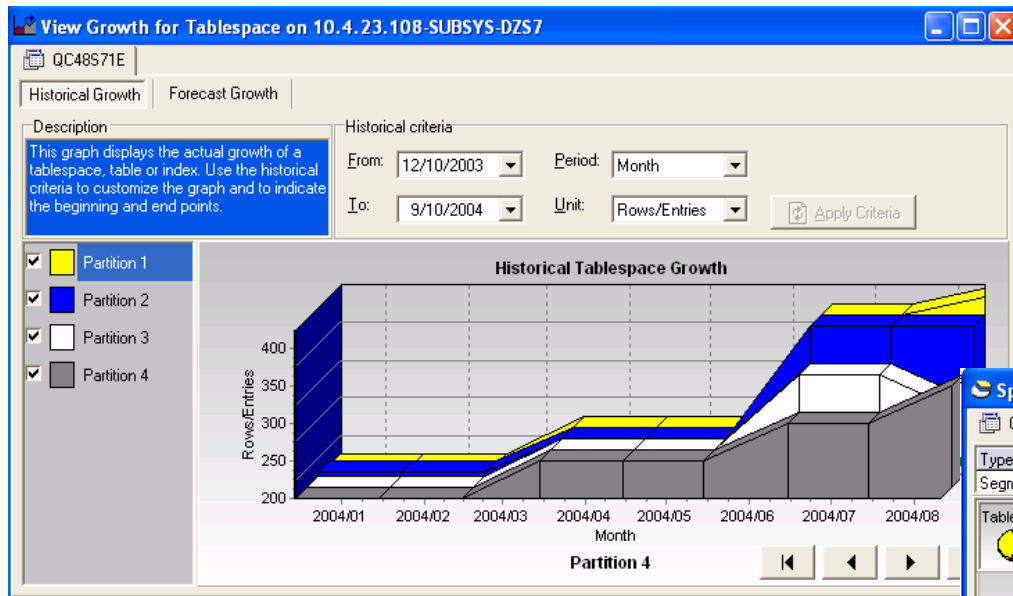
Severity	LEAFDIST	OFFPOS %	NLEVELS	Pseudo Delete %
Severe	90	0	20	10
Warning	80	80	10	

Edit...

Severity	IX Creator	IX Name	Part	LEAFDIST	OFFPOS %	NLEVELS	Pseudo Delete %	Card	Clustering	Clustered	NEAROFFPOS	FAROF
Severe	QC48	XEMP2	0	0	25	2	N/A	44	N	N	0	11
Severe	Q	Properties...	0	0	39	2	N/A	160	N	N	55	8
	Q	Utilities						18	N	Y	0	0
		Collect Statistics...										
		Reorg...										
	QC48	XACT				2	N/A	18	N	Y	0	0
	QC48	XCONA1	0	0		0	N/A	0	N	N	0	0
	QC48	XDEPT1	0	0		2	N/A	14	N	Y	0	0
	QC48	XDEPT2	0	0		2	N/A	14	N	Y	0	0
	QC48	XDEPT3	0	0		2	N/A					0
	QC48	XDSPTXT1	0	0		2	N/A					0
	QC48	XEMPPROJ...	0	0		2	N/A					0
	QC48	XEMPPROJ...	0	0		2	N/A					0
	QC48	XEMP1	1	0		2	N/A					0

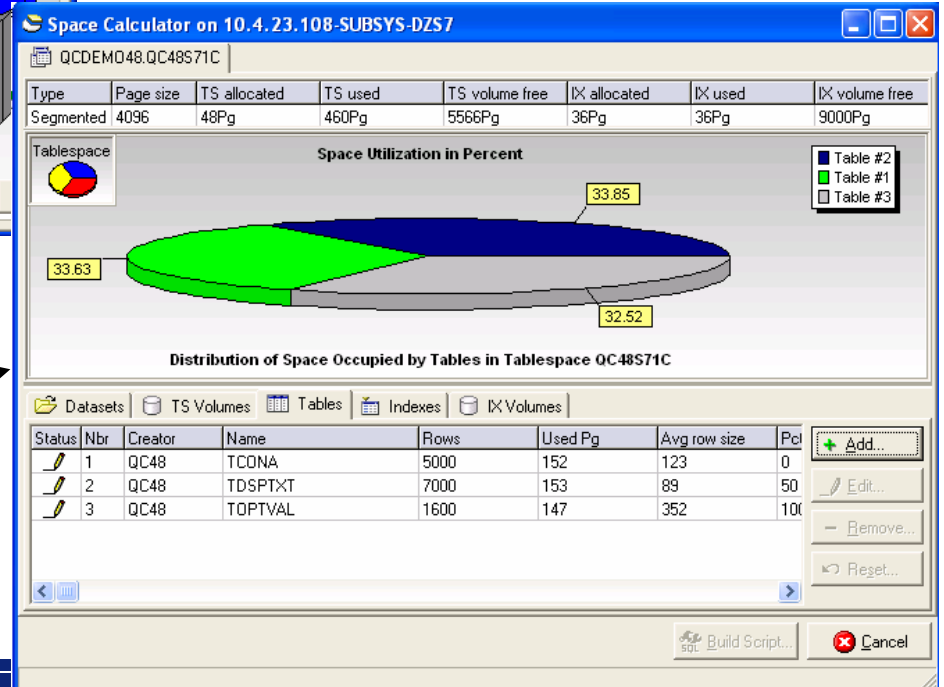
Generate Reorg Directly from Report

Quest Central Space Reporting



Historical Growth Reporting

Detailed space Utilization and "What-If" Analysis



Quest Central Utility Generation

The screenshot shows the Quest Central for DB2 interface. The main window displays a tree view on the left and a table of tablespaces in the center. A context menu is open over the table, with the 'Utilities' option selected. A yellow box with the text 'Wizards to build all Utilities Or...' is overlaid on the menu, with an arrow pointing to the 'Reorg Tablespace' dialog box that is open in the foreground.

Wizards to build all Utilities Or...

Reorg Tablespace on 10.4.23.103-DSC7

Tablespace Name	Database Name	Partitions
BASEBALL	QCDEMO045	N/A

Table Statistics

Table Creator	Table Name
<input checked="" type="checkbox"/> QC31	ALLSTARS
<input checked="" type="checkbox"/> QC31	AWARDS
<input checked="" type="checkbox"/> QC31	BATTING
<input checked="" type="checkbox"/> QC31	FIELDING
<input type="checkbox"/> QC31	HOF
<input type="checkbox"/> QC31	HOF1

Unload options:

- Complete Reorg
- Unload and leave status
- Unload and remove
- Unload external and remove status

Miscellaneous options:

- Log records during reorg
- Keep dictionary
- Fast switch

Criteria options:

- Offset position limit:
- Indirect reference limit:
- Drop limit:

Buttons: Build Script... Cancel

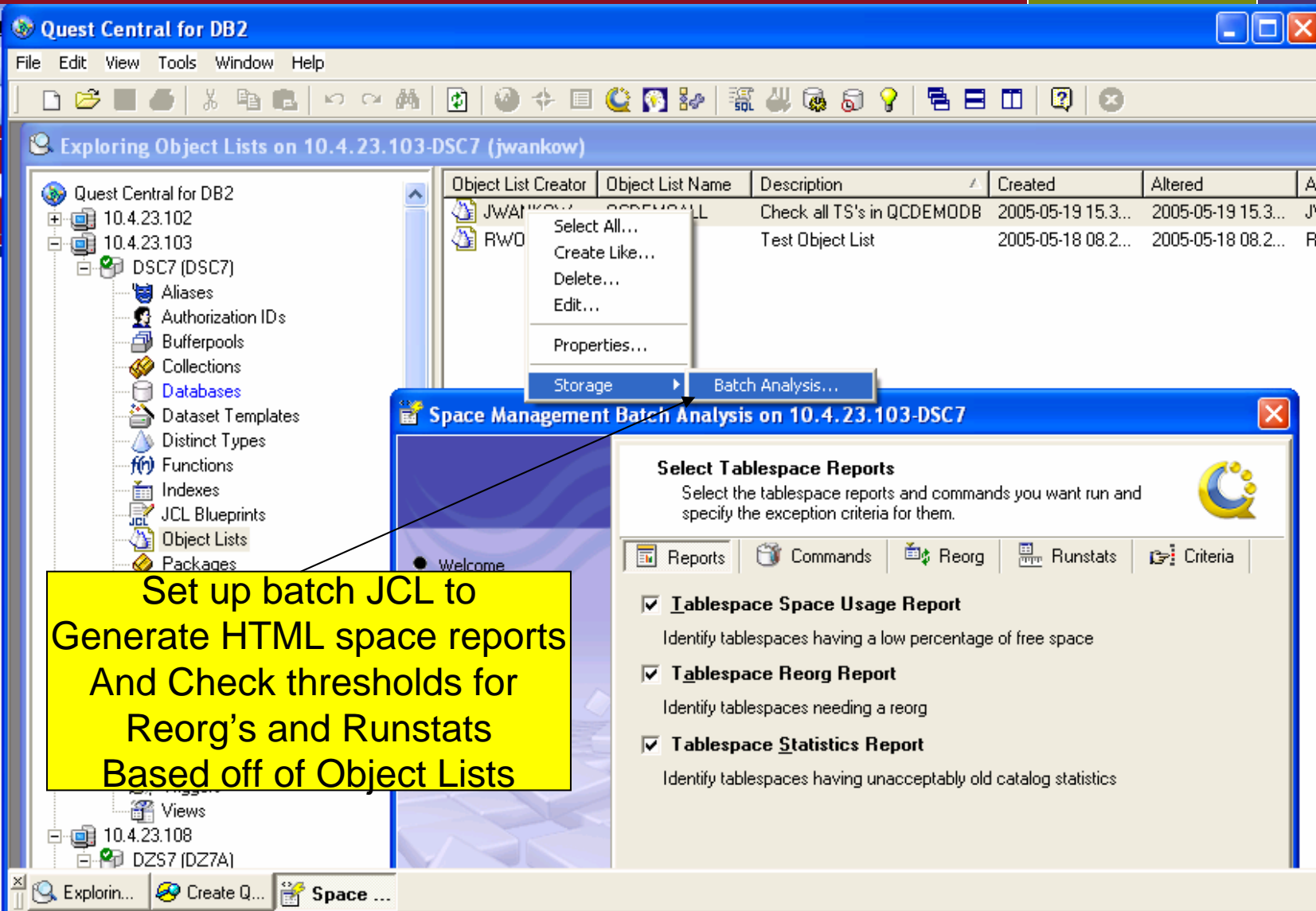
Automated Utility Generation

The screenshot displays the Quest Central for DB2 application window. The main interface is titled "Exploring Object Lists on 10.4.23.103-DSC7 (jwankow)". On the left, a tree view shows the database structure, with "Object Lists" selected. A context menu is open over the "Object Lists" folder, with "Properties..." highlighted. A "Properties of Object List on 10.4.23.103-DSC7" dialog box is open, showing the "Object List Name" as "QCDEMOALL" and the "Description" as "Check all TS's in QCDEMODB". Below the dialog, a list of tablespaces is shown, including QCDEMODB.BASEBALL, QCDEMODB.DSNRFUNC, QCDEMODB.DSNRSTAT, QCDEMODB.PLANRTAB, and several QC31S71* tablespaces. A yellow callout box with the text "Object lists define Objects to be checked" has arrows pointing to the "Object Lists" folder in the tree and the "Tablespaces" list in the dialog.

Object List Creator	Object List Name	Description	Created	Altered	Al
	QCDEMOALL	Check all TS's in QCDEMODB	2005-05-19 15.3...	2005-05-19 15.3...	JV
				8 08.2...	R

Object lists define Objects to be checked

Batch Space Analysis



The screenshot shows the Quest Central for DB2 interface. The main window displays a tree view of object lists for server 10.4.23.103. A context menu is open over the 'Object Lists' folder, with the 'Storage' option selected, leading to a 'Batch Analysis...' dialog box. The dialog box is titled 'Space Management Batch Analysis on 10.4.23.103-DSC7' and contains a 'Select Tablespace Reports' section with three checked options: 'Tablespace Space Usage Report', 'Tablespace Reorg Report', and 'Tablespace Statistics Report'. A yellow callout box points to the 'Batch Analysis...' option in the context menu.

Object List Creator	Object List Name	Description	Created	Altered	AI
JWANKOW	QCDEMOALL	Check all TS's in QCDEMODB	2005-05-19 15.3...	2005-05-19 15.3...	JV
RWO	Test Object List		2005-05-18 08.2...	2005-05-18 08.2...	R

Set up batch JCL to
Generate HTML space reports
And Check thresholds for
Reorg's and Runstats
Based off of Object Lists

Application Design

- SQL Design considerations
- Optimization Hints

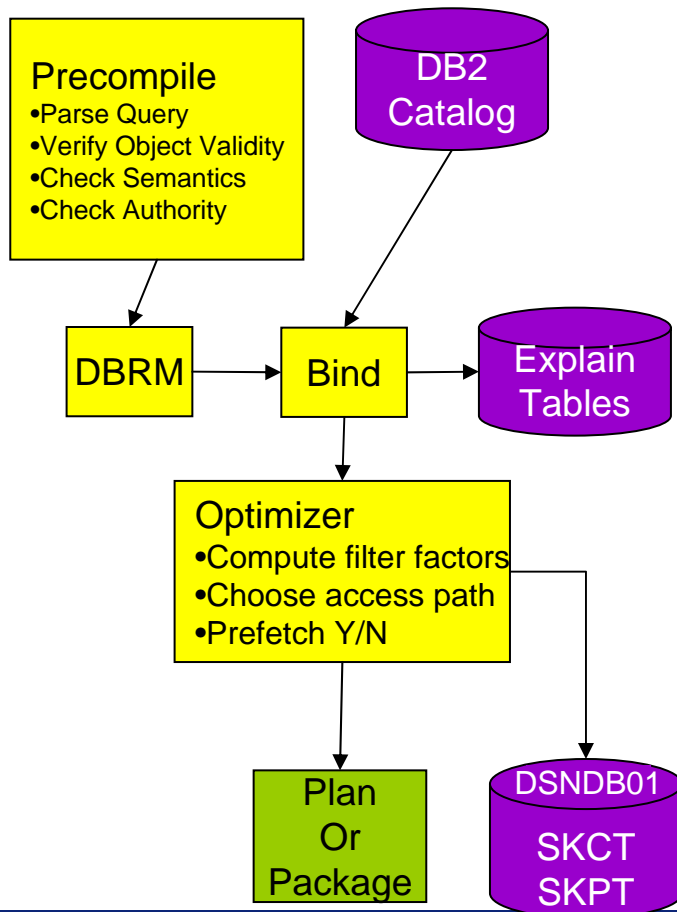
Efficient application design is the single most important aspect of an efficiently performing subsystem

SQL Coding Factors

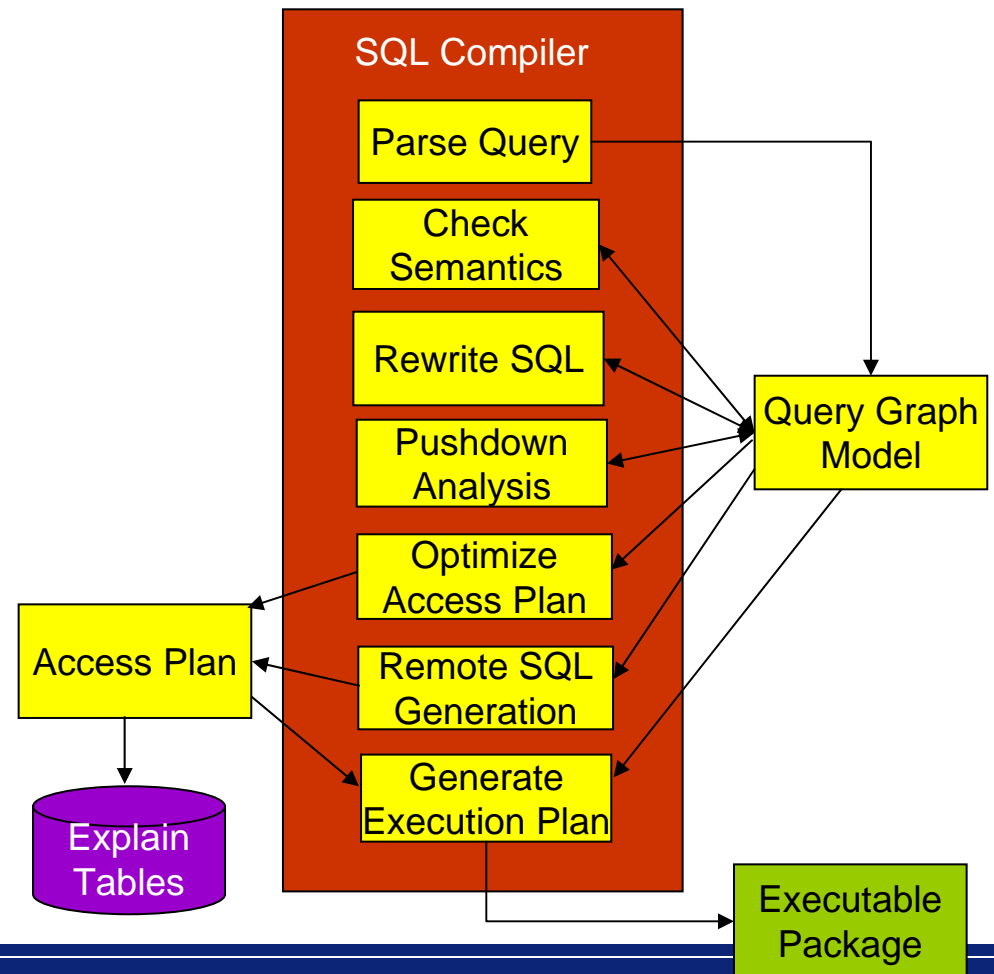
- Many ways to write a SQL to return the same data
- Small differences in coding SQL can have great performance implications
- Different SQL versions may produce different access plans

SQL Optimization

z/OS



L,U,W



Optimization Tips – z/OS & LUW

- Make sure statistics are accurate
- Use stage 1 vs. stage 2 predicates
- Only select required columns
 - Avoid SELECT *
- Keep predicates as restrictive as possible
 - Minimize # rows returned
 - Minimize program filtering and let DB2 do the work
- Order predicates from most to least restrictive
- Avoid sorts
 - ASC/DESC indexes can help avoid excessive sorting
- Avoid UNION clause
 - CASE expression more efficient
- Use Appropriate Optimizer Class (LUW)
 - 0 through 9 (5 default)
 - Use lower class for Dynamic SQL
 - Higher class for Static
- Use “Optimize for n Rows”
 - Minimizes optimization cost
- Use “Fetch First n Rows”
- Avoid Data Type Conversions

EXPLAIN!!!

Optimizer Class

- DB2 Optimizer Class
 - Values are between 0 and 9, default is 5
 - Determines the intensity used by the DB2 SQL Compiler when rewriting SQL
 - Dynamic SQL can't spend time optimizing, use lower class
 - Static SQL optimizes once, use a higher class
 - "dft_queryopt" database setting
 - SET CURRENT QUERY OPTIMIZATION n

Level	Recommendation
0	Minimal amount of optimization. Only recommended for very simple SQL accessing well indexed tables. Only nested loop joins and IX scans enabled.
1	Similar to 0 except Merge Scan and TS scan enabled.
2	Recommended for very complex queries which are infrequently executed in a decision support or OLAP environment.
3	Closest to OS/390 optimizer. Recommended for queries with 4 or more joins.
5	DEFAULT – Most cost effective method for mix of simple and complex queries. Optimization will be automatically reduced for complex dynamic SQL if optimizer determines that the resources are not necessary.
7	Same as 5 except optimization not reduced for complex dynamic SQL
9	Used to determine whether more comprehensive optimization can generate better access plan for very complex long running queries using large tables

Quest's SQL Tuning Solutions

SQL Optimizer

- Quest SQL Optimizer for DB2 UDB
 - Analyzes application code and identifies SQL most in need of tuning
 - AI based Automated SQL write
 - Benchmark SQL statements to find the most efficient statement
 - Index advisor and usage analysis
- Standalone Product today
 - Merged into Quest Central in Q4, 2005

Built in Database Explorer

The screenshot displays the Quest SQL Optimizer for DB2 UDB - [Database Explorer] interface. The left pane shows a tree view of Database Objects, including Schemas, Tables, Summary Tables, Tablespace, Views, Aliases, Nicknames, Procedures, Functions, Triggers, and Packages. The right pane shows the Object Information for a selected object, displaying SQL code. A yellow callout box highlights the text: "Database explorer provides Easy catalog access to review Objects and SQL".

Database Objects

- Schemas
- Tables
- Summary Tables
- Tablespaces
- Views
- Aliases
- Nicknames
- Procedures
- Functions
- Triggers
 - TRI_DPT_DEL
 - TRI_DPT_INS
 - TRI_DPT_UPD
- Packages
 - P1418650
 - P1448340
 - P1500310
 - P1517980
 - P1534400
 - P1557250
 - PACK1
 - PACK2
 - PACK3
 - NULLID.SQLA1D01
 - NULLID.SQLA2D01
 - NULLID.SQLA3D01
 - NULLID.SQLA4D01
 - NULLID.SQLA5D01

Object Information

```

1  ROLLBACK TO SAVEPOINT
2
3  SELECT *
4  FROM EMPLOYEE
5  WHERE NULLIF (EMP_SALARY,
6  :H00016 :H00017 ) BETWEEN 49000
7  AND 50000
8  ORDER BY EMP_ID,
9  EMP_SALARY
10
11 SELECT EMP_ID,
12 EMP_NAME,
13 EMP_ADDRESS,
14 EMP_DEPT,
15 EMP_SALARY
16 FROM EMPLOYEE
17 WHERE EMP_SALARY
18 AND 70000
19
20 SELECT CASE WHEN ( :H00012 :H00013 IS NULL ) THEN 1 ELSE 0 E
21 FROM (VALUES 1) AS X
22
23 UPDATE EMPLOYEE
24 SET EMP_SALARY = 50000
25 WHERE EMP_ID = :H00004 :H00005
    
```

Schema: DB2ADMIN Login: db2admin Database Alias: DEMO1

SQL Scanner

The screenshot displays the Quest SQL Optimizer for DB2 UDB interface. At the top, the title bar reads "Quest SQL Optimizer for DB2 UDB". Below it is a menu bar with options: File, Edit, Search, Report, Job, SQL, View, Database, Tools, Window, Help. A toolbar with various icons is positioned below the menu bar.

The "Job Manager" window shows a table with the following data:

File / Database Object	Status	Valid SQL	Problematic S...	Complex SQL	Simple SQL	File Si...
DB2ADMIN->Procedures->MY_PROC ()	2 SQL Scanned	1	0	0	1	771
DB2ADMIN->Procedures->PROCEDURE_A ()	4 SQL Scanned	4	0	0	4	1386
DB2ADMIN->Procedures->PROCEDURE_B ()	2 SQL Scanned	2	0	0	2	941
DB2ADMIN->Procedures->PROCEDURE_C (IN VARCH...	3 SQL Scanned	3	0	2	1	1461
DB2ADMIN->Procedures->PROCEDURE_D (IN VARCH...	1 SQL Scanned	1	0	0	1	837

The "Scanned SQL Viewer [All SQL]" window shows the following SQL code:

```

DB2ADMIN->Views->CUSTOMER_PROMOTION
Scanned SQL
1 select *
2   from customers CUSTOMER
3   where EXISTS (SELECT 'X'
4                 FROM invoice_customer INV
5                 WHERE promotion_id = any (select pro
6
7
8
9
10
11
12
13
14

```

The "SQL Information" window displays an Access Plan Graph with the following components:

- FETCH (2) 15019.12
- FETCH (4) 177.60
- CUSTOMERS
- GRPBY (9) 210.31
- BSCAN (10) 210.31

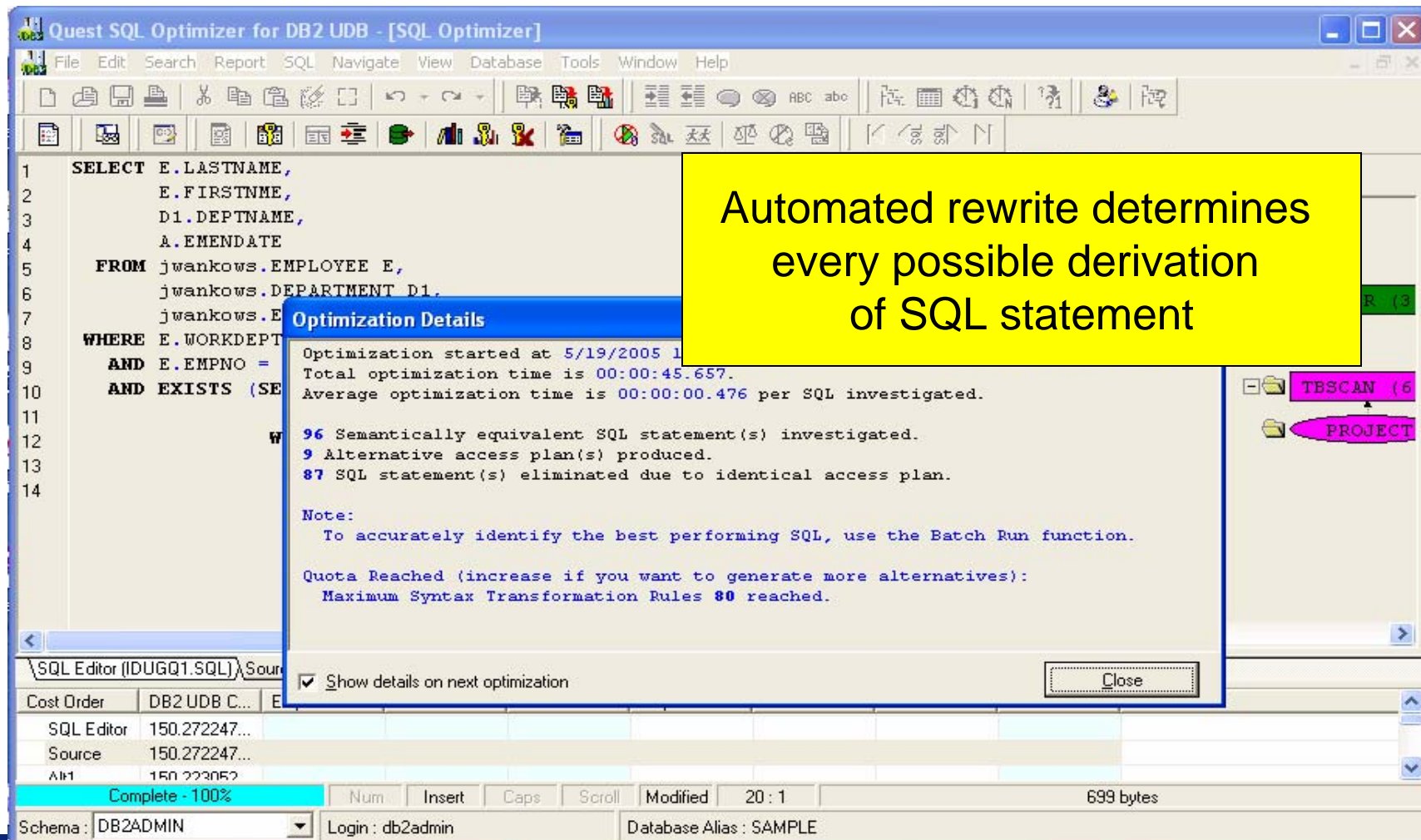
The "Information" window shows the following text:

Problematic SQL Statement
 4 Table operations found in access plan exceeding the upper limit of the Complex SQL Table Range.
 (DB2ADMIN.CUSTOMERS,DB2ADMIN.CUSTOMERS,DB2ADMIN.PROMOTIONS,DB2ADMIN.CUSTOMERS)

At the bottom of the interface, the "Schema" is set to "DB2ADMIN", "Login" is "db2admin", and "Database Alias" is "DEM01".

Analyzes application code and identifies problematic SQL

Automated SQL Rewrite



The screenshot displays the Quest SQL Optimizer for DB2 UDB interface. The main window shows an SQL statement being optimized:

```
1 SELECT E.LASTNAME,  
2 E.FIRSTNAME,  
3 D1.DEPTNAME,  
4 A.EMENDATE  
5 FROM jwankows.EMPLOYEE E,  
6 jwankows.DEPARTMENT D1,  
7 jwankows.EMPLOYEE A  
8 WHERE E.WORKDEPT = D1.DEPTNAME  
9 AND E.EMPNO = A.EMPNO  
10 AND EXISTS (SELECT * FROM jwankows.EMPLOYEE B  
11 WHERE B.LASTNAME = E.LASTNAME  
12 AND B.FIRSTNAME = E.FIRSTNAME  
13 AND B.EMPNO = A.EMPNO  
14 AND B.WORKDEPT = D1.DEPTNAME)
```

An "Optimization Details" dialog box is open, providing the following information:

- Optimization started at 5/19/2005 1:00:00 PM
- Total optimization time is 00:00:45.657.
- Average optimization time is 00:00:00.476 per SQL investigated.
- 96 Semantically equivalent SQL statement(s) investigated.
- 9 Alternative access plan(s) produced.
- 87 SQL statement(s) eliminated due to identical access plan.

Note:
To accurately identify the best performing SQL, use the Batch Run function.

Quota Reached (increase if you want to generate more alternatives):
Maximum Syntax Transformation Rules 80 reached.

Show details on next optimization

Close

The interface also shows a file explorer on the right with folders for "TBSCAN (6)" and "PROJECT". At the bottom, a status bar indicates "Complete - 100%", "Schema: DB2ADMIN", "Login: db2admin", and "Database Alias: SAMPLE".

Automated rewrite determines every possible derivation of SQL statement

Benchmark SQL Rewrites

Benchmarks SQL rewrites and identifies optimal SQL syntax

```

1  SELECT E.LASTNAME,
2         E.FIRSTNME,
3         D1.DEPTNAME,
4         A.EMENDATE
5  FROM  jwankows.EMPLOYEE E,
6        jwankows.DEPARTMENT D1,
7        jwankows.EMP_ACT A
8  WHERE E.WORKDEPT = D1.DEPTNO
    
```

Cost Order	DB2 UDB C...	Elapsed Tim...	Times of Im...	Records Re...	Elapsed Tim...	Times of Im...	Records Re...	Remark
SQL Editor	150.272247...							
Source	150.272247...	00:00:00.61...		75				2nd Run time
Alt1	150.223052...	00:00:00.10...	6.10	75				
Alt2	150.256164...	00:00:00.60...	1.02	75				
Alt3	150.472030...	00:00:00.60...	1.02	75				
Alt4	150.491363...	00:00:00.61...		75				
Alt5	150.507537...	00:00:00.60...	1.02	75				
Alt6	151.331558...	00:00:00.60...	1.02	75				
Alt7	151.347640...	00:00:00.60...	1.02	75				
Alt8	152.353546...	00:00:00.90...		75				
Alt9	153.003509...	00:00:00.90...		75				

Schema: DB2ADMIN Login: db2admin Database Alias: SAMPLE

Index Advisor

Identifies new index scenarios to influence SQL performance Using simulated indexes.

Quest SQL Optimizer for DB2 UDB - [Index Expert]

File Edit Search Report SQL Navigate View Database Tools Window

	Schema	Table	Columns	Index	Selectivity	Cl
DB	"DB2A...	"EMP_ACT"	"PROJNO" DESC	"IDX050118105555000"	N/A	CI
DB	"DB2A...	"PROJECT"	"PROJNO", "PRSTAFF"	"QIDX93004655"	N/A	--
DB	"DB2A...	"EMP_ACT"	"EMENDATE", "EMPNO"	"QIDX167422257"	N/A	--
DB	"DB2A...	"EMPLOYEE"	"FIRSTNAME", "LASTNAME", "EMPNO", "WORKDEPT"	"QIDX120068666"	N/A	--
DB	"DB2A...	"DEPARTMENT"	"DEPTNO", "DEPTNAME"	"QIDX153243280"	N/A	--

SQL Editor | Index List | Time | Summary

```

1 CREATE INDEX IDX050118105555000 ON
2 "CFERNAND"."EMP_ACT"
3 ("PROJNO" DESC)
4 ALLOW REVERSE SCANS
5 ;
6 -- Index Name IDX050118105555000 is replac
7 CREATE INDEX "QIDX93004655" ON
8 "CFERNAND"."PROJECT"
9 ("PROJNO" ASC, "PRSTAFF" ASC)
10 ALLOW REVERSE SCANS
11 ;
12 -- Index Name IDX050118105555000 is replac

```

Used Index: DB2UDB

Num Insert Caps Scroll Modified 15:1 425 bytes

Schema: CFERNAND Login: db2admin Database Alias: SAMPLE

Index Usage Analysis

Identifies unused indexes, most frequently accessed tables and indexes

Quest SQL Optimizer for DB2 UDB - [Index Usage Analyzer]

File Edit Search Report Analysis View Database Tools Window Help

INDEX USAGE ANALYZER

- ANALYZER1
 - Tables Analyzed
 - SQL Analyzed
- ANALYZER2
 - Tables Analyzed
 - SYSIBM.SYSCOLUMNS
 - SYSIBM.SYSTABLES
 - SYSIBM.SYSTABLESPACES
 - SYSIBM.SYSXMLRELATIONSHIPS
 - SQL Analyzed
 - SQL\demo\NULLID->Packages->SQLUAE05\
 - SQL\demo\NULLID->Packages->SQLUFE03\
 - SQL\demo\NULLID->Packages->SQLUFE03\
 - SQL\demo\NULLID->Packages->SQLUHE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SQLUJE00\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\demo\NULLID->Packages->SYSSTAT\
 - SQL\test\SYSFUN->Functions->OID2PATH\

Total Indexes = 17

82.35 %

17.65 %

3 Used 14 Unused

Properties Index Summary

Unused	Table	Index	Index Key
	SYSIBM."SYSTABLES"	IBM00	+CREATOR+NA...
▶		IBM21	+TBSpace
▶		IBM22	+INDEX_TBSP...
▶		IBM23	+LONG_TBSPA...
▶		IBM78	+ROWTYPESE...
▶		IBM137	+TID+FID
▶	SYSIBM."SYSTABLESPA...	IBM49	+TBSpace
▶		IBM50	+DEFINER
▶		IBM65	+NGNAME
▶		IBM66	+TBSpaceID
▶		IBM70	+RIFFERPOOLID

Schema: CFERNAND Login: db2admin Database Alias: SAMPLE

SQL TUNING with Quest Central

- Improve overall response time
 - Provide SQL advice
 - Compare variations of the same SQL
 - Virtual index for “What-if” testing (Unix/Windows)
- Reduce risk of database outage
 - Detect SQL with high resource usage
- Reduce risk of data loss
 - Compare data

Quest Central for DB2 SQL TUNING

**Analyze and tune SQL
using multiple scenarios**

**Virtual index
For "What-if" testing
(Distributed Only)**

Object name	Type	Rows	Pages
DB2ADMIN.MASTER	Table	15350	260
DB2ADMIN.BATTING	Table	78881	1951

Index name	Leaf	Le
DB2ADMIN.BATTING_U...	63	3
BATTINGIX	55	3

- Create Virtual Index...
- Recommend Virtual Indexes...**
- Create...

Column name	Data type	User type	Avg col ler
RECNUM	INTEGER (4)	No	4
LAHMANID	VARCHAR (12)	No	13
DATE_YYYY	SMALLINT (2)	No	2
TEAM_ID	VARCHAR (3)	No	7
LEAGUE_ID	VARCHAR (3)	No	5
GAME_QTY	SMALLINT (2)	No	3

SQL Tuning on JWANKOWSKI-DB2-BASEBALL

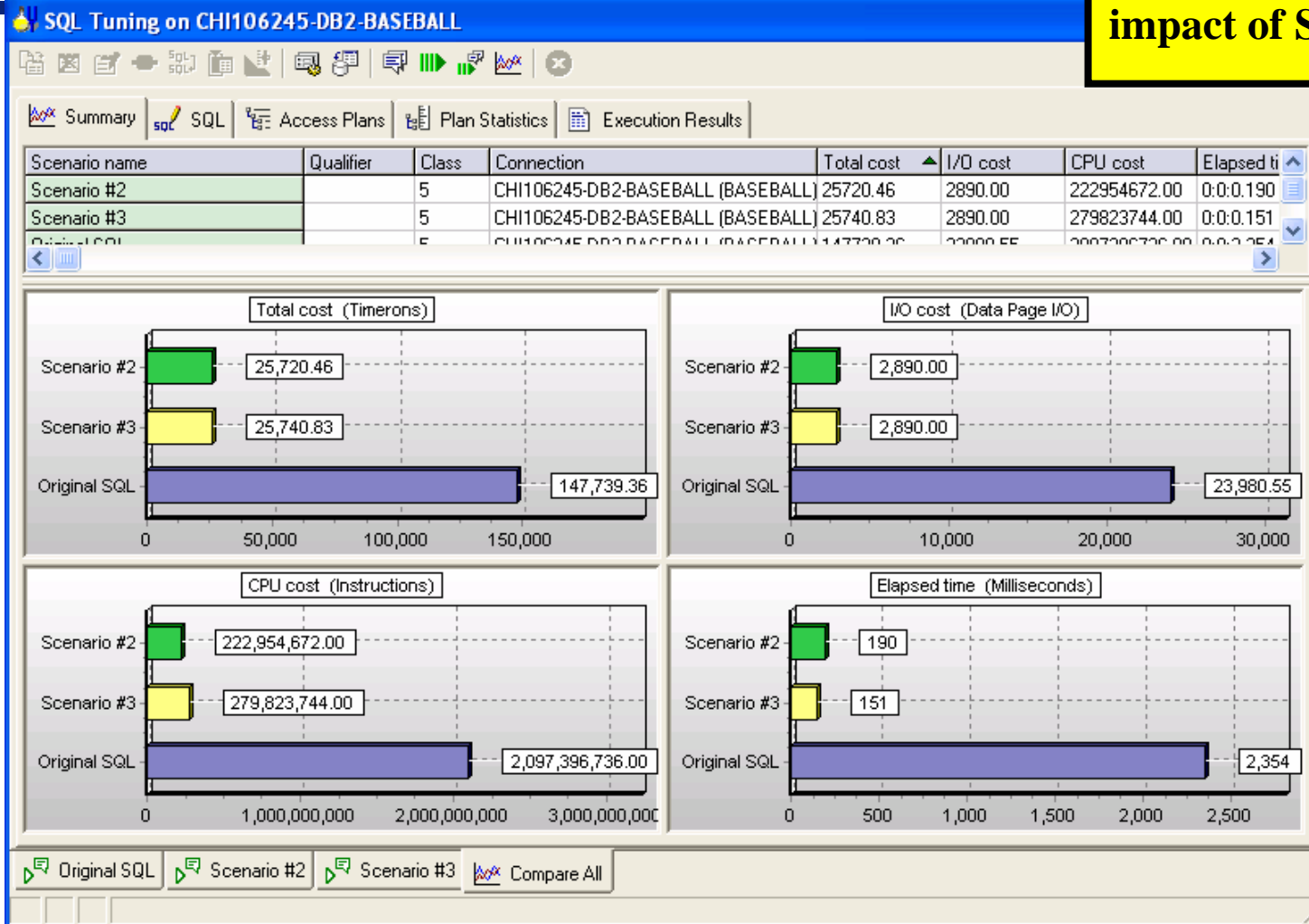
Access Plan | Plan Statistics | Plan Dependencies | Execution Results | Advice

Original SQL

Class: 5 | Qualifier: db2admin | Connection: JWANKOWSKI-DB2-BASEBALL (BASEBALL)

Scenario Compare

Compare scenarios to see impact of SQL changes



Quest Central for DB2 SQL Analysis

DB2 L,U,W

Analyze SQL Collected from JWANKOWSKI-DB2-BASEBALL

Collection Criteria
 Creator: * [dropdown] Name: * [dropdown] Status: All [dropdown] Start Date: 5/30/2003 [dropdown] Refresh

Name	Status	Job ID	Start time	End time	Ext. Analysis	Duration
db2admin.Collection 2	Complete	8	2003-05-02 04:55:25.708734	2003-05-02 05:07:11.453608	Complete	00:15
db2admin.Demo Run	Complete	1	2003-03-26 16:58:58.083157	2003-03-26 17:13:49.056003	Complete	00:15

Select a collection to analyze

Summary of SQL activity For the collection selected.

Statement Statistics for db2admin.Demo Run

Statement Type	Count
Statements	1,184
Selects	1,114
Updates	0
Inserts	0
Deletes	70

Percent of Total

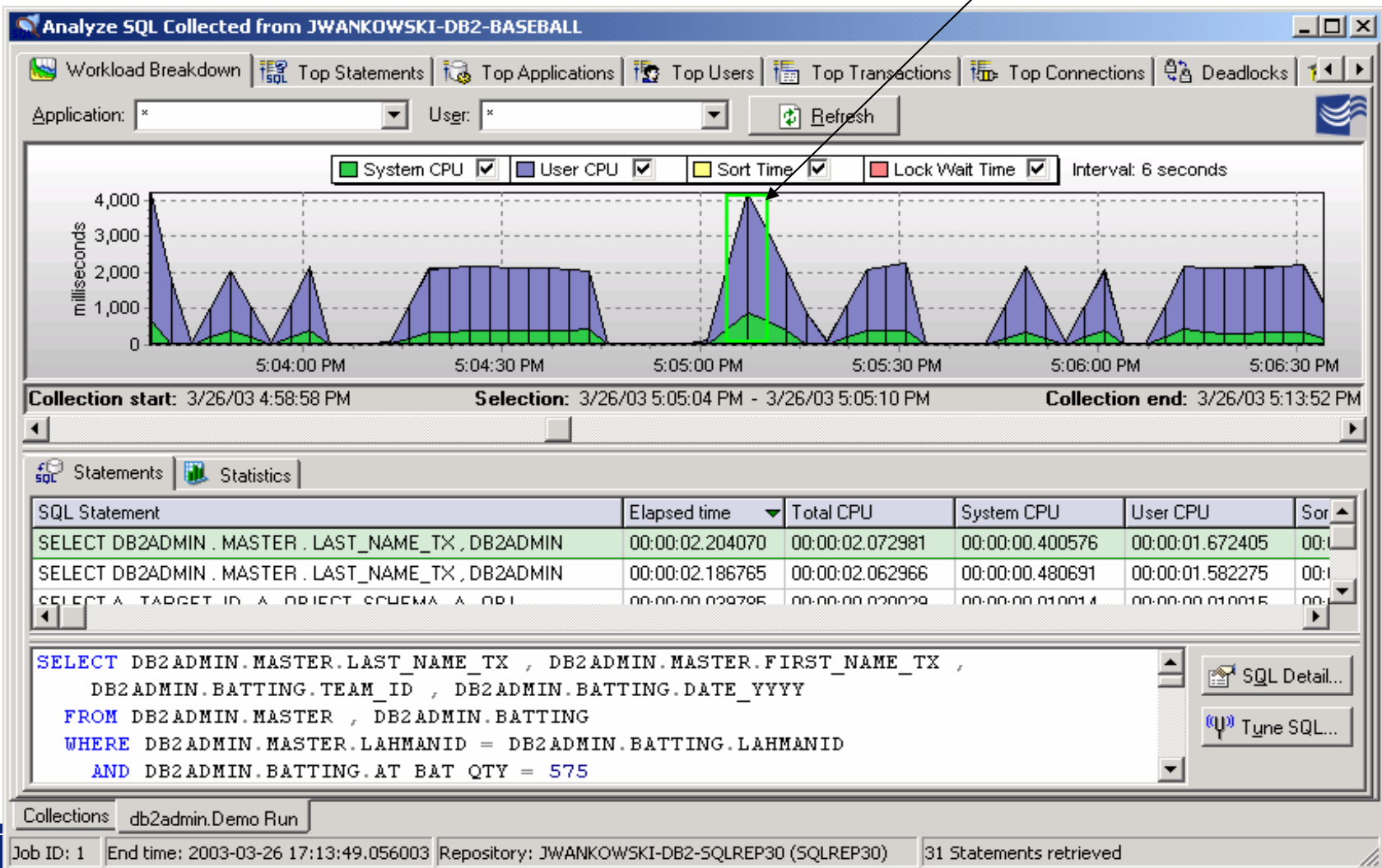
Statement Type	Percent of Total
Selects	94%
Deletes	0%
Inserts	0%
Updates	0%

Statement Statistics | Workload Statistics | Row Statistics | Lock Statistics | Sort Statistics | Status

Collections
 Repository: JWANKOWSKI-DB2-SQLREP30 (SQLREP30) 2 Collections retrieved

Workload Breakdown

Workload breakdown allows you to pinpoint performance spikes and focus in on activity during that time period



Detailed SQL Analysis

The screenshot displays the 'Analyze SQL Collected from JWANKOWSKI-DB2-BASEBALL' window. The main table lists SQL statements sorted by Total CPU. The top statement is a SELECT query involving DB2ADMIN.MASTER.LAST_NAME_TX and DB2ADMIN.BATTING. A yellow callout box points to the 'Top Statements', 'Top Applications', 'Top Users', and 'Top Transactions' tabs, stating: 'Drilldowns allow you to examine historical data From different points of view'.

SQL Statement	Total CPU	System CPU	User CPU
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.213182	00:00:00.440633	00:00:01.772549
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.183139	00:00:00.450648	00:00:01.732491
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.173125	00:00:00.390562	00:00:01.782563
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.153096	00:00:00.390562	00:00:01.762534
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.133067	00:00:00.400576	00:00:01.732491
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.082996	00:00:00.350504	00:00:01.732492
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.082996	00:00:00.400576	00:00:01.682420
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.072981	00:00:00.400576	00:00:01.672405
SELECT DB2ADMIN.MASTER.LAST_NAME_TX, DB2ADMIN	00:00:02.072980	00:00:00.340489	00:00:01.732491

SQL Statement Detail	
User ID	JWANKOWSKI
Auth ID	DB2ADMIN
Application	QuestCentral.exe
	SQLLF000
	4
	SQLCUR4
Application ID	*LOCAL.DB2.030326225904
Agent ID	23
Elapsed time	00:00:05.725966

Drilldowns allow you to examine historical data From different points of view

Operations: PREPARE, OPEN, CLOSE

Collections: db2admin.Demo Run

Job ID: 1 | End time: 2003-03-26 17:13:49.056003 | Repository: JWANKOWSKI-DB2-SQLREP30 (SQLREP30) | 10 Statements retrieved

Extended Object Analysis

Analyze SQL Collected from JWANKOWSKI-DB2-BASEBALL

Top Statements | Top Applications | Top Users | Top Transactions | Top Connections | Deadlocks | **Extended Analysis**

Objects | Access Plan

Creator	Name	Type	SQL activity	Select activity	Insert activity	Update activity	Dele
DB2ADMIN	EXPLAIN_OBJECT	Table	35	35	0	0	
DB2ADMIN	EXPLAIN_INSTANCE	Table	35	0	0	0	
DB2ADMIN	EXPLAIN_ARGUMENT	Table	35	35	0	0	
DB2ADMIN	BATTING	Table	323	323	0	0	
DB2ADMIN	ADVISE_INDEX	Table	35	0	0	0	

Object Criteria
Schema: *
Refresh

Statements | Columns | Indexes

Creator	Name	Unique rule	In use	Column	In use	Seqno	Order	Basic pred.
DB2ADMIN						1	Asc	
DB2ADMIN								

Extended analysis provides optional historical analysis of objects accessed by collected SQL

ID	SQL Statement	Executions	Total Elapsed time	Total CPU user

SQL Detail...
Tune SQL...
Save All SQL...

Collections: db2admin.Demo Run

Job ID: 1 | End time: 2003-03-26 17:13:49.056003 | Repository: JWANKOWSKI-DB2-SQLREP30 (SQLREP30) | 20 Objects / 2 Indexes / 0 Statements

Deadlock Analysis

Deadlock Report

Requested Lock
Held Lock
Rejected Application ID

Rejected Application ID	Agent ID	Start time	Connections
*LOCAL.DB2.030211153432	63	2003-02-11 09:36:25.163003	4

Rejected Application Deadlock Detail

Application ID	*LOCAL.DB2.030211153432
Agent ID	63
Lock wait start time	2003-02-11 09:36:25.163003
Application ID (Holding lock)	*LOCAL.DB2.030211153439
Lock object type	Tablespace
Tablespace name	USERSPACE1
Transaction ID	113
Status	Rollback due to deadlock
CPU user	00:00:00.070100
CPU system	00:00:00.040058

Save Print Cancel

Job ID: 4 End time: 2003-02-11 09:39:12.011551 Repository: SOLSONXP-DB2-VERSION3 (VERSION3) 1 Deadlocks / 2 Statements retrieved

**Deadlock Analysis
Identifies any deadlocks which occurred
During collection**

Summary

- Build a strategy for how you will monitor/tune your DB2 environment
 - Try to identify most critical applications and begin focusing on those
 - Set realistic performance expectations
 - Tune, monitor, tune, monitor...



THANK YOU
FOR LISTENING!

*For More Information on Quest's
DB2 Solutions and to download free
trials:*

WWW.Quest.Com/DB2