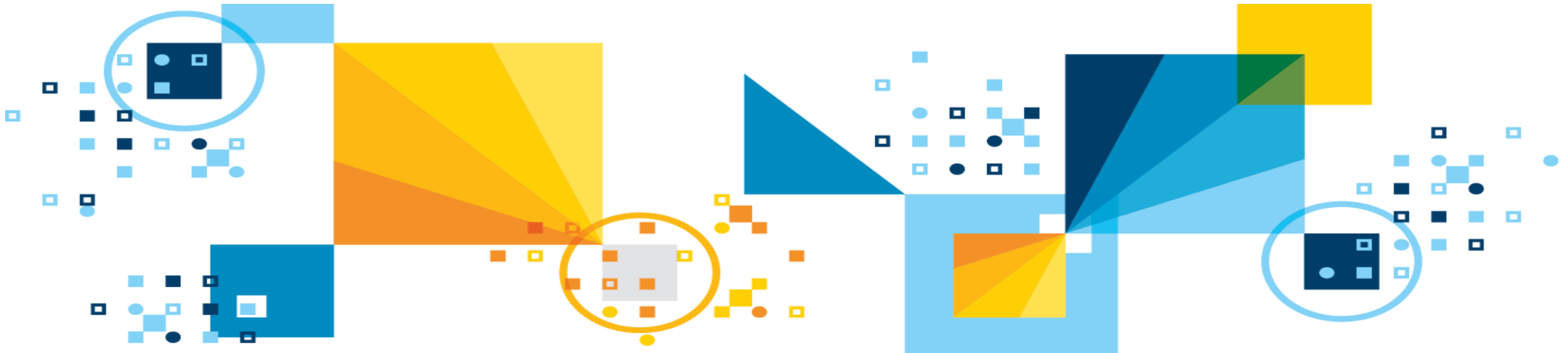


# IBM Data Server Manager for z/OS

John Casey  
Senior Solutions Advisor  
[jcasey@rocketsoftware.com](mailto:jcasey@rocketsoftware.com)





IBM Data Server Manager

Home / Welcome

### Welcome to Data Server Manager

**Set Up and Configure**

- Manage users and privileges
- Set up a repository database
- Add database connections

**Monitor Databases**

- Add database connections
- Edit monitoring profiles
- View reporting data

**Administer and Configure Databases**

- Add database connections
- Manage databases
- Create and manage database objects
- Track changes

**Diagnose and Optimize SQL Statements**

- Add database connections
- Filter and run SQL statements
- Tune SQL statements
- Discover related connections

## IBM Data Server Manager

User name

Password

Log In



**IBM Knowledge Center**  
Find answers quickly with IBM product documentation



**Redbooks**  
Complimentary, step-by-step guides for download and mobile access



**Community & Forum**  
Explore technical topics, find trial software, and join the community



**Video**  
Start your learning journey with best practices and new product features



**Support portal**  
Get help today for the IBM services and software you own



**Twitter**  
Contact us by Twitter

# Performance tuning challenges across the organization



Application Developer

*"I don't have time to hone my SQL skills. I need to focus on developing core application functionality."*

*"It is very challenging to aggregate performance data across our complex data environment."*



DBA

*"I don't understand why our developers aren't focused on creating better performing SQL."*



LOB Manager

*"I need to get my business results fast and accurate. What's going on?"*



IT Manager

*"Performance problems seem to appear without warning and deep technical skills are hard to find."*



QA Manager

*"We can't adequately test for peak workload since we don't have enough human or IT resources."*

## Reactive vs. Proactive performance management

- Problems addressed after performance impact
- Measuring flashing light indicators
- Noticing either the very good or the very bad
- Takes longer to react to bad performance because of measuring lagging indicators



- Understanding what behavior is desired
- Measuring leading indicators
- Capture best practices and procedures
- Team responsible for creating the measurements understands the what and why

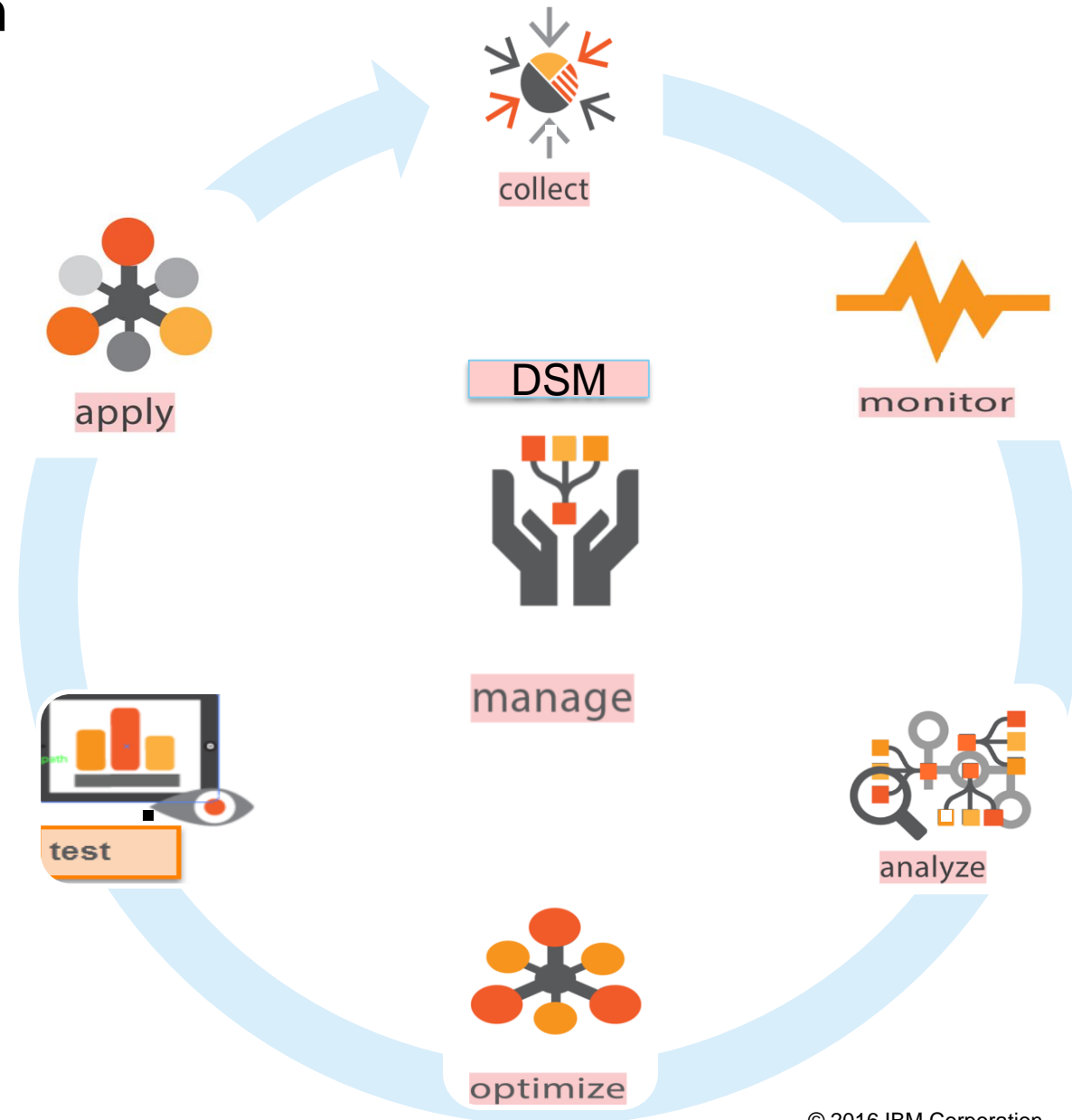
## What should you look for in a performance management solution?

- Cost reductions of Db2 and associated applications
- Faster identification and resolution
- Improved overall performance
- Replacing ad-hoc methods
- Faster Db2 and application migration



# IBM Db2 Performance Management Solution provides:

- ✓ Fast identification with automated alerts, proactive notification and 24x7 monitoring
- ✓ Tuning of queries and workloads proactively
- ✓ Expert advice with built-in advisors
- ✓ Diverse set of capabilities managed via Data Server Manager (DSM)
  - ✓ Easy-to-use integrated view of overall Db2 performance management
  - ✓ Seamless navigation and movement via functional capabilities versus individual products



## Welcome to Data Server Manager

### Set Up and Configure



- Manage DSM users and privileges
- Set up a repository database
- Add database connections

### Monitor Databases



- Add database connections
- Edit monitoring profiles
- View monitoring data

### Administer and Configure Databases



- Add database connections
- Manage databases
- Create and manage database objects
- Track changes

### Execute and Optimize SQL Statements



- Add database connections
- Edit and run SQL statements
- Tune SQL statements
- Discover object dependencies

## Data Server Manger – Where DBAs Spend Time



### Administration

- helps you manage, and maintain complex database environments for increased productivity and optimized use of system resources



### Performance Tuning

- helps you develop and implement a performance strategy including providing expert recommendations to improve query workload performance



### Identifying Environment changes

- offers centralized management of database and client configuration



### Troubleshooting

- capture production application workloads then compare capture and enforce configuration settings



# Features of the Data Server Manager z/OS Based Tools - At A Glance

**NO CHARGE**

## Data Server Manager Base

- Connect to Db2 for z/OS V10/ V11/V12
- Database object navigation, viewing object detail, and linking to related objects.
- Database object dependency display.
- Data browsing and editing.
- Basic database object operations, such as creation of tables, indexes, constraints, and tablespaces; dropping of tables, indexes and constraints; altering tables.
- Show system privilege from the perspective of Group/User, Role, or SQL object. Choose:
  - "Group/User" to see the role and the relative object privilege for a user account;
  - "Role" to see the role a user account belongs to and its relative object privilege;
  - "SQL object" to see a specific object and users or roles that have the relative authority.
- Single query tuning
  - Statistics Advisor
  - Query Environment Capture
  - Access Path Graph

## Db2 Performance Solution Pack

### IBM Query Workload Tuner

- Launch of visual explain and tune query on the SQL editor
- Tuning wizard to capture SQL statements from multiple sources
- Tuning advisors provide recommendations for:
  - Statistics Advisor
  - Index Advisor
  - IDAA Advisor
- Problem analysis of query or workload
  - Access plan graph
  - Query formatting and annotation
  - Tuning Report
  - Test Candidate Index
  - Access Plan Comparison
  - Index Impact Analysis
  - Query and Workload Environment Capture
  - Selectivity Override

### IBM Db2 Query Monitor

- Launching of DSM from Query Monitor Web UI for end to end performance analysis
- Host variable collection

### OMEGAMON XE for Db2 PE

- Key Performance Indicators (KPIs) displayed in Data Server Manager on the Subsystems

## Db2 Admin Solution Pack

Configuration Manager for z/OS V5.1

- Track configuration changes
- Configure zParm
- Compare and clone configurations
- Manage application profile
- Manage alias
- Manage and control clients

## Db2 Utility Solution Pack V4.1

- Customizable profiles for performing conditional object evaluations and generating actions mapped to resolving utilities (reorg, copy, runstats, etc)
- Ability to control prioritization of objects, evaluation conditions and generated resolving actions.
- Ability to define maintenance windows for enabling autonomics, allowing Db2 to self manage utility runs
- Graphical trend analysis of historical RTS
- Capture of utility history, recording utility output, time, duration, etc.

# Where do I start ?..... Data Server Manager



<http://ibm.biz/IWANTDSM>

# Simple 3-step setup

Step 1 of 3

Accept the terms of the license agreement to continue

International License Agreement for Ea

Part 1 - General Terms

BY DOWNLOADING, INSTALLING, COP  
TO THE TERMS OF THIS AGREEMEN  
HAVE FULL AUTHORITY TO BIND LIC

\* DO NOT DOWNLOAD, INSTALL, COP

\* PROMPTLY RETURN THE UNUSED I  
PROGRAM.

1. Definitions

"Authorized Use" - the specified level of  
service units ("MSUs"), Processor Value

"Early Release" - a release of a Program  
potentially unreliable.

"IBM" - International Business Machine

I accept the terms of the

Step 2 of 3

Specify credentials and port numbers

Specify a us

User name:

Password:

Re-enter passwor

Change port

HTTP port for acct

HTTPS port for aci

Port used internall

Step 3 of 3

Setup is complete

The server is registered as a Windows service

You can open the IBM Data Server Manager web console by using the following URLs:

<http://localhost:11080>

<https://localhost:11081> (secure)

Record the URLs of the web console so that you can use them to log in later.

<http://ibm.biz/IWANTDSM>



# Migrating workloads from QWT 4.1.x to DSM

## Database connections

Database connections			
Query Tuning   Blackout   Validate Credentials			
Name	Database name	Host name	Port num
CSDEC720		STLEC1	9.30.222.46   446
LABEC504V11NFM		STLEC1	9.30.112.62   446
LABEC508 V10NFM	DB2 for z/OS (V10.1.5)	STLEC1	9.30.113.49   446
LABEC508_ADMF002	DB2 for z/OS (V10.1.5)	STLEC1	9.30.113.49   446
LABEC508sa	DB2 for z/OS (V11.1.5)	STLEC1	9.30.112.62   446
REGEC580	DB2 for z/C		

- Start Tuning
- Configure for Tuning
- Generate Configuration Report
- Manage Query Tuning Results and Jobs
- Migrate from Optim Query Workload Tuner 4.1.x**

### Migrate from Optim Query Workload Tuner 4.1.x

Specify file exported from Optim Query Workload Tuner 4.1.x

The maximum size of upload file is 100M

Choose a Workload File

File Name

Size

No items to display

# Manage Databases Using the Database Explorer

The screenshot shows the IBM Data Server Manager interface. The top navigation bar includes 'Home', 'Administer', 'Run SQL', 'Monitor', 'Jobs', 'Settings', and 'Help'. The 'Administer' menu is expanded, showing options like Tables, Views, Indexes, Constraints, MQTs, Aliases, Synonyms, Auxiliary Tables, Schemas, Storage Objects, Application Objects, and Privileges. The main area displays a table of database objects with columns for Schema, Table space, Cardinality, Number of columns, Encoding, Created time, and Statistic time. A search filter is visible at the top right of the table area.

Schema	Table space	Cardinality	Number of columns	Encoding	Created time	Statistic time
DDS1401	G07MDERI	100	9	EBCDIC	2010-10-26 01:19	2013-01-21
DNET064	JIDAA	-1	3	EBCDIC	2015-01-28 18:15	0001-01-01
DNET783	DDS0904S	-1	2	EBCDIC	2010-11-02 11:44	0001-01-01
MIGUSER	ETWRR143	-1	1	EBCDIC	2006-08-24 08:53	0001-01-01
DDS0376	BCZUT175	0	2	EBCDIC	2009-05-25 19:59	2013-01-21
DNET377	TAAA0002	-1	9	EBCDIC	2006-11-27 09:44	0001-01-01
DBA031	AADELTAB	3	2	EBCDIC	2009-07-02 16:15	2013-01-21
JSIZTO	ABC	9	1	EBCDIC	2016-12-27 17:30	2016-12-27
ABP22PLN	ABPAUDTS	-1	10	EBCDIC	2016-10-25 13:20	0001-01-01
ABP22PLN	ABPPREIX	-1	5	EBCDIC	2016-10-25 13:20	0001-01-01
ABP22BNY	ABPPREIX	-1	5	EBCDIC	2014-02-05 16:05	0001-01-01
ABP22BNY	ABPPREDI	-1	4	EBCDIC	2014-02-05 16:05	0001-01-01
ABP22PLN	ABPPREDI	-1	4	EBCDIC	2016-10-25 13:20	0001-01-01
ABP22BNY	ABPPREIP	-1	4	EBCDIC	2014-02-05 16:05	0001-01-01

Callouts in the image:

- Manage database objects**: Points to the top right of the interface.
- Explore database object properties**: Points to the left-hand navigation menu.
- Explore the catalog**: Points to the bottom right of the interface.

# Develop and Run SQL Scripts

The screenshot displays the IBM Analytics SQL Editor interface. At the top, there are tabs for 'Database Explorer', 'SQL Editor', and 'Browse/Edit Data'. Below the tabs is a toolbar with buttons for 'Run', 'Format', 'Syntax Assist', 'Save', 'Explain', 'Tune', and 'Learn more'. The main area contains a SQL script editor with two lines of code: `1 SELECT * FROM DSN8B10. ACT;` and `2 SELECT * FROM DSN8B10. EMP;`. A 'Validate SQL' callout box is positioned over the script. To the right, there are configuration fields for 'Statement terminator', 'Current SQL ID', 'Default schema', 'Run method' (set to 'JDBC'), and 'Maximum row returned' (set to '100'). A 'Options' button is also visible. Below the editor, a 'Saved Queries' section shows a table of execution results. The table has columns for 'Status', 'Run time (second)', 'Statement', 'Date', and 'Report'. Two rows are shown, both with a 'Succeeded' status. The first row has a run time of 1.444 and a date of 12/11/2015, 5:31:08. The second row has a run time of 0.148 and a date of 12/11/2015, 5:31:09 PM. Below the table, there is a 'Log' button and a toolbar with icons for filtering, grid view, and other actions. A callout box is positioned over the table with the text 'Customize and filter result output' and 'Save execution results'.

- Manage scripts
- Explain SQL
- Tune SQL

Validate SQL

Status	Run time (second)	Statement	Date	Report
✓ Succeeded - EC_V11A	1.444		12/11/2015, 5:31:08	Download history
✓ Succeeded	0.148	SELECT * FROM DSN8B10.ACT	12/11/2015, 5:31:09 PM	Download history   data

- Customize and filter result output
- Save execution results

# Create and schedule jobs

The screenshot illustrates the process of scheduling a job in IBM Analytics. It shows three main panels:

- Top Panel:** A 'Create Table' job is selected in the 'Database Explorer'. A callout bubble points to the 'Schedule as a job' option in the 'Run schedule' section.
- Middle Panel:** The 'Scheduled Commands' view is shown. A callout bubble points to the 'Alter/Create Table' dropdown menu. Below this is a table of existing jobs:

Job ID	Job name	Interval	Start time
1449892843365	Create Table	Once	2015-12-11, 23:15

- Bottom Panel:** The 'Add Job' dialog is open. A callout bubble points to the 'Type' dropdown menu, which has 'SQL Only Script' selected and highlighted with a red box.

Callout bubbles provide additional context:

- 'Schedule a Create Table job from Explore Databases' (points to the top panel)
- 'Scheduled commands from Explore Databases' (points to the middle panel)
- 'Alternatively, create a script and schedule the job' (points to the bottom panel)

<http://ibm.biz/IWANTDSM>



# Db2 Utilities Solution Pack 2.2

The screenshot displays the IBM Data Server Manager interface for a Db2 subsystem named TOLEC107 (DB2A). The interface is divided into several sections:

- Properties:** Shows subsystem details such as Subsystem name (TOLEC107 (DB2A)), Level (111), System name (TOLEC107), IRLM subsystem (PR21), IRLM procedure (PRLMPR21), Status (ACTIVE), Catalog level (111), Mode (NFM), and Thread count (5).
- Symptoms:** A list of critical symptoms, including "Leaf page distance < 9000" for various tablespaces (DBAUB701 > IAUB7021 > 5, etc.).
- Actions:** A list of scheduled actions, primarily "DB2 REORG INDEX UTILITY" for various tablespaces.
- Utilities:** A table showing the execution history of various utilities, including DB2AUUTL1 and REORG\_INDEX.
- Details Panel:** A pop-up window showing details for utility DB2AUUTL1, including its ID, step name, authorization, start date, and elapsed time.

Symptoms/actions on subsystem and object dashboards

“More integration, greater value”

Optimize, control manage & automate

- Components:**
- Db2 Automation Tool
  - Db2 High Performance Unload for z/OS
  - Db2 Sort for z/OS
  - Db2 Utilities Enhancement Tool
  - Autonomics support
  - Data Server Manager

Db2 Utilities Solution

Automate Data Collection Utility History

View upcoming autonomic maintenance windows with scheduled actions

## Traditional Reactive Tuning



## What to do next .... Performance Tuning Using Data Server Manager : Query Workload Tuner 5.1



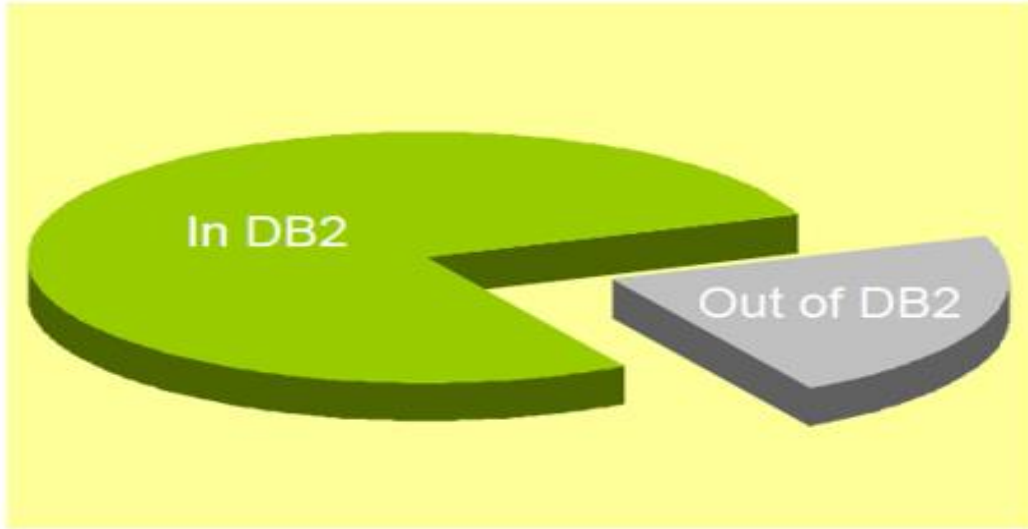
<http://ibm.biz/IWANTDSM>

## Why is workload tuning important?

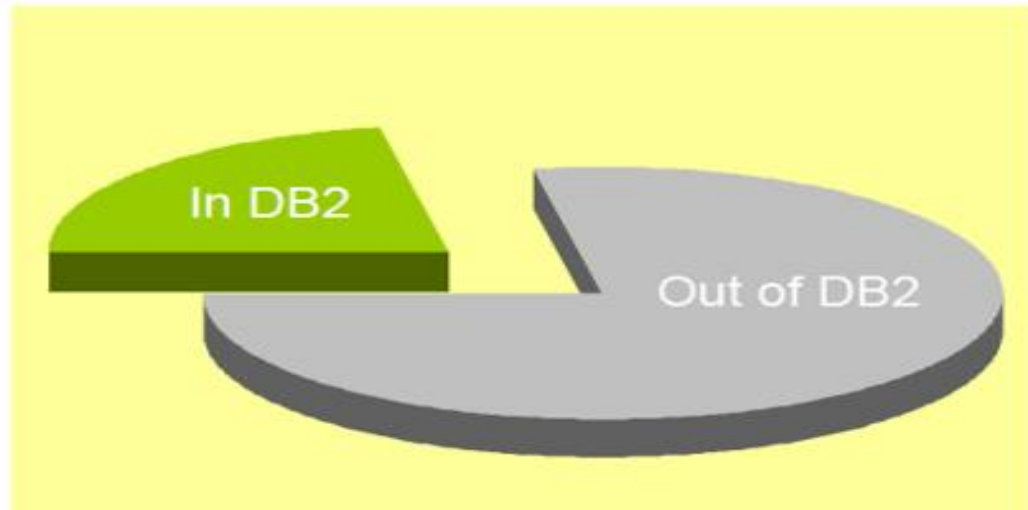
### *Workload: Multiple SQL statements defined by user*

- The effort for tuning the whole application with good performance by evaluating every statement is overwhelming. Optimization decisions are based on trade offs:
  - Statistics – CPU costs vs. query savings
  - Indexing – query speed vs resource and transaction
- Sometimes performance improvement for one statement in an application may regress other statements in the application.
- When your application data grows, allows you to do proactive application health check periodically to find potential problems earlier before costly application outages
- Workload tuning speeds up analysis
  - Analyzes multiple queries at once
- Workload tuning consolidates and optimizes recommendation for overall workload
  - Statistics recommendations
  - Index recommendation

# Where is the most time spent?



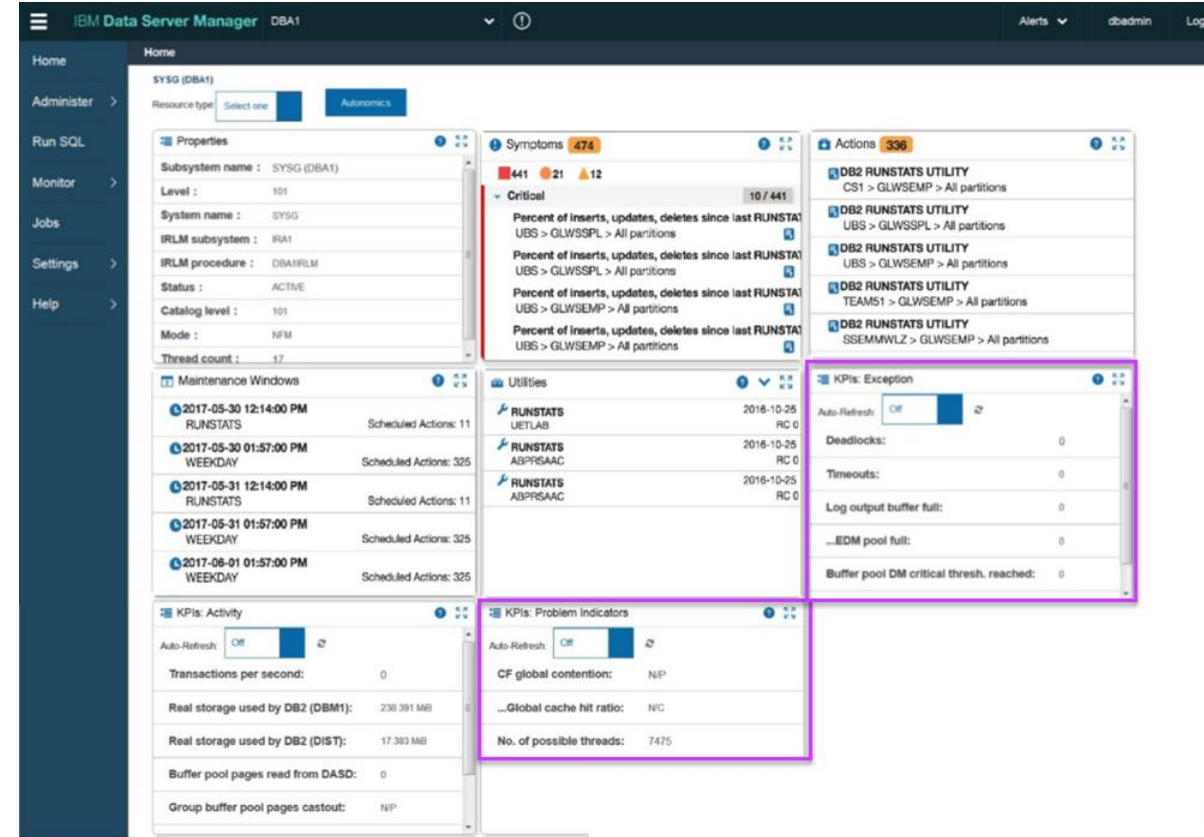
Need to analyze time distribution:  
where is the time really spent?



- Application logic inefficiency
  - often combined with  
Class 2 CPU << Class 1 CPU
- Network problems
- Class 2 not active all the time

# Improve Performance and Reduce Costs

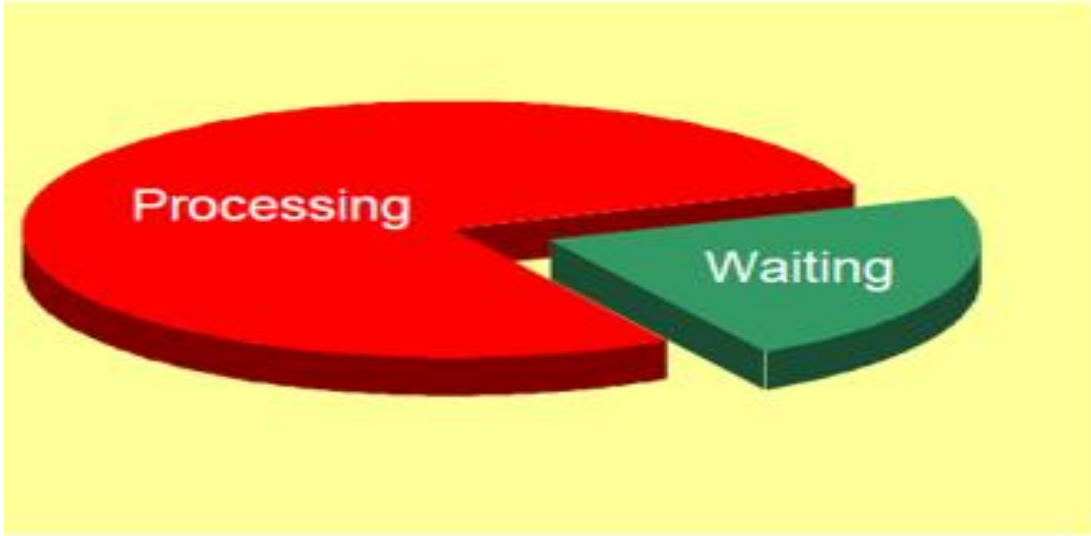
- Improve end-user experience of performance
  - Monitor KPIs that better reflect end-user experience
    - i.e., transaction response time
  - Get early warning of degrading performance before users are affected
  - Isolate problems to correct area for fast response
  - Get expert advice for improving query and workload performance
- Reduce costs
  - Improve performance and govern system utilization to defer upgrades
  - Save hours of staff time and stress
    - Isolate problems to the right layer of the application stack, database component, even the line of code
  - Enable developers and novice DBAs to tune like an expert



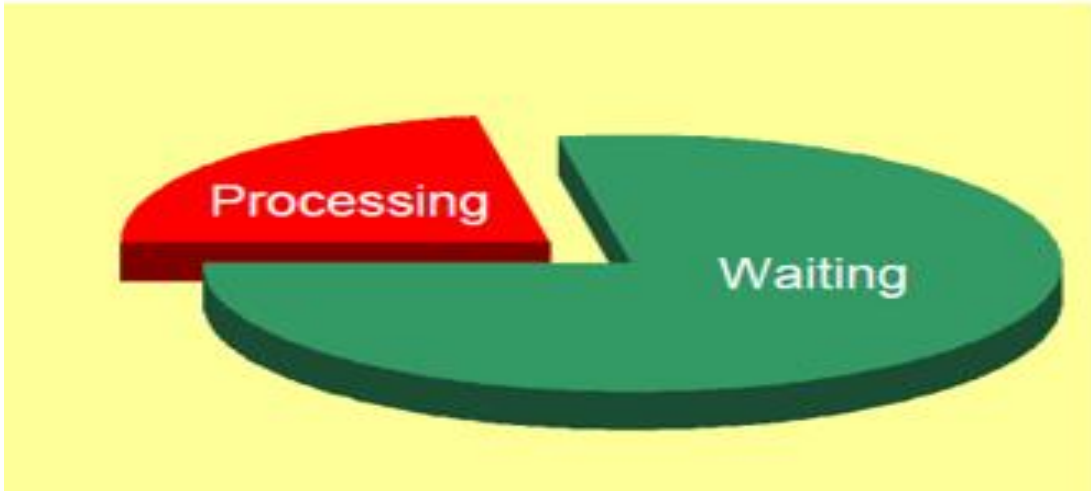
Accelerate analysis and reduce downtime for urgent situations



# Where is the time spent within Db2?



- Turn off expensive traces
- Inefficient access paths
  - Explain



- What is the largest contributor
  - Class 3 and 8 analysis

What is your query tuning objective .....  
reduced CPU usage or reduced  
elapsed time ?





## Common Scenarios & Collection

- Most expensive SQL statement in your Db2 subsystem
- Most expensive SQL within a PLAN
- All of the PLANS where a specific package is used
- All of the “exceptional SQL for a given plan”
- All of the objects accessed by a specific package
- All of the SQL which access a specific object
- Unnecessary negative SQLCODES





# Determine what data needs to be collected

- Data available to be collected or viewed
  - SQL metrics
  - Db2 object access
  - SQL text and host variables
  - Db2 commands
  - Negative SQLCODES
  - Expanded and grouped Information about exceptions
  - Buffer Pool Statistics
  - Delays
- Three types of data
  - **Summary** – data summarized for each unique SQL statement executed in a particular interval of time
    - Plan + Program + Section + Statement # + Statement type
    - SQL Codes are not collected by default
  - **Exceptions** – individual SQL calls that have exceeded user defined thresholds
  - **Alerts** – events that require immediate attention; can be classified as exceptions



## 2 Identify workload for proactive tuning

- Identify the topN (50 , 200 , 250 ..... ) most expensive queries
- Drill down into results
- Save workload and/or start tuning

The screenshot displays the IBM Analytics interface for query performance analysis. The main table shows the following data:

Command	Exec Count	CPU	zIIP CPU	Delay	Getpages	SQL Text	DB2
	120	1.212301	0.000000	3.511193	209,585	SELECT EMP_...	
	192	15.493578	0.000000	0.004839	1,790,887	DELETE FROM...	DB2
	570	1.368879	0.000000	0.000015	0	CQM145I - SQL...	DB2
	570	3.443662	0.000000	0.456599	108,655	DECLARE C1 C...	DB2
	193	1.793139	0.000000	24.144000	494,184	DELETE FROM...	DB2
	990	0.455963	0.000000	4.019060	160,020	UPDATE GLWT...	DB2
	976	0.004593	0.000000	0.000000	0	DECLARE C1 C...	DB2
	7,924	0.686005	0.000000	0.000004	71,233	UPDATE GLWT...	DB2
	0	3.569418	0.000000	41.238604	1,691,484	DECLARE C1 C...	DB2
	976	1.608397	0.000000	0.019863	315,366	SELECT MIN(D...	DB2
	976	1.815363	0.000000	4.998261	315,366	SELECT MIN(D...	DB2
	359	1.243767	0.000000	9.714349	625,217	DECLARE C1 C...	DB2

The right sidebar shows the following analysis details:

- Elapsed Time**: Value 4.939776
- Total Delay Time**: Value 3.511193, Overall 71.08%
- Total IO Delay**: Value 3.511042, Overall 71.08%, Percent 100.00%
- Other Read Delay Time**: Value 2.895085, Overall 58.61%, Percent 82.46%
- Synchronous I/O Delay Time**: Value 0.615957, Overall 12.47%, Percent 17.54%
- Other Write Delay Time**: Value 0.000000, Overall 0.00%, Percent 0.00%
- Archive Log Quiesce Delay Time**: Value 0.000000, Overall 0.00%, Percent 0.00%



2

# Drill down into results

Tune or Tune all

The screenshot shows the IBM Analytics Activity Browser interface. The main table displays execution details for various SQL commands. A callout box highlights a specific row with a CPU time of 25:52.661214 and a delay of 22:37.039138. A blue callout bubble points to the 'Tune All' button in the top toolbar. On the right, the 'Analysis' pane is open, showing 'Elapsed Time' (52:02.821881) and a detailed 'Delays' table.

Command	Exec Count	CPU	zIIP CPU	Delay	Getpages	SQL Text	DB2
	90,504	25:52.661214	25 minutes 52.661214 seconds	22:37.039138	1,511,755,479	SELECT MIN(E...	DBE
	192	15.493578	0.000000	0.004839	4,790,887	DELETE FROM...	DBE
	1,911	12.570105	0.000000	1:57.671076	6,636,974	SELECT MIN(E...	DBE
	1,911	8.547753	0.000000	6.503590	690,358	INSERT INTO G...	DBE
	1,911	7.310767	0.000000	0.002798	567,567	DECLARE GLO...	DBE
	89,541	4.705117	0.000000	0.011513	55,419	INSERT INTO S...	DBE
	95,929	3.727368	0.000000	0.113124	0	CQM145I - SQL...	DBE
	0	3.569418	0.000000	41.238604	1,691,484	DECLARE C1 C...	DBE
	89,541	3.543430	0.000000	0.108491	0	CALL EMPSEL ...	DBE
	570	3.443662	0.000000	0.456599	108,655	DECLARE C1 C...	DBE
	23,131	2.800626	0.000000	0.118497	0	CALL : H USIN...	DBE
	1,911	2.477874	0.000000	0.109868	0	CQM145I - SQL...	DBE

Delay Event	Event Count	Delay Time
Lock Latch Delay Time	27,526	0.113124
Synchronous I/O Delay Time	3,722,940	19:29.466114
Database I/O Delay Time	3,722,940	19:29.466114
Log Write I/O Delay Time	0	0.000000
Other Read Delay Time	353,348	3:07.459899
Other Write Delay Time	0	0.000000
Servtask Switch Delay Time	0	0.000000
Update Commit Delay Time	0	0.000000



analyze

# Capture from Statement Cache & customize collection

- Accelerate analysis, reduce downtime

The screenshot shows the 'Optimize' section of the IBM Data Server Manager. The 'Statement cache' tab is selected. Under 'More options', the 'Default SQLID' is set to 'SYSADM'. The 'Using the SYSPROC.OPT\_RUNSQL' toggle is turned off. Both 'Exclude system statements' and 'Capture from data sharing' are checked. Below this, there is a table of available database connections for data sharing.

NAME	LOCATION	HOST NAME
<input type="checkbox"/>	LABEC504adm002	STLEC1 9.30.112.62
<input type="checkbox"/>	LOCDB21	LOCDB21 9.125.72.130

Capture & Select from Data Sharing members

The 'Manage filters' dialog box is shown, listing various filters with checkboxes for selection. The columns include NAME, SHARED, OWNER, and DATE MODIFIED. A pink arrow points from a callout box to the 'SHARED' column.

NAME	SHARED	OWNER	DATE MODIFIED
<input type="checkbox"/> STABILIZED=Y	<input type="checkbox"/>	admin	2017-06-11 19:20:39
<input type="checkbox"/> EP_RATIO_DESC	<input checked="" type="checkbox"/>	SYSTEM	2017-05-23 19:43:26
<input type="checkbox"/> PCT_CPU_DESC	<input checked="" type="checkbox"/>	SYSTEM	2017-05-23 19:43:15
<input type="checkbox"/> PCT_ELAP_DESC	<input checked="" type="checkbox"/>	SYSTEM	2017-05-23 19:43:08
<input type="checkbox"/> JASON_GETPAGES	<input checked="" type="checkbox"/>	admin	2017-05-16 11:00:02
<input type="checkbox"/> aa-copied2	<input type="checkbox"/>	admin	2017-04-27 16:04:19
<input type="checkbox"/> aa-copied1	<input type="checkbox"/>	admin	
<input type="checkbox"/> cc	<input type="checkbox"/>	admin	
<input type="checkbox"/> aa	<input type="checkbox"/>	admin	
<input type="checkbox"/> samename	<input type="checkbox"/>	hjq	
<input type="checkbox"/> samename-copied2	<input type="checkbox"/>	zhw	
<input type="checkbox"/> samename-copied1	<input checked="" type="checkbox"/>	zhw	
<input type="checkbox"/> defaultFilter468-copied1	<input checked="" type="checkbox"/>	zhw	
<input type="checkbox"/> defaultFilter468	<input type="checkbox"/>	admin	
<input type="checkbox"/> ACCUM_CPU_DESC-copied2	<input checked="" type="checkbox"/>	zhw	
<input type="checkbox"/> samename	<input type="checkbox"/>	admin	
<input type="checkbox"/> ACCUM_CPU_DESC-copied1	<input checked="" type="checkbox"/>	zhw	
<input type="checkbox"/> defaultFilter70	<input checked="" type="checkbox"/>	zhw	
<input type="checkbox"/> ACCUM_CPU_DESC	<input checked="" type="checkbox"/>	SYSTEM	
<input type="checkbox"/> ACCUM_ELAP_DESC	<input checked="" type="checkbox"/>	SYSTEM	

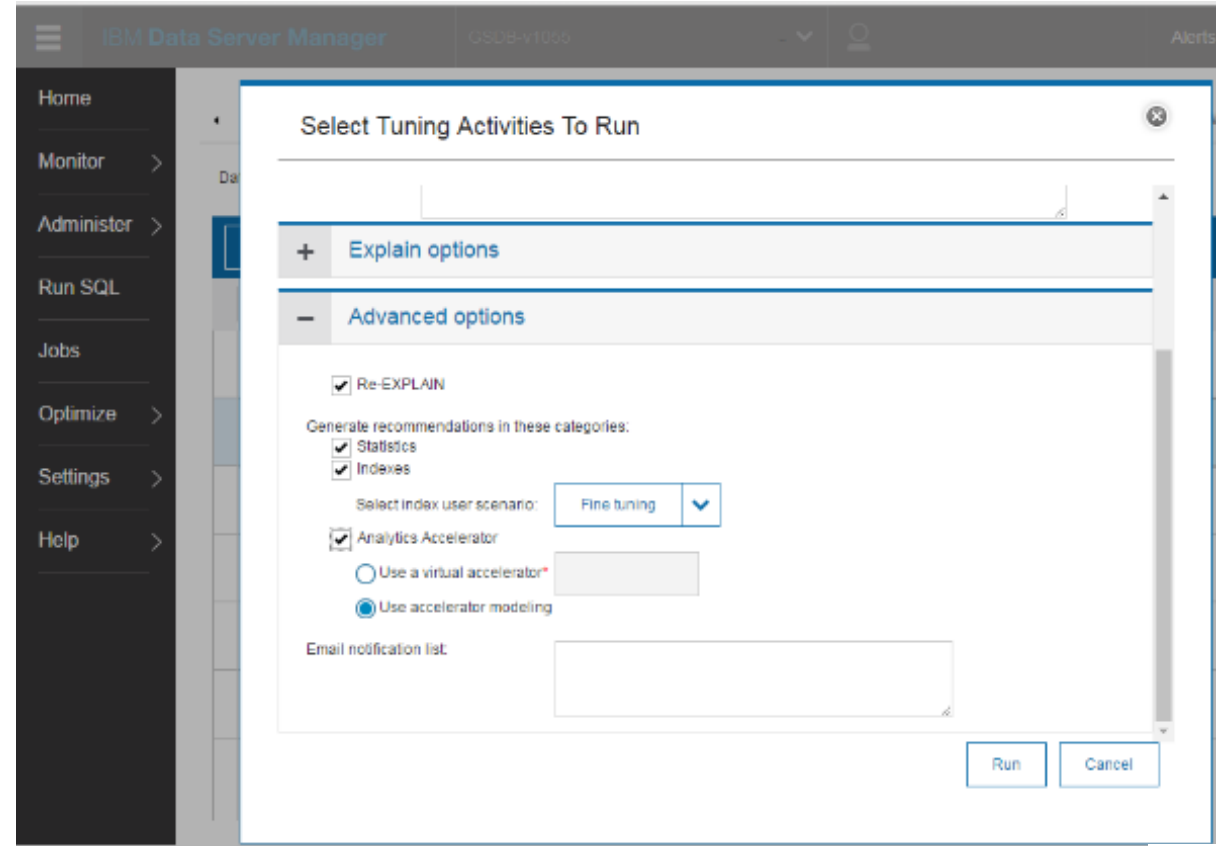
Share & Save Filters to speed up analysis

This screenshot shows the configuration options for a filter. The filter name is 'defaultFilter897'. The 'Shared' checkbox is checked. Below, there are sections for selecting filters to capture statements from the dynamic statement cache, filters to capture statements if statement cache trace is enabled, and columns to sort the captured result. The 'Maximum statements' field is set to 100.

Create new filter either shared or private

## 3 Execute Advisors

- **Statistics**
  - Get recommendations on the best statistics to capture to influence access path selection
- **Index**
  - Get recommendations on indexes changes that can reduce database scans
- **Analytics Accelerator**
  - Get recommendations on optimizing and managing accelerated analytic queries and applications



# Improve statistics quality and collection

Query and Workload Job **Tuning Results(tpch60GB)** x Tuning Results(Query\_1449642937354-Result\_1449642937354) x

Recommendations

View Workload Statements

**RUNSTATS Script**

Index Script

Analytic Acceleration Script

Tuning Options Used

Tuning Job Log

Run Recommended RUNSTATS Open in New Window View Detail report and Conflict detail

```
--
-- RUNSTATS command to repair problem statistics
--
RUNSTATS TABLESPACE "TPCH060DB"."TSCUS"
TABLE("TPCH60"."CUSTOMER")
COLUMN("C_ACCTBAL")
COLGROUP("C_ACCTBAL") HISTOGRAM NUMQUANTILES 20
COLGROUP("C_MKTSEGMENT") FREQVAL COUNT 10
SORTDEVT SYSDA SORTNUM 4
INDEX("TPCH60"."IXCUS01")
SHRLEVEL CHANGE REPORT YES UPDATE ALL HISTORY NONE
;
RUNSTATS TABLESPACE "TPCH060DB"."TSLI"
TABLE("TPCH60"."LINEITEM")
COLUMN("L_ORDERKEY")
COLGROUP("L_QUANTITY") FREQVAL COUNT 10 HISTOGRAM
NUMQUANTILES 20
COLGROUP("L_ORDERKEY","L_SUPPKEY")
COLGROUP("L_RETURNFLAG") FREQVAL COUNT 10
COLGROUP("L_PARTKEY","L_SUPPKEY")
COLGROUP("L_SHIPINSTRUCT") FREQVAL COUNT 10
SORTDEVT SYSDA SORTNUM 4
INDEX("TPCH60"."IXLI01",
"TPCH60"."LINEITEM_VIRT_IDX_144910527477340314")
SHRLEVEL CHANGE REPORT YES UPDATE ALL HISTORY NONE
;
RUNSTATS TABLESPACE "TPCH060DB"."TSNTN"
TABLE("TPCH60"."NATION")
COLUMN("N_NAME")
COLGROUP("N_NAME") FREQVAL COUNT 10
SORTDEVT SYSDA SORTNUM 4
INDEX("TPCH60"."IXNS01")
SHRLEVEL CHANGE REPORT YES UPDATE ALL HISTORY NONE
;
```

### Statistics Advisor Detail Report

Statistics Advisor Detail Report  
 Analysis start time: 2015-12-02 18:31:48.504682  
 Analysis end time: 2015-12-02 18:31:50.455052

---

TABLE TPCH60.CUSTOMER  
 Table type: Table  
 Cardinality: 60000.0  
 Collection time: 2015-11-17 01:30:26.468977  
 Statistics status: OK

INDEXES:

TPCH60.IXCUS01 (C\_CUSTKEY)  
 First key cardinality: 60000.0  
 Full key cardinality: 60000.0  
 Data repetition factor: 1170.0  
 Collection time: 2015-11-17 01:30:26.468977  
 Statistics status: conflicting

Interesting columns:

C\_MKTSEGMENT  
 Cardinality: 5.0  
 Uniform statistics collection time: 2015-11-17 01:30:26.468977  
 Uniform statistics status: OK  
 Frequency statistics collection time: null  
 Frequency statistics status: missing  
 Histogram statistics collection time: null  
 Histogram statistics status: missing  
 Possibly point skewed: Yes  
 Symptom: Columns with low COLCARD (the number of distinct values in a column)  
 Possibly range skewed: No

Conflicting statistics explanation

- Results
  - Accurate estimated costs
  - Better query performance
  - Less CPU consumption
  - Improved maintenance window throughput



optimize

*“80 % of access path PMRs could be resolved by statistics advisor before calling IBM support.” – IBM Support*

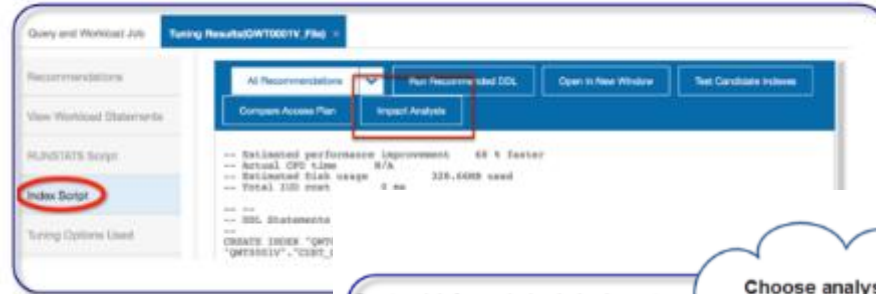
# Indexing advice to improve database design

## Workload Index Impact Analysis



analyze

- Indexes are decided at design stage
  - Lot of effort is spent making SQL to use the provided indexes
  - But what if the SQL is "right" and it's the indexes that are "wrong"
  - Cost resources to maintain
  - *How do you simply test your hypotheses without impacting production?*



- Removing obsolete indexes simplify use
  - Consolidate indexes and provide a single recommendation
  - Enables what-if analysis
  - Provides DDL to create indexes
  - Run immediately or save

% Performance gain on Statement/ Package level

Index	Table	Key Columns	Estimated Performance Gain(%)	Statement Impact	Action
Original Index Recommendations					
CUST_CREDIT_CARD_VIRT_IDX_145215367825162295	CUST_CREDIT_CARD	CUST_CODE(ASC), CREDIT_METHOD_CODE(ASC)	99.81	4	<a href="#">View Statements</a>
CUST_ORDER_DETAIL_VIRT_IDX_145215367820745351	CUST_ORDER_DETAIL	CUST_ORDER_NUMBER(ASC)	99.85	6	<a href="#">View Statements</a>
CUST_ORDER_HEADER_VIRT_IDX_145215367822725941	CUST_ORDER_HEADER	CUST_CODE(ASC), CREDIT_METHOD_CODE(ASC)	99.83	1	<a href="#">View Statements</a>
CUST_ORDER_HEADER_VIRT_IDX_145215367821540997	CUST_ORDER_HEADER	CUST_CODE(ASC), CREDIT_METHOD_CODE(ASC)	99.99	2	<a href="#">View Statements</a>



test

- Test before deployment
  - Use virtual index capabilities built into the Db2 engine



# Optimizing the selection and tuning of accelerated workloads



## ■ Workload Analytics Accelerator Advisor

- Identify candidate queries and tables to be routed to the Accelerator
- Identify candidate tables to be routed to the accelerator
- Implement advisor-based tuning recommendations for mixed workloads of accelerated and un-accelerated queries
- Diagram accelerated queries in Access Plan Graphs
- Integrates with Query Monitor and OMPE for capturing query workloads for complete analysis
- Enable “what if” analysis

## ■ Benefits

- Shorten the process of selecting tables to be accelerated
- Visualize access paths of accelerated queries
- Increase productivity by working with accelerated queries through a unified interface
- Increase overall system capacity

The screenshot displays the 'Tuning Results(idaa\_tpcds\_10)' window. On the left is a navigation pane with options: Recommendations, View Workload Statements, RUNSTATS Script, Index Script, Analytic Acceleration Script (highlighted), Tuning Options Used, and Tuning Job Log. The main area shows a 'Choose an Accelerator' dropdown, a refresh icon, and buttons for 'Run Recommended DDL' and 'Open in New Window'. Below these are performance metrics and SQL scripts for table acceleration.

```

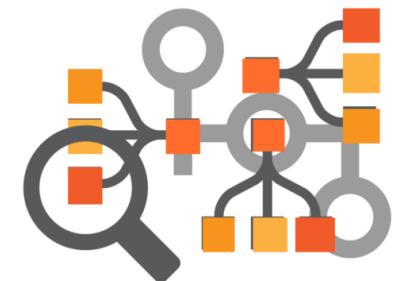
-- Estimated cost saving from query acceleration (sec): 9,217.06
-- Estimated CPU saving from query acceleration (sec): 2,999.34

--Script to add tables to accelerator

-- Table: TPCDS.STORE_SALES
-- Cardinality: 2,880,404.00
-- References to Table: 18.00
-- Cumulative Estimated Cost: 28,156,359.25
-- Cumulative CPU Cost: 6,189,292.00
-- NPAGES: 92,917.00

-- Table: TPCDS.CATALOG_SALES
-- Cardinality: 1,441,548.00
-- References to Table: 11.00
-- Cumulative Estimated Cost: 25,622.75
-- Cumulative CPU Cost: 5,526,377.00
-- NPAGES: 68,646.00

-- Table: TPCDS.DATE_DIM
-- Cardinality: 73,049.00
-- References to Table: 49.00
-- Cumulative Estimated Cost: 26,479.74
  
```



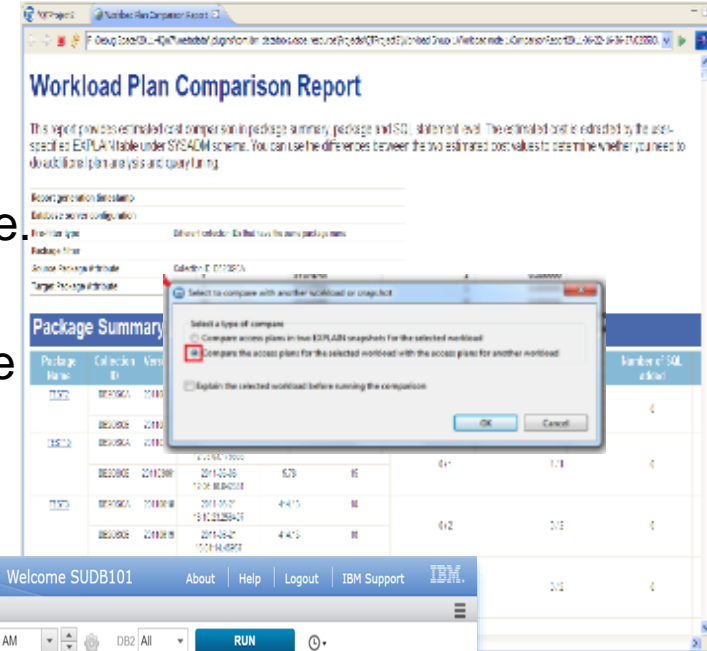
analyze

4

# Prevent problems before they impact the business

- Optimize beyond the prior level of service
  - Determine whether the later version of the collection has degraded performance
  - Determine whether any packages have errors.
  - Identify which packages have SQL statements that have degraded performance

Workload Comparison



## Available Actions

- Apply filters and review Comparison result
- Review comparison result
- Generate HTML comparison report or in csv
- Generate new query workload for tuning & perform analysis

## Enhancements

- Compare two different workloads

Workload Access Plan Comparison

Command	Program	Row Status	Exec Count	Calls	Elapsed
EMPFND	RETAINED	-1,176	Old: 2,104 New: 928 Change: -1,176 Percent Change: -55.89%	-2.891822	Old: 7.426930 New: 4.535108 Change: -2.891822 Percent Change: -38.94%
DPTUPR	RETAINED	-4,284	-2,112,745	-2.112745	-0.187570
PRJSEL	RETAINED	-1,873	-5,184	-1.037176	-0.06
STRSEL	RETAINED	-1,690	-1,690	-0.090531	-0.01
PRJANO	RETAINED	-121	-121	-0.016235	-0.00
CHCPADB	RETAINED	-2,298	-3,166	-2.293830	-0.13
DSNADMCD	RETAINED	-108	-164	-0.004759	-0.00
EMPUPD	RETAINED	-1,242	-1,242	-0.074590	-0.00
DPTANO	RETAINED	-804	-804	-0.044093	-0.00
DPTUPD	RETAINED	-2,489	-2,489	-0.232317	-0.01
PRJADD	RETAINED	-104,280	-121,180	-33.813864	-1.87
CHCDLRDB	RETAINED	-30	-30	-0.002591	-0.00

**Analysis**

**Elapsed Time**  
Value: -2.891822

Unaccounted Time  
Value: 0.000000  
Overall: 0.00%

Specific Elapsed  
Value: -0.101926  
Overall: 3.52%

CPU time  
Value: -1.283498  
Overall: 44.38%

**Total Delay Time**  
Value: -1.548210  
Overall: 53.54%

Other Delay  
Value: +0.000031  
Overall: -0.00% Percent: -0.00%

Total Lock Delay  
Value: +0.000001  
Overall: -0.00% Percent: -0.00%

Servtask Switch Delay Time  
Value: 0.000000

4

# Prevent problems before they impact the business

*migration comparison*

	A	B	C	D
1				
2	General information		Run A	Run B
3	DB2 Group name		0	0
4	DB2 Subsystem name		0	0
5	Workload description		???	???
6	DB2 Version, Mode		V10 NFM ???	V11 CM ???
7	DB2 code level		PUTxxxx	PUTxxxx
8	z/OS level		z/OS V1R13	z/OS V1R13
9	Processor Model (RMF CPU)		??? 2097-E64	??? 2097-E64
10	Statistics with ZOSMETRICS (or RMF report)			
11	Number processor (AVG per Interval)	QWOSLNCP	#DIV/0!	#DIV/0!
12	CPU utilization (AVG per Interval and per processor)	QWOSLPRU	#DIV/0!	#DIV/0!
13	Aggregated Accounting for each CONN TYPE or ACCOUNTING TRACE			
14	CONN TYPE		0	0
15	CL1 Elapsed / QUANTITY		#DIV/0!	#DIV/0!
16	CL1 CPU / QUANTITY		#DIV/0!	#DIV/0!
17	CL1 SE CPU / QUANTITY		#DIV/0!	#DIV/0!
18	CL2 Elapsed / QUANTITY		#DIV/0!	#DIV/0!
19	CL2 CPU / QUANTITY		#DIV/0!	#DIV/0!
20	CL2 SE CPU / QUANTITY		#DIV/0!	#DIV/0!
21	CL3 SUSP / QUANTITY		#DIV/0!	#DIV/0!
22	CL2 NOT ACCOUNT / QUANTITY		#DIV/0!	#DIV/0!
23	QUANTITY (Number of transaction aggregated for the connection type)		0	0
24	COMMIT / Rollback			
25	COMMIT		0	0
26	ROLLBACK		0	0
27	DML per COMMIT			
28	SELECT		#DIV/0!	#DIV/0!
29	INSERT		#DIV/0!	#DIV/0!
30	NUMBER OF ROWS		#DIV/0!	#DIV/0!
31	UPDATE		#DIV/0!	#DIV/0!
32	NUMBER OF ROWS		#DIV/0!	#DIV/0!
33	MERGE		#DIV/0!	#DIV/0!
34	DELETE		#DIV/0!	#DIV/0!
35	NUMBER OF ROWS		#DIV/0!	#DIV/0!
<span>DB2 11 ESP measurement result</span> <span>Input A Stats</span> <span>Input A plan</span> <span>Input A BP</span> <span>Input B Stats</span> <span>Input B plan</span> <span>Input B BP</span>				



Derived Worksheet with V11 ↔ V12 comparison

Import generated CSV data from V11 and V12 execution into the provided several worksheets

## Host variable Collection & Selectivity Override

- Why did the Db2 Optimizer choose that path?
- Helps users improve query access plans for dynamic queries with parameter markers
- The selectivity override feature utilizes parameter marker information
- Users can deploy a selectivity profile generated by this function to create better access plans.



```
SELECT * FROM  
EMPLOYEES WHERE  
SALARY BETWEEN ?  
AND ?
```



# 1 Create a Baseline

- Run a test application now to get a baseline.
- Average execution time for this application is: 125ms
- Note this query is well tune before selectivity override analysis
- Remember this number

### Execution Log

ID:	1484464703338
Name:	Appl_Ad_Hoc_4_min_01
Database Name:	WKLDDVR01
Start Time:	1484600890461
End Time:	1484601132118
Result:	The job executed successfully.

Commands Executed:

```
cd /tmp;  
/usr/bin/java -jar /home/db2inst1/hol/queryTuning/Appl_Ad_Hoc_01.jar 4 0 01;
```

Output from executing:

```
Connected to DEMOMVS  
Running program for: 4 minute(s)  
Starting time: 13:08:11:925  
.....  
Ending time: 13:12:12:005  
  
Execution count: 1908  
Average execution time: 125ms
```



# 2 Collect Activity

**Summaries**  
 Access and refine view of your system's query activity

**Collection Period**  
 Choose the Activity Browser data from a past time period

**Collect Host Variables**  
 Configure how the host variable information is captured

**Request host variable collection**

The screenshot displays the DB2 Query Monitor for z/OS v3.3 interface. The top section shows the 'Activity Browser' with a list of activity intervals. A callout points to the 'now' interval, indicating the collection period. Below this, a table shows query execution details for various commands. A second callout points to the 'Collect Host Vars' button, which is used to request host variable collection. The bottom section shows a detailed view of SQL execution, including SQL text, execution counts, and performance metrics.

Command	Plan	Authid	Exec Count	Calls	Elapsed	%Elap	AvgX Elapsed	CPU	%CPU	AvgX Cpu
DISTSERV		TSJAI	4	12	0.274883	0.08	0.068820	0.005883	2.07	0.0014
DB2		TSJAI	253	452	2.375971	0.75	0.009987	0.132582	48.43	0.0014

Command	SQL Text	Exec Count	Calls	DB2 SSID	Plan	Program	Current Schema	Section	Collection
	DELETE FROM ...	1	3				TSJAI		
	INSERT INTO C...	1	3				TSJAI		
	SELECT COUNT ...	1	3				TSJAI		
	UPDATE COMQ...	1	3				TSJAI		



# Analyze Hostvars details and identify candidate query

**Tune or Tune all**  
**For selectivity override**  
**analysis**

- With parameter marker, high elapse time, CPU time, execution count etc.

The screenshot shows the IBM Analytics Activity Browser interface. At the top, there are navigation tabs: Home, Alerts, Activity Browser, Configuration, and Action Console. Below these are filters for Sources (CQM Subsystems), Targets (DEMOMVS.CQ3), Intervals (1/23/2017, 6:00:00 AM), and a 'RUN' button. A toolbar contains buttons for 'Collect HostVars' (highlighted in red), 'Capture Host Variables', 'Show SQL', 'Tune All', and 'Details'. Below the toolbar is a table with columns: Command, SQL Text, Elapsed, %Elap, AvgX Elapsed, CPU, %CPU, and AvgX. The first row is highlighted, with red boxes around the 'Elapsed' value (1:35.081484) and the 'CPU' value (27.691391). Below the table, there are tabs for 'Analysis', 'SQL Text', 'HostVars', and 'General'. The 'SQL Text' tab is active, showing the SQL query for the first row. The query is: `select count () from GOSALESC03.CUST_CUSTOMER CUST_CUSTOMER, GOSALESC03.CUST_ORDER_HEADER CUST_ORDER_HEADER, GOSALESC03.CUST_ORDER_DETAIL CUST_ORDER_DETAIL where CUST_CUSTOMER.CUST_CODE = CUST_ORDER_HEADER.CUST_CODE and CUST_ORDER_HEADER.CUST_ORDER_NUMBER = CUST_ORDER_DETAIL.CUST_ORDER_NUMBER and CUST_CUSTOMER.GENDER_CODE = ? and CUST_ORDER_HEADER.CRDT_METHOD_CODE = ? and CUST_ORDER_DETAIL.CUST_QUANTITY = ?`. Red arrows point to the parameter markers '?' in the SQL text.



Workload Viewer

SQL Text	Plan	Collection	Program	Contoken	Section	Stmt Number	Statement ID	Current Schema	Exec Count	Elaps
SELECT "ROW...						0		TS5700	56	
INSERT INTO S...						0		TS5700	0	
DELETE FROM...						0		TS5700	56	
/* IBM_DSSNA...						0		TS5700	4	
/* IBM_DSSNA...						0		TS5700	4	
DECLARE DB...									8	
DECLARE MSG...									8	
INSERT INTO S...									7,048	
DELETE FROM...									8	
DECLARE THR...									16	
DELETE FROM...									16	
DELETE FROM...									48	
DELETE FROM...									24	
DELETE FROM...									8	
DELETE FROM...									8	
DELETE FROM...									28	

**Tune SQL**

Tuner Client

Data Server Manager (DSM) [Configure DSM...](#)

Query Workload Tuner

*Tuner is not available. Please check configuration.*

DB2 Subsystems

DBB6

Database Connection in DSM

CQM\_RS27DDS6\_PDS...

OK Close

**Log in to Data Server Manager**

In order to retrieve connection information, you need to provide your credentials for IBM Data Server Manager (DSM).

DSM User Name:

Password:

DSM: dsm:HTTP://mvt-vm-kjwin764:11080

Login Cancel

Show Raw SQL Text **Tune**

```
SELECT "ROWNUM", "TEXT"
FROM SYSIBM.SYSLOG ORDER
BY "ROWNUM"
```

(\*) Host variable collection ignores CURRENT SCHEMA. A result may contain host variable values collected when any value of CURRENT SCHEMA was in effect for the s

**IBM Data Server Manager** CQM\_RS27DDS6\_PDSOKO

Query and Workload Job Tuning Results(QM\_6841) x

Database:  Tuning type:  Status:  [More Filters...](#)

View Results Retune... Compare Set Retention... Cancel

Database	Job name	Created by	Tuning Type	SQL Text	Status	Progress
CQM_RS27DDS6_PDSOKO	QueryMonitor6778	admin	Single-query	SELECT "ROWNUM","TEXT" FROM SYS IBM.SYSLOG ORDER BY...	Succeeded	Completed
CQM_RS27DDS6_PDSOKO	QueryMonitor4516	admin	Single-query	SELECT "ROWNUM","TEXT" FROM SYS IBM.SYSLOG ORDER BY...	Succeeded	Completed
CQM_RS27DDS6_PDSOKO	QM_6841	admin	Workload	--	Succeeded	Completed

Home

Administer >

Run SQL

Monitor >

Jobs

Optimize >

Settings >

Help >



# 4 Tune selectivity override

Optimize / Tuning Jobs

Database: - All - Tuning type: - All - Status: - All - More Filters

View Results Retune Compare Set Retention Cancel Delete

View job status, recommendations and tuning options used.		Tuning Type	SQL Text	Status	Progress	St
DSNC03	Selectivity_Override_e_4323	dsmuser03	Workload	--	Succeeded	Completed 2012

Selectivity Override Job

- Go to View Workload Statements, you can see the query is Selectivity Override Candidate

- Then, select Host Variables

Optimize / Tuning Jobs

Query and Workload Jobs **Tuning Results(Selectivity\_Override\_4323)**

View Full Statement Tune Highlighted Statement View Access Plan Graph More Actions(2)

Default Schema	Explain Status	Selectivity Override Candidate	Statement Text	Host Variables
DSMADM	Yes	Yes	select count(*) from GOSALECT03.CUST_CUSTOMER CUST_CUSTOMER, GOSALECT03.CUST_ORDER_HEADER CUST_OR...	Refine

View Workload Statements >

RUNSTATS Script

Tuning Options Used



# Review analysis

- In this dialog, you can see:
  - parameter markers distribution
  - Weight of each parameter marker value set
  
- Select the sets (all) for Selectivity Override analysis
  
- Click Selectivity Override

Host variable/Parameter marker values (X) [Learn more](#)

---

Selectivity Override
View Full Statement

<input checked="" type="checkbox"/>	Set	Count	Weight	Position	Predicate Text	Candidate	Value
<input checked="" type="checkbox"/>	▼	1	18	0.18			
				1	"CUST_CUSTOMER"."G ENDER_CODE" = ?	Yes	1
				2	"CUST_ORDER_HEADE R"."CRDT_METHOD_CO DE" = ?	Yes	29
				3	"CUST_ORDER_DETAIL" ."CUST_QUANTITY" = ?	Yes	1
<input checked="" type="checkbox"/>	▶	2	15	0.15			
<input checked="" type="checkbox"/>	▶	3	14	0.14			

Execution count: 100  
Total literal value sets: 13 Selected: 13

Cancel



5

# View results and deploy the selectivity profile

- A selectivity override analysis job is created
  - Click View Results when it is completed

Database	Selectivity Override	User	Tuning Type	SQL Text
DSNC03	SelectivityOverride6612	dsmuser03	Selectivity Override Analysis	select co GOSALE MER C.
DSNC03	Selectivity_Overrid e_4323	dsmuser03	Workload	--

- Run recommended scripts
- Flush the statement cache

```

-- Estimated performance improvement: -110.12% faster
DELETE FROM DSMADM.DSN_USERQUERY_TABLE;

DELETE FROM DSMADM.DSN_PREDICAT_TABLE WHERE QUERYNO=310409815;

DELETE FROM DSMADM.DSN_PREDICATE_SELECTIMTY WHERE QUERYNO=310409815;

INSERT INTO DSMADM.DSN_USERQUERY_TABLE
(QUERYNO, SCHEMA, HINT_SCOPE, QUERY_TEXT, COLLECTION, VERSION,
PACKAGE, SELECTVY_OVERRIDE, ACCESSPATH_HINT, OPTION_OVERRIDE) VALUES(310409815,'DSMADM',0,CLOB('select count(*)
from GOSALE5CT03.CUST_CUSTOMER CUST_CUSTOMER,
GOSALE5CT03.CUST_ORDER_HEADER CUST_ORDER_HEADER,
GOSALE5CT03.CUST_ORDER_DETAIL CUST_ORDER_DETAIL
where CUST_CUSTOMER.CUST_CODE = CUST_ORDER_HEADER.CUST_CODE
and CUST_ORDER_HEADER.CUST_ORDER_NUMBER = CUST_ORDER_DETAIL.CUST_ORDER_NUMBER
and CUST_CUSTOMER.GENDER_CODE = ?
and CUST_ORDER_HEADER.CRD_METHOD_CODE=?
and CUST_ORDER_DETAIL.CUST_QUANTITY=?'),'','Y','N');
    
```





## Compare against baseline

- Run the test application again run after Selectivity Override analysis
- Average execution time for this application is: 92ms
- Improvement of 26%
- On an already well-tuned query!!

### Execution Log

ID: 148461703

Name: Appl

Database Name: > **26% faster!!**

Start Time: 148461703

End Time: 148461703

Result: The job executed successfully.

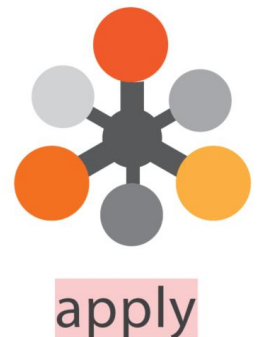
Commands Executed:

```
cd /tmp;
/usr/bin/java -jar /home/db2inst1/hol/queryTuning/Appl_Ad_Hoc_01.jar 4 0 01;
```

Output from executing:

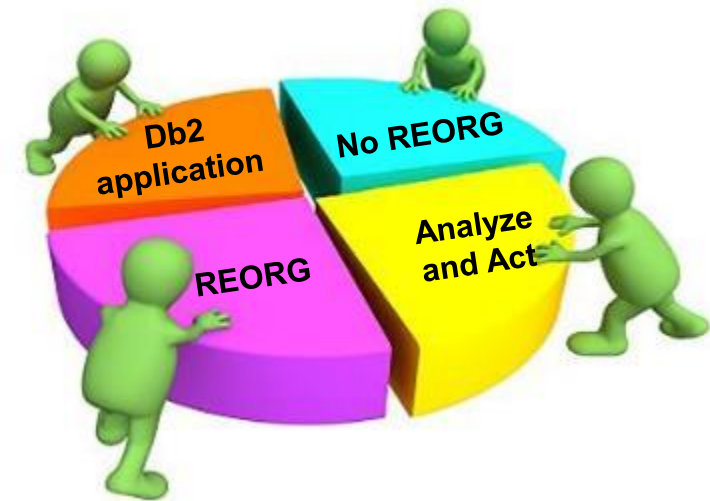
```
Connected to DEMOMVS
Running program for: 4 minute(s)
Starting time: 18:17:01:823
.....
Ending time: 18:21:01:913

Execution count: 2595
Average execution time: 92ms
```



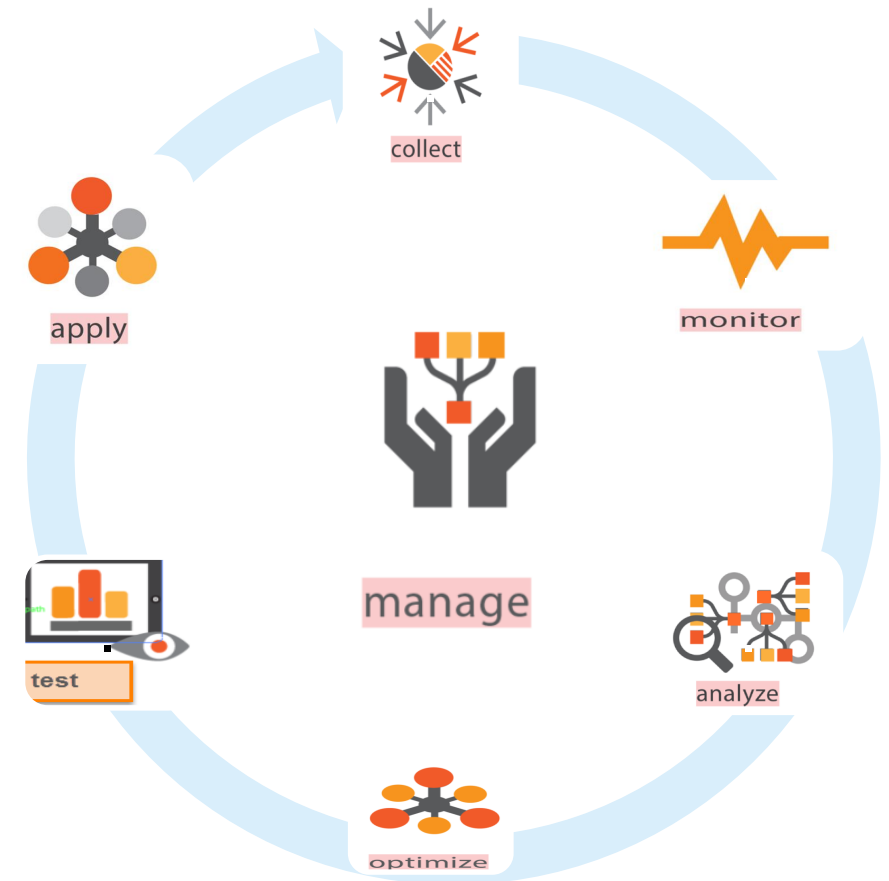
## Move from reactive to proactive to predictive results

- Db2 performance depends on “health” of Db2 objects
  - Often, a REORG is done on Db2 application objects whether needed or not, affecting resources and availability
  - The best REORG is the one you don’t perform
- Consider the following:
  - Granular query monitoring and analysis at object level:
  - Ability to detect, apply intelligence and decide best action for object:
    - Perform REORG if performance will improve
    - Skip REORG if determined no performance benefit
- Aligning data management with application needs
  - Hands-free performance monitoring tied into maintenance actions
  - Improves application performance, reduce system and IT resources



## *IBM delivers complete Db2 performance management*

- Reduce costs of Db2 for z/OS and applications
  - Improve performance of all package applications
  - Tune performance of query warehouse
- Identify and solve faster closing the loop on problem resolution
- Replace ad-hoc methods with integrated solutions for scalable, robust approach to performance management
- Improve performance and time to resolution by up to 50%
- Speed Db2 and application migration with comprehensive comparison capabilities



Thank  
You

