

Monitoring Your DB2 LUW Database with Just SQL



Chris Eaton
WW Technical DB2 Specialist
IBM Toronto Lab
ceaton@ca.ibm.com

Agenda

- Introduction to DB2 Monitoring Internals
- Introduction to monitoring via SQL
- Monitoring Status and Performance with SQL
- Monitoring Health and Diagnosing problems with SQL
- Using the Database Health Monitor
- Using Optim Performance Manager included with DB2 AESE

Introduction to DB2 Monitoring Internals

DB2 Monitoring Internals

- **What is Snapshot monitoring?**
 - A “picture” of the state of the DB2 system at a point in time
 - A report on a set of counters (mostly) stored inside DB2
 - Just like a camera, a snapshot is initiated by a human

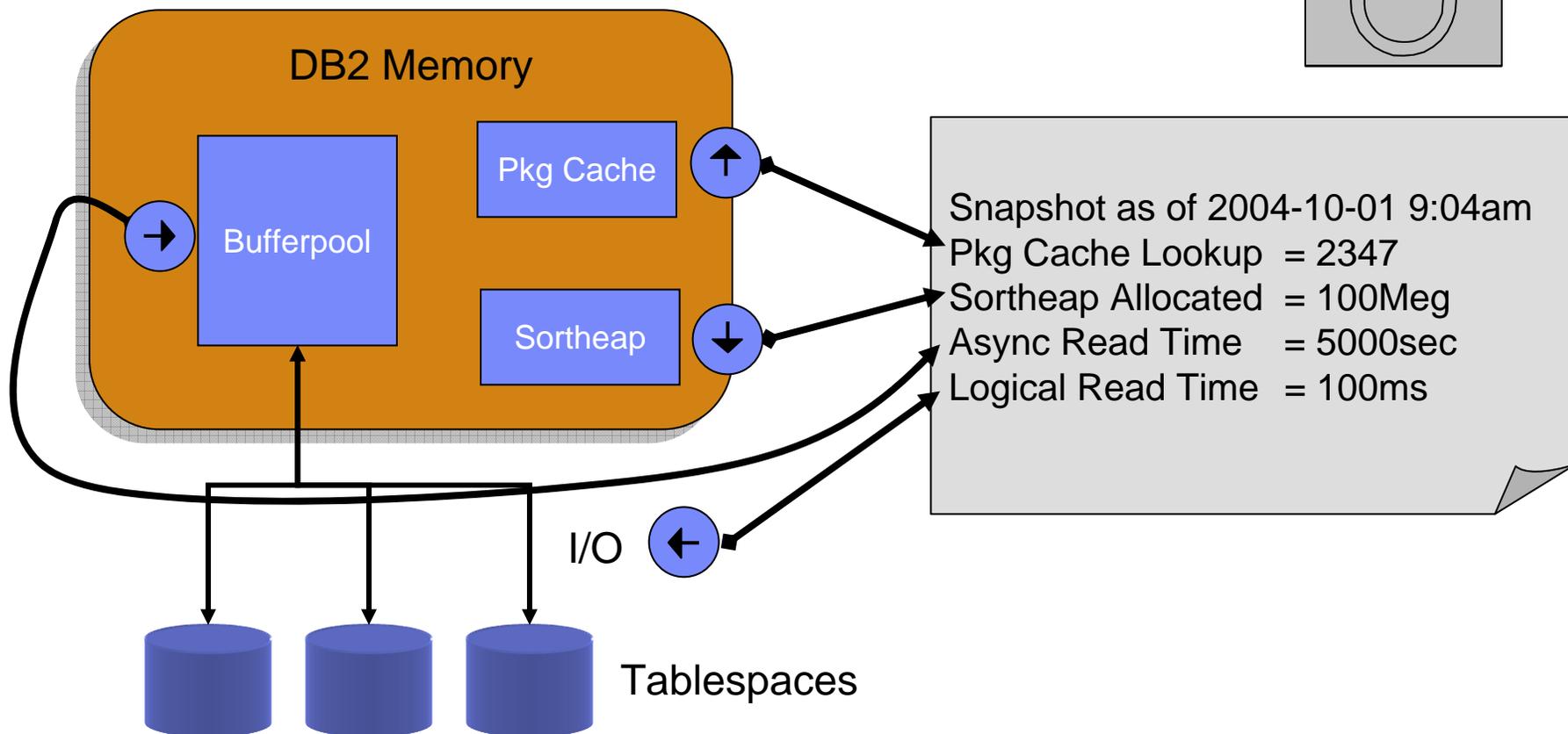
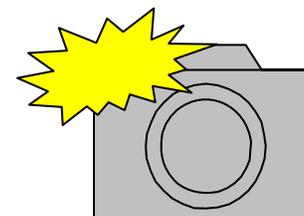
- **What is an Event monitor?**
 - A similar set of information (counters mostly) triggered by a defined event
 - For example, information about what an application did when it disconnects from the database
 - We won't discuss Event Monitoring in this session

Types of Monitor Elements

- **Counters**
 - Measures the number of times an activity occurs (always increases) – Can be reset
 - E.g.: Rows read from a table, number of physical page reads, etc.
- **Gauges**
 - Indicates the current value of an item (may increase or decrease over time) – not reset (value are current state)
 - E.g.: Number of currently active sorts, amount of log space currently allocated, etc.
- **Information**
 - Reference type information about a monitor element – not reset
 - E.g.: Server Platform, Authentication ID of connected user, etc.
- **Timestamp**
 - Indicates the date and time an activity took place. – not reset. Number of seconds and microseconds since Jan 1, 1970
 - E.g.: Last time a database was backed up, snapshot time, etc.
- **Time**
 - Returns the number of seconds and microseconds spent on an activity – Can be reset
 - E.g.: Time spent reading data pages, elapsed time of a unit of work, etc.

How Does It Work?

db2 get snapshot for database ...



Command Line Syntax

- **GET SNAPSHOT FOR**
 - DATABASE MANAGER
 - DATABASE ON <dbname>
 - TABLESPACES ON <dbname>
 - TABLES ON <dbname>
 - BUFFERPOOLS ON <dbname>
 - LOCKS ON <dbname>
 - APPLICATIONS ON <dbname>
 - DYNAMIC SQL ON <dbname>

- **You must have SYSADM, SYSCTRL, SYSMANT or SYSMON authority**

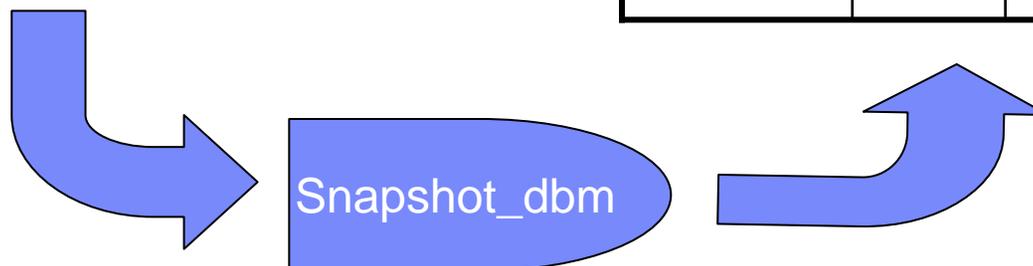
Introduction to Monitoring via SQL Functions

What's a Table UDF

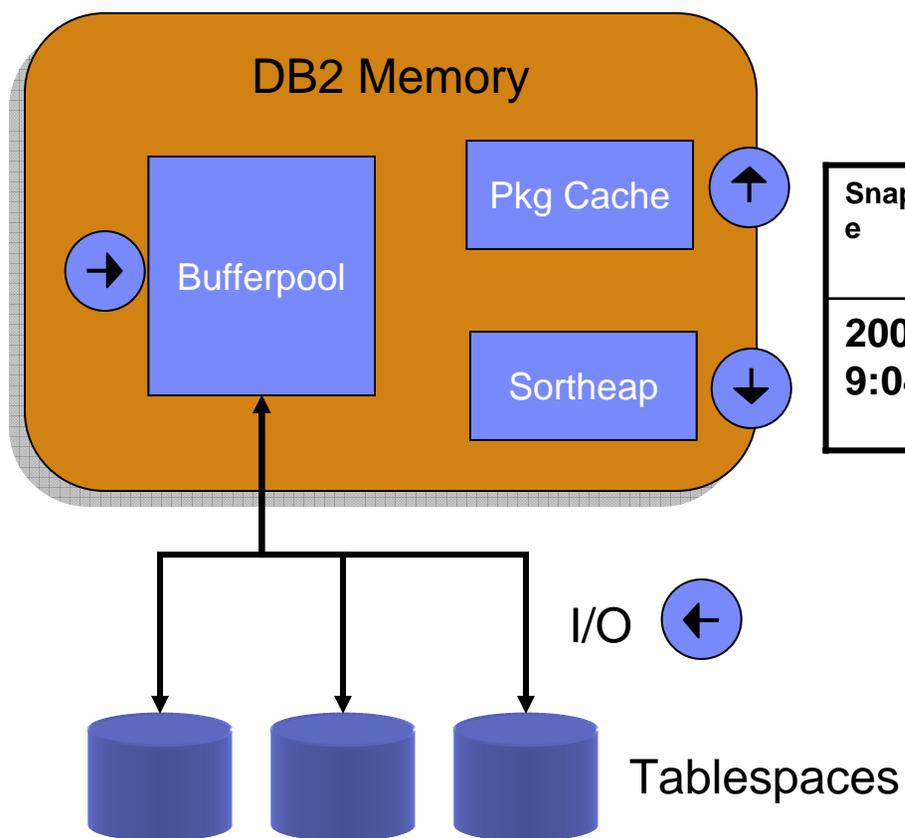
- **UDF = User Defined Function**
 - Shipped with DB2 – not user defined
- **A function that takes a structured set of information and makes appear to be a table**

Instance name = DB2
Database manager status = Active
Service level = s040219
Private Sort heap allocated = 0
Private Sort heap high water mark = 277

Instance_name	Status	Serv_level	Priv_sort_alloc	Priv_sort_high
DB2	Active	S040219	0	277



How Does It Work?



SELECT * FROM TABLE(SNAP_GET_DBM)

Snapshot_time	Pkg_cache_lookup	Sortheap_alloc	Async_read_time	Logical_read_time
2004-10-01 9:04am	2347	100	5000	100

The Syntax of a Select Statement

Name given
to table

```
select * from table(snap_get_dbm(-1)) as sntable
```

Table
Function

Name of the
table function

Argument
-1 = current partition number

```
select * from table(snap_get_db_v91('',-1)) as sntable
```

Arguments
"" = current database
-1 = current partition number

DB2 9 Makes Your Life Simpler

– Administrative Views

- **Table Functions still exist but now you have VIEWS**
- **All views are in the SYSIBMADM schema**
- **Convert coded values to text strings**
- **Can be a control point to allow people with lower authority to view monitor information**
 - Grant select on view and execute on table function

SNAPSHOT Views

- Database Manager
 - SNAPDBM
 - SNAPDBM_MEMORY_POOL
- Database Level
 - **SNAPDB**
 - SNAPDB_MEMORY_POOL
 - SNAPBP
 - SNAPBP_PART
 - SNAPHADR
- Application Level
 - **SNAPAPPL**
 - **SNAPAPPL_INFO**
 - **SNAPLOCKWAIT ***
 - SNAPSTMT
 - SNAPAGENT
 - SNAPSUBSECTION
 - SNAPAGENT_MEMORY_POOL
 - SNAPDYN_SQL
 - SNAPLOCK *
- Object Level
 - SNAPTAB
 - SNAPTAB_REORG
 - SNAPTbsp
 - SNAPTbsp_PART
 - SNAPTbsp_QUIESCER
 - SNAPCONTAINER
 - SNAPTbsp_RANGE
 - SNAPUTIL
 - SNAPUTIL_PROGRESS
 - SNAPDETAILLOG
 - SNAPSTORAGE_PATHS
- Database Partitioning Feature (DPF)
 - SNAPFCM
 - SNAPFCM_PART

* Deprecated in 9.7 FP1

“Convenience” Monitor Views

- APPLICATIONS
- APPL_PERFORMANCE
- BP_HITRATIO
- BP_READ_IO
- BP_WRITE_IO
- CONTAINER_UTILIZATION
- LOCKS_HELD *
- LOCKWAIT *
- LOG_UTILIZATION
- LONG_RUNNING_SQL
- QUERY_PREP_COST
- TBSP_UTILIZATION
- TOP_DYNAMIC_SQL

* Deprecated in 9.7 FP1

Administrative Views

- **ADMINTABINFO**
- **ADMINTABCOMPRESSINFO**
- **ADMIN_GET_INDEX_INFO**
- **ADMIN_GET_INDEX_COMPRESS_INFO**
- **ADMIN_EST_INLINE_LENGTH**
- **ADMIN_IS_INLINED**
- **ADMIN_GET_DBP_MEM_USAGE**
- **DBCFCG**
- **DBMCFG**
- **REG_VARIABLES**
- **DB_PARTITIONS**
- **DB_HISTORY**

New 9.7 Monitor Functions

New Time Spent and Time Waiting Metrics – find bottlenecks

- **Application Information**
 - **MON_GET_CONNECTION**
 - **MON_GET_CONNECTION_DETAILS**
 - **MON_GET_PKG_CACHE_STMT**
 - **MON_GET_UNIT_OF_WORK**
 - **MON_GET_UNIT_OF_WORK_DETAILS**
- **Workload Management**
 - **MON_GET_WORKLOAD**
 - **MON_GET_WORKLOAD_DETAILS**
 - **MON_GET_SERVICE_SUBCLASS**
 - **MON_GET_SERVICE_SUBCLASS_DETAILS**
- **Object**
 - **MON_GET_TABLE**
 - **MON_GET_INDEX**
 - **MON_GET_TABLESPACE**
 - **MON_GET_CONTAINER**
 - **MON_GET_BUFFERPOOL**
 - **MON_GET_EXTENT_MOVEMENT_STATUS**

Monitoring Performance With SQL Select Statements



Long Running SQL

```
SELECT ELAPSED_TIME_MIN,  
       SUBSTR(AUTHID,1,10) AS AUTH_ID,  
       AGENT_ID,  
       APPL_STATUS,  
       SUBSTR(STMT_TEXT,1,20) AS SQL_TEXT  
FROM   SYSIBMADM.LONG_RUNNING_SQL  
WHERE  ELAPSED_TIME_MIN > 0  
ORDER BY ELAPSED_TIME_MIN DESC
```

ELAPSED_TIME_MIN	AUTH_ID	AGENT_ID	APPL_STATUS	SQL_TEXT
6	EATON	878	LOCKWAIT	update org set deptn

Buffer Pool Query

Display buffer pool hit ratios (data, index and XML)

```
SELECT      SUBSTR(BP_NAME,1,20) as BP_NAME,  
            TOTAL_HIT_RATIO_PERCENT as ALL_HR,  
            DATA_HIT_RATIO_PERCENT as DATA_HR,  
            INDEX_HIT_RATIO_PERCENT as INX_HR,  
            XDA_HIT_RATIO_PERCENT as XML_HR  
FROM SYSIBMADM.BP_HITRATIO;
```

BP_NAME	ALL_HR	DATA_HR	INX_HR	XML_HR
IBMDEFAULTBP	98	80	99	0
LARGE_BP	99	99	0	0
SMALL_BP	25	25	0	0

Package Cache Query

- **Look at all the queries in the package cache**
 - Both Dynamic and Static
 - See execution time, wait time (by component), and much more

```

SELECT
  SUBSTR(STMT_TEXT,1,20) AS STMT,
  SECTION_TYPE AS TYPE,
  NUM_EXECUTIONS,
  TOTAL_ACT_TIME AS TOTAL_TIME,
  TOTAL_ACT_WAIT_TIME AS WAIT_TIME
FROM TABLE(MON_GET_PKG_CACHE_STMT("","",",-1))

```

STMT	TYPE	NUM_EXECUTIONS	TOTAL_TIME (ms)	WAIT_TIME(ms)
Select * from emp	D	10	123	7
with aa as (select *	D	100	2845	860

Package Cache Query

- **Other useful bits of information in MON_GET_PKG_CACHE_STMT function**
 - NUM_EXECUTIONS
 - PREP_TIME
 - TOTAL_ACT_TIME
 - TOTAL_ACT_WAIT_TIME
 - TOTAL_CPU_TIME
 - LOCK_WAIT_TIME
 - TOTAL_SECTION_SORT_TIME
 - TOTAL_SECTION_SORTS
 - LOCK_ESCALS
 - LOCK_WAITS
 - ROWS_MODIFIED
 - ROWS_READ
 - TOTAL_SORTS
 - SORT_OVERFLOW
 - DEADLOCKS
 - LOCK_TIMEOUTS
 - LOG_BUFFER_WAIT_TIME
 - LOG_DISK_WAIT_TIME
 - STMT_TEXT CLOB(2MB)

Lock Wait Query

```
select substr(ai_h.appl_name,1,10) as "Hold App",
       substr(ai_h.primary_auth_id,1,10) as "Holder",
       substr(ai_w.appl_name,1,10) as "Wait App",
       substr(ai_w.primary_auth_id,1,10) as "Waiter",
       lw.lock_mode as "Hold Mode",
       lw.lock_object_type as "Obj Type",
       substr(lw.tabname,1,10) as "TabName",
       substr(lw.tabschema,1,10) as "Schema",
       timestampdiff(2,char(lw.snapshot_timestamp -
lw.lock_wait_start_time))
       as "waiting (s)"
```

Who is holding
the lock

Who is waiting on the
lock

How long is the wait

```
from sysibmadm.snapappl_info ai_h,
     sysibmadm.snapappl_info ai_w, sysibmadm.snaplockwait lw
where lw.agent_id = ai_w.agent_id
and   lw.agent_id_holding_lk = ai_h.agent_id
```

Hold App	Holder	Wait App	Waiter	Hold Mode	Obj Typ	TabName	Schema	waiting
db2bp.exe	CEATON	db2bp.exe	USER2	X	Row	T1	CEATON	15
db2bp.exe	CEATON	db2bp.exe	USER1	X	Row	T1	CEATON	6

Excessive Sorting

Show the sort time, and wait time for all sorts by connection

```
SELECT
  APPLICATION_HANDLE AS APP_HDL,
  SUBSTR(CLIENT_USERID,1,10) AS USERID,
  TOTAL_SECTION_SORTS AS NUM_SORTS,
  TOTAL_SECTION_SORT_TIME AS TOTAL_TIME,
  TOTAL_SECTION_SORT_PROC_TIME AS SORT_TIME,
  TOTAL_SECTION_SORT_TIME -
  TOTAL_SECTION_SORT_PROC_TIME AS WAIT_TIME
FROM TABLE(MON_GET_CONNECTION(NULL,-1))
```

APP_HDL	USERID	NUM_SORTS	TOTAL_TIME	SORT_TIME	WAIT_TIME
7	CEATON	36	7579	7495	84

Top Consuming Transactions

Show the transactions with the most CPU and most Wait Time

```
SELECT
  APPLICATION_HANDLE AS APP_HDL,
  SUBSTR(CLIENT_USERID,1,10) AS USERID,
  TOTAL_RQST_TIME,
  TOTAL_CPU_TIME,
  TOTAL_WAIT_TIME,
  CLIENT_IDLE_WAIT_TIME
FROM TABLE(MON_GET_UNIT_OF_WORK(NULL,-1))
```

New in FP1

- Unit of Work monitor also includes

TOTAL_COMPILE_TIME	BIGINT	Reserved for future use.	
TOTAL_COMPILE_PROC_TIME	TOTAL_APP_ROLLBACKS	BIGINT	Reserved for future use.
TOTAL_COMPILATIONS	INT_ROLLBACKS	BIGINT	Reserved for future use.
TOTAL_IMPLICIT_COMPILE_TIME	TOTAL_RUNSTATS_TIME	BIGINT	Reserved for future use.
TOTAL_IMPLICIT_COMPILE_PROC_TIME	TOTAL_RUNSTATS_PROC_TIME	BIGINT	Reserved for future use.
TOTAL_IMPLICIT_COMPILATIONS	TOTAL_RUNSTATS	BIGINT	Reserved for future use.
TOTAL_SECTION_TIME	TOTAL_REORG_TIME	BIGINT	Reserved for future use.
TOTAL_SECTION_PROC_TIME	TOTAL_REORG_PROC_TIME	BIGINT	Reserved for future use.
TOTAL_APP_SECTION_EXECUTIONS	TOTAL_REORGS	BIGINT	Reserved for future use.
TOTAL_ACT_TIME	TOTAL_LOAD_TIME	BIGINT	Reserved for future use.
TOTAL_ACT_WAIT_TIME	TOTAL_LOAD_PROC_TIME	BIGINT	Reserved for future use.
ACT_RQSTS_TOTAL	TOTAL_LOADS	BIGINT	Reserved for future use.
TOTAL_ROUTINE_TIME	CAT_CACHE_INSERTS	BIGINT	Reserved for future use.
TOTAL_ROUTINE_INVOCATIONS	CAT_CACHE_LOOKUPS	BIGINT	Reserved for future use.
TOTAL_COMMIT_TIME	PKG_CACHE_INSERTS	BIGINT	Reserved for future use.
TOTAL_COMMIT_PROC_TIME	PKG_CACHE_LOOKUPS	BIGINT	Reserved for future use.
TOTAL_APP_COMMITS	THRESH_VIOLATIONS	BIGINT	Reserved for future use.
INT_COMMITS	NUM_LW_THRESH_EXCEEDED	BIGINT	Reserved for future use.
TOTAL_ROLLBACK_TIME	UOW_LOG_SPACE_USED	BIGINT	Reserved for future use.
TOTAL_ROLLBACK_PROC_TIME	ADDITIONAL_DETAILS	BLOB(100K)	Reserved for future use.

Monitoring Health And Status With SQL Select Statements



Monitoring Table Access

Show the most active tables

```

SELECT
    SUBSTR(TABSCHEMA,1,10) AS SCHEMA,
    SUBSTR(TABNAME,1,20) AS NAME,
    TABLE_SCANS,
    ROWS_READ,
    ROWS_INSERTED,
    ROWS_DELETED
FROM TABLE(MON_GET_TABLE("",-1))
ORDER BY ROWS_READ DESC
FETCH FIRST 5 ROWS ONLY
  
```

SCHEMA	NAME	TABLE_SCANS	ROWS_READ	ROWS_INSERTED	ROWS_DELETED
CEATON	WIKI_ACTIONS	14	6608	500	0
SYSIBM	SYSTABLES	16	6161	0	0
CEATON	WIKI_VISITORS	12	5664	0	70
SYSTOOLS	HMON_ATM_INFO	19	3627	0	0
SYSIBM	SYSINDEXES	0	348	0	0

Monitoring Index Access

- **Show me the indexes that have been most active**
 - Metrics will only be returned for indexes on tables that have been accessed since the database was activated.

```

SELECT
  SUBSTR(TABSCHEMA,1,10) AS SCHEMA,
  SUBSTR(TABNAME,1,20) AS NAME,
  IID,  NLEAF, NLEVELS,
  INDEX_SCANS,
  KEY_UPDATES,
  BOUNDARY_LEAF_NODE_SPLITS +
  NONBOUNDARY_LEAF_NODE_SPLITS AS PAGE_SPLITS
FROM TABLE(MON_GET_INDEX("","",-1))
ORDER BY INDEX_SCANS DESC
FETCH FIRST 5 ROWS ONLY

```

SCHEMA	NAME	IID	NLEAF	NLEVELS	INDEX_SCANS	UPDATES	SPLITS
SYSTOOLS	HMON_ATM_INFO	1	2	2	754	0	0
SYSIBM	SYSUSERAUTH	1	8	2	425	0	0
SYSIBM	SYSPLANAUTH	1	9	2	192	0	0
SYSIBM	SYSTABLES	1	6	2	186	0	0
SYSIBM	SYSINDEXES	2	5	2	145	0	0

SQL to View Notification Log

- Show me all the Critical and Error messages in the last 24 hours

```
SELECT TIMESTAMP, SUBSTR(MSG,1,400) AS MSG
FROM SYSIBMADM.PDLOGMSGS_LAST24HOURS
WHERE MSGSEVERITY IN ('C','E')
ORDER BY TIMESTAMP DESC
```

TIMESTAMP	MSG
2009-03-16-09.41.47.673002	ADM6044E The DMS table space "SMALLTBSP" (ID "2") is full. If this is an autoresize or automatic storage DMS tablespace, the maximum table space size may have been reached or the existing containers or storage paths cannot grow any more. Additional space can be added to the table space by either adding new containers or extending existing ones using the ALTER TABLESPACE SQL statement.

SQL to View Database History

- Show the average and maximum time taken to perform full backups

```
SELECT AVG(TIMESTAMPDIFF(QUARTER,  
START_TIME      SQLCODE      CMD  
-----  
20061114093635      -204  DROP TABLESPACE IBMDB2SAMPLEXML  
20061218125352      -1422 CREATE  REGULAR  TABLESPACE SMALLTSP  
WHERE OPERATION = 'B'  
AND OPERATIONTYPE = 'F'
```

```
AVG_BTIME      MAX_BTIME  
-----  
17              25
```

Finding the Log Hog

Display information about the application that currently has the oldest uncommitted unit of work

```
SELECT      AI.APPL_STATUS as Status,
            SUBSTR(AI.PRIMARY_AUTH_ID,1,10) AS "Authid",
            SUBSTR(AI.APPL_NAME,1,15) AS "Appl Name",
            INT(AP.UOW_LOG_SPACE_USED/1024/1024)
            AS "Log Used (M)",
            INT(AP.APPL_IDLE_TIME/60) AS "Idle for (min)",
            AP.APPL_CON_TIME AS "Connected Since"
FROM        SYSIBMADM.SNAPDB DB,
            SYSIBMADM.SNAPAPPL AP,
            SYSIBMADM.SNAPAPPL_INFO AI
WHERE       AI.AGENT_ID = DB.APPL_ID_OLDEST_XACT
AND        AI.AGENT_ID = AP.AGENT_ID;
```

What's New in 9.7 FP1

- **CREATE EVENT MONITOR FOR PACKAGE CACHE**
 - records events from both dynamic and static SQL when they are flushed from package cache
 - Information collected same as `MON_GET_PKG_CACHE_STMT`
- **Can view the information from event monitor as**
 - An XML document created by the new `EVMON_FORMAT_UE_TO_XML` table function
 - Relational tables populated by the new `EVMON_FORMAT_UE_TO_TABLES` procedure

Must run db2updv97

New Lightweight Lock Monitors

- **MON_GET_APPL_LOCKWAITS table function**
 - Returns information about the locks that all applications are waiting to acquire
- **MON_GET_LOCKS table function**
 - Returns a list of all locks held
- **MON_FORMAT_LOCK_NAME table function**
 - Formats the internal lock name and returns details about the lock in a row-based format. Each row consists of a key-value pair pertaining to a particular lock.
- **MON_LOCKWAITS View**
 - Returns information about agents working on behalf of applications that are waiting to obtain locks in the currently connected database.
- **Deprecated:**
 - SNAPLOCK, SNAPLOCKWAIT, LOCKS_HELD, LOCKWAITS views

Lock Wait Query

```

select substr(ai_h.appl_name,1,10) as "Hold App",
       substr(ai_h.primary_auth_id,1,10) as "Holder",
       substr(ai_w.appl_name,1,10) as "Wait App",
       substr(ai_w.primary_auth_id,1,10) as "Waiter",
       lw.lock_mode as "Hold Mode",
       lw.lock_object_type as "Obj Type",
       substr(lw.tabname,1,10) as "TabName",
       substr(lw.tabschema,1,10) as "Schema",
       timestampdiff(2,char(lw.snapshot_timestamp -
       lw.lock_wait_start_time))
           as "waiting (s)"
from sysibmadm.snapappl_info ai_h,
     sysibmadm.snapappl_info ai_w, sysibmadm.snaplockwait lw
where lw.agent_id = ai_w.agent_id
and   lw.agent_id_holding_lk = ai_h.agent_id

```

Who is holding
the lock

Who is waiting on the
lock

How long is the wait

Hold App	Holder	Wait App	Waiter	Hold Mode	Obj Typ	TabName	Schema	waiting
db2bp.exe	CEATON	db2bp.exe	USER2	X	Row	T1	CEATON	15
db2bp.exe	CEATON	db2bp.exe	USER1	X	Row	T1	CEATON	6

New Lightweight Version

```
select substr(HLD_APPLICATION_NAME,1,10) as "Hold App",  
       substr(HLD_USERID,1,10) as "Holder",  
       substr(REQ_APPLICATION_NAME,1,10) as "Wait App",  
       substr(REQ_USERID,1,10) as "Waiter",  
       LOCK_MODE as "Hold Mode",  
       LOCK_OBJ_TYPE as "Obj Type",  
       TABNAME,1,10) as "TabName",  
       TABSCHEMA,1,10) as "Schema",  
       LOCK_WAIT_ELAPSED_TIME as "waiting (s)"  
from SYSIBMADM.MON_LOCKWAITS;
```

Also available:

```
REQ_STMT_TEXT  
HLD_CURRENT_STMT_TEXT  
LOCKNAME
```

MON_FORMAT_LOCK_NAME

```
SELECT SUBSTR(NAME,1,20) AS NAME,
       SUBSTR(VALUE,1,50) AS VALUE
FROM TABLE(
  MON_FORMAT_LOCK_NAME(
    '0000000E000000000000B00C152'))
```

NAME

LOCK_OBJECT_TYPE

TBSP_NAME

TABSCHEMA

TABNAME

ROWID

VALUE

ROW

PRODTBSPACE1

CEATON

PRODUCTS

00 00 00 0C 00 C1

Other New Admin Views in FP1

- **MON_BP_UTILIZATION**
- **MON_TBSP_UTILIZATION**
- **MON_LOCKWAITS**
- **MON_PKG_CACHE_SUMMARY**
- **MON_CURRENT_SQL**
- **MON_CURRENT_UOW**
- **MON_SERVICE_SUBCLASS_SUMMARY**
- **MON_WORKLOAD_SUMMARY**
- **MON_CONNECTION_SUMMARY**
- **MON_DB_SUMMARY**

Data Studio Health Monitor



Data Studio Health Monitor

- **Included with DB2**
 - Freely downloadable from www.ibm.com/software/data/optim/data-studio/
- **View system health at a glance.**
 - Visualize warnings and problem areas instantly
 - Configure alert thresholds for health indicators, such as data server status and space utilization
- **Browse alert history**
 - Collect and retain alert history for up to **seven days**.
 - Display alert statistics by time period, database, or alert type.
- **Manage current application connections.**
 - Track information such as rows read and idle time for currently connected applications.
 - Verify that applications can access the database.
 - Force applications to enhance system performance.
- **View the current state of the table spaces of your database.**
 - View information such as state, total size, and current utilization for the table spaces of your databases.
- **View the status of utilities operating on your database.**

Quickly Visualize High Level Database Status

The screenshot shows the 'Data Studio Health Monitor' interface. At the top, there's a navigation bar with 'admin | Log out | About'. Below that, a toolbar contains 'Task Manager', 'Manage Database Connections', and 'Welcome - My Optim Central'. The main area is titled 'Health Summary' and includes a 'Recent 60 minutes' filter and buttons for 'Configure Health Summary...', 'Configure Alerts...', and 'Open Dashboard...'. On the left, an 'Alert Severity' sidebar lists 'All', 'Critical Alerts (0)', 'Warning Alerts (0)', and 'Critical and Warning Alerts (0)'. The central table displays health data for 'All' and 'SAMPLE' data sources. The table has columns for 'Alerts' (Critical, Warning), 'System' (Data Server status, Connections, Storage), and 'Data Source' (Recovery). The 'All' row shows 0 Critical, 0 Warning, a green diamond for Data Server status, and dashes for Connections, Storage, and Recovery. The 'SAMPLE' row shows 0 Critical, 0 Warning, a green diamond for Data Server status, and dashes for Connections, Storage, and Recovery. At the bottom, it indicates '2 total items' and '10 Items per page', 'Page 1 of 1'.

Data Source	Alerts		System			Data Source
	Critical	Warning	Data Server status	Connections	Storage	Recovery
All	0	0	◆	--	--	--
SAMPLE	0	0	◆	--	--	--

Manage Current Application Connections

- See the state of each application connected to the database

The application connections that are listed are currently active for the selected database. To improve performance, you can disconnect applications.

  SAMPLE

Last updated: 02/28/2011 01:48:32 PM

Agent ID	Name	Application ID	Authorization ID	Status	Client Product	Client ID	Idle Time
5432	db2bp.exe	*LOCAL.DB2.11...	CHRIS	Waiting for a Lock	SQL09070	2432	0
5458	db2bp.exe	*LOCAL.DB2.11...	CHRIS	Unit of Work Waiting	SQL09070	7360	184
5410	db2jcc_application	9.23.36.75.443...	CHRIS	Unit of Work Waiting	JCC03580	0	590
5416	db2fw1	*LOCAL.DB2.11...	CHRIS	Database Connection Complet...	JCC03580	0	593
5415	db2fw0	*LOCAL.DB2.11...	CHRIS	Database Connection Complet...	JCC03580	0	593



Sort Applications by Rows Read, Written, Idle Time

- Easy to see if applications are stuck waiting and for how long
- Are there applications doing table scans?

Agent ID	Authorization ID	Status	Idle Time	Rows Read <input type="button" value="v"/>	Rows Written
5410	CHRIS	Unit of Work Waiting	815	181	0
5432	CHRIS	Waiting for a Lock	0	77	7
5458	CHRIS	Unit of Work Waiting	409	2	1
5416	CHRIS	Database Connection Complet...	818	0	0
5415	CHRIS	Database Connection Complet...	818	0	0

View Tablespace Utilization and Container locations

- Show free space and space consumed for each tablespace
- Drill down to see the containers for each tablespace

ID	Name	Type	Content type	State	Utilization 	Free Size (KB)
0	SYSCATSPACE	DMS	ANY	NORMAL	81.47%	18208
2	USERSPACE1	DMS	LARGE	NORMAL	44.88%	17920
4	IBMDB2SAMPLE...	DMS	LARGE	NORMAL	35.43%	20992
3	IBMDB2SAMPLE...	DMS	LARGE	NORMAL	21.25%	25600
5	SYSTOOLSPACE	DMS	LARGE	NORMAL	2.73%	31840
1	TEMPSPACE1	SMS	SYSTEMP	NORMAL	--	--
6	SYSTOOLSTMPS...	SMS	USRTEMP	NORMAL	--	--

Optim Performance Manager

See the snapshot history



Optim Performance Manager - Overview

Dashboard Time Slider and Time Controls

The screenshot shows a dashboard interface for time-based data. It includes a 'Recent' button, a clock icon, a date and time display (04/09/10 10:22), a time slider showing a 1-hour range from 12:22 to 13:22 on 04/09/10, and a 'Duration' dropdown menu set to 1 Hour. The interface also features a 'History' button, an 'Aggregation level' dropdown, and a 'Hide' button. A message 'Not enough performance data is available...' is displayed below the time slider.

Recent Button
Displays the latest collected data. Content is refreshed when a new sample is taken.

Clock Button
Indicates the time remaining until the dashboard is refreshed. Click to start or stop refresh.

Zoom Out
Shows more of the timeline & more data.

Zoom In
Shows less of the timeline to be able to position the time slider to specific data.

Back and Forward to Data Point
These two arrows move the time slider to the previous or next data point.

End Time
Specifies the latest time on the time slider clock.

History Button
Displays data based on the position of the time slider on the time line.

Aggregation Level
Indicates the level of granularity of the data.

Missing Data Points
Missing lines indicate that no monitored data points are available.

Data Point Line
Blue lines indicate points in time for which monitoring data is available.

Hide Button
Click to hide time controls.

Time slider & line
Indicates the range of time shown in the dashboard.

Duration
Specifies how much data is shown at one time in the dashboard.

“Diagnose” using OPM

Extended Insight Analysis Dashboard

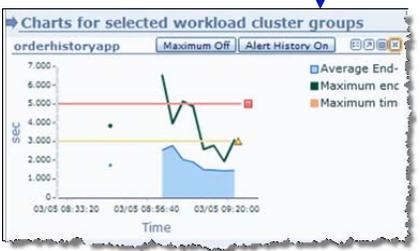
View any and all information through time slider controls

Set OPM to run in your Tivoli Enterprise Portal Console

Alert by the connection attribute grouping

Graph	Workload Cluster Group/Workload Cluster	Average End-to-End Response Time (s)	Maximum Inflight Elapsed Time (s)	Maximum End-to-End Response Time (s)	Average Data Server Time (s)	Average Network Time (s)	Average Client Time (s)	Warning (%)	Critical (%)	Transactions (/min)
Show	Clientbenutzer-IDs	0.884	36.734	52.984	0.619	0.002	0.058	0.02	4.995	27.32
Show	deploy_admin	2.104	2.453	11.484	1.458	0.153	0.002	N/P	N/P	0.199
Show	mary	2.051	36.734	52.984	1.643	0.003	0.120	0.062	15.361	8.884
Show		0.484	0	1.125	0.175	0	0	N/P	N/P	3.017
Show	paul	0.104	0	1.469	0.096		0.030	0	0	7.608

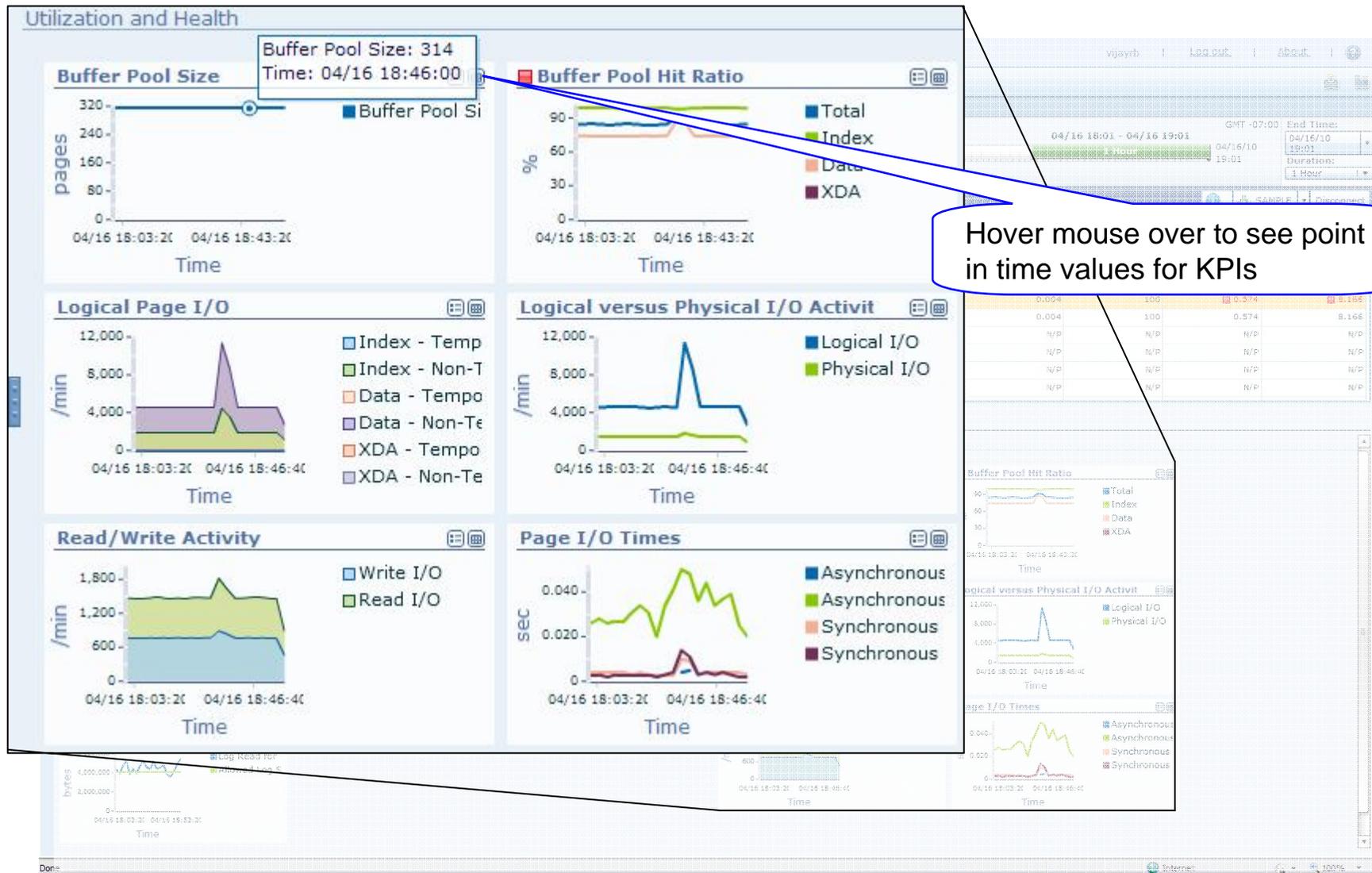
Graphically represent your trouble spot data



Drill down into end-to-end details

Show comparisons of the client, data server and network times to see where the bulk of the time is being spent

Hover KPI info



Easy collaboration

Optim Performance Manager V4.1
Database Connection Report

Report Information >>

Report Description
The Database Connection Report gives you an overview about the active database connections at a specific time. It can be used to identify not well performing applications, or applications causing problems in performance indicators such as lock-wait times, physical and logical reads and writes, or other connection statistics.
It is recommended to use the Optim Performance Manager Extended Insight feature instead of this report, but the report is provided as an alternative if the Extended Insight feature is not available or for users who are not using the Client.
The user can navigate back and forward between snapshots in the report to identify when problems occurred or to compare the activity of the database application between certain times. He can also click on a connection in an additional report.

Report Parameters

Connection: GSDB
Database: LOCALHOST_50002/GSDB
Partition: ALL
Time of data: Mar 8, 2010 2:31 PM ◀ ▶
Report Build: Mar 8, 2010 4:48 PM

Email alerting from OPM when a metric has been exceeded



Active connections

Application name	Application handle	Application ID	Partition	Application Status	Time (min)
OPMRepositoryServer	17,407	127.0.0.1.14096.100308222433	0	UOW waiting	0.7
db2strmm	17,411	*LOCAL.DB2.100308222455	0	connect completed	0.0
db2taskd	17,412	*LOCAL.DB2.100308222456	0	connect completed	0.0
db2wimd	17,413	*LOCAL.DB2.100308222457	0	connect completed	0.0
db2evmg_DB2DETAILDEA	17,414	*LOCAL.DB2.100308222458	0	connect completed	0.0
db2evmt_DB2_02@SVL@T	17,429	*LOCAL.DB2.100308222513	0	connect completed	0.0
db2evmt_@@@@@@@_DB	17,431	*LOCAL.DB2.100308222515	0	connect completed	0.0
db2jcc_application	17,480	127.0.0.1.33296.100308223121	0	UOW executing	6.4

```

From: Thuan Bui/Santa Teresa/IBM@IBMUS
To: Thuan Bui/Santa Teresa/IBM@IBMUS
Date: 04/02/2010 01:53 PM
Subject: IBM Optim Performance Manager.

IBM Optim Performance Manager.

DB2 threshold violation was detected at [Apr 2, 2010, 1:52:14 PM PDT] on DB2 system 'LOCALHOST_50001_GSDB'.

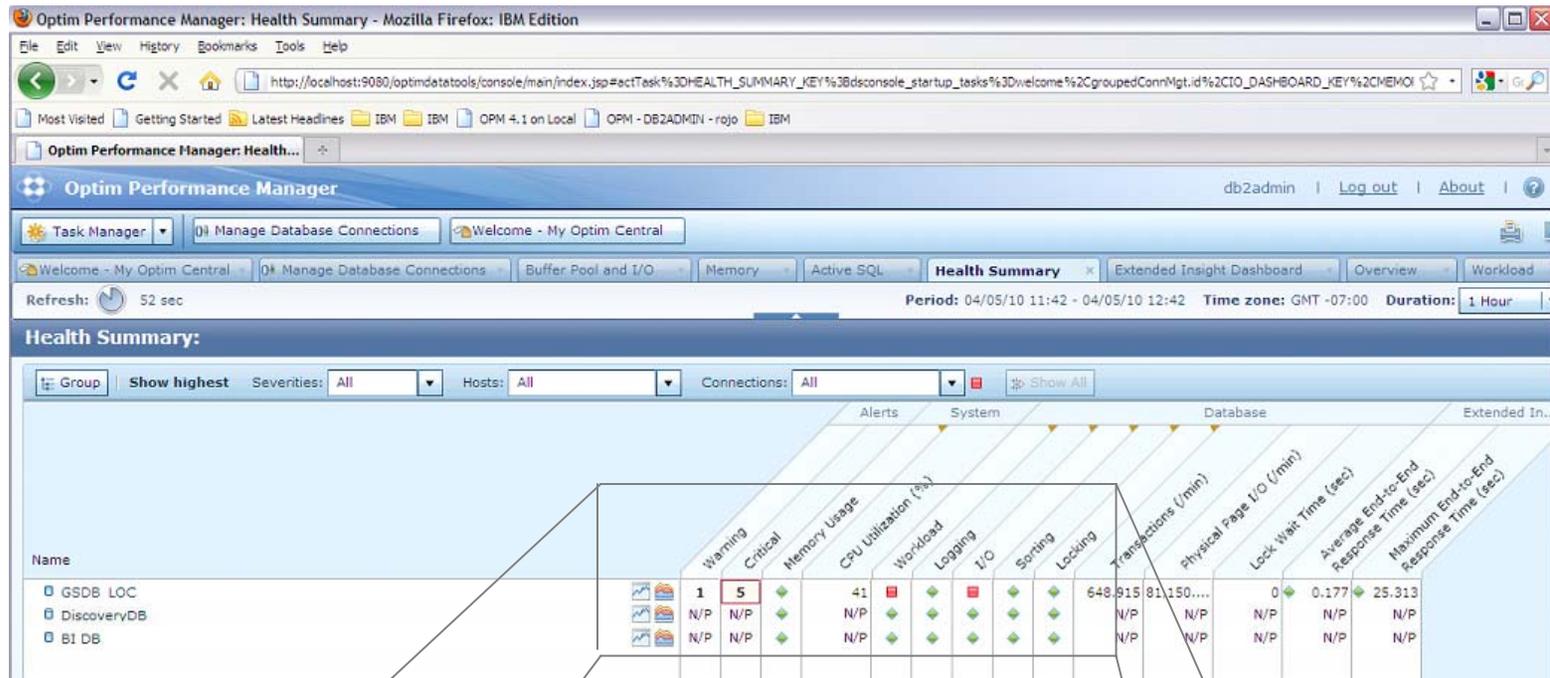
Details :

Counter : AVG_NUMBER_ROWS_SELECTED_ROW, Average number of rows read per selected row (rows)
Severity : PROBLEM
Owner : DB2ADMIN

Current value : 26.83
Warning level : > 5.0
Problem level : > 10.0

Violator :
Database name : GSDB
Database path : C:\DB2_01\NODE0000\SQL00002\
Total connections to database : 10
Database status : Database is active
Database connection time : 2010-04-01 18:39:10.253002
Partition name : PART0
Partition number : 0
    
```

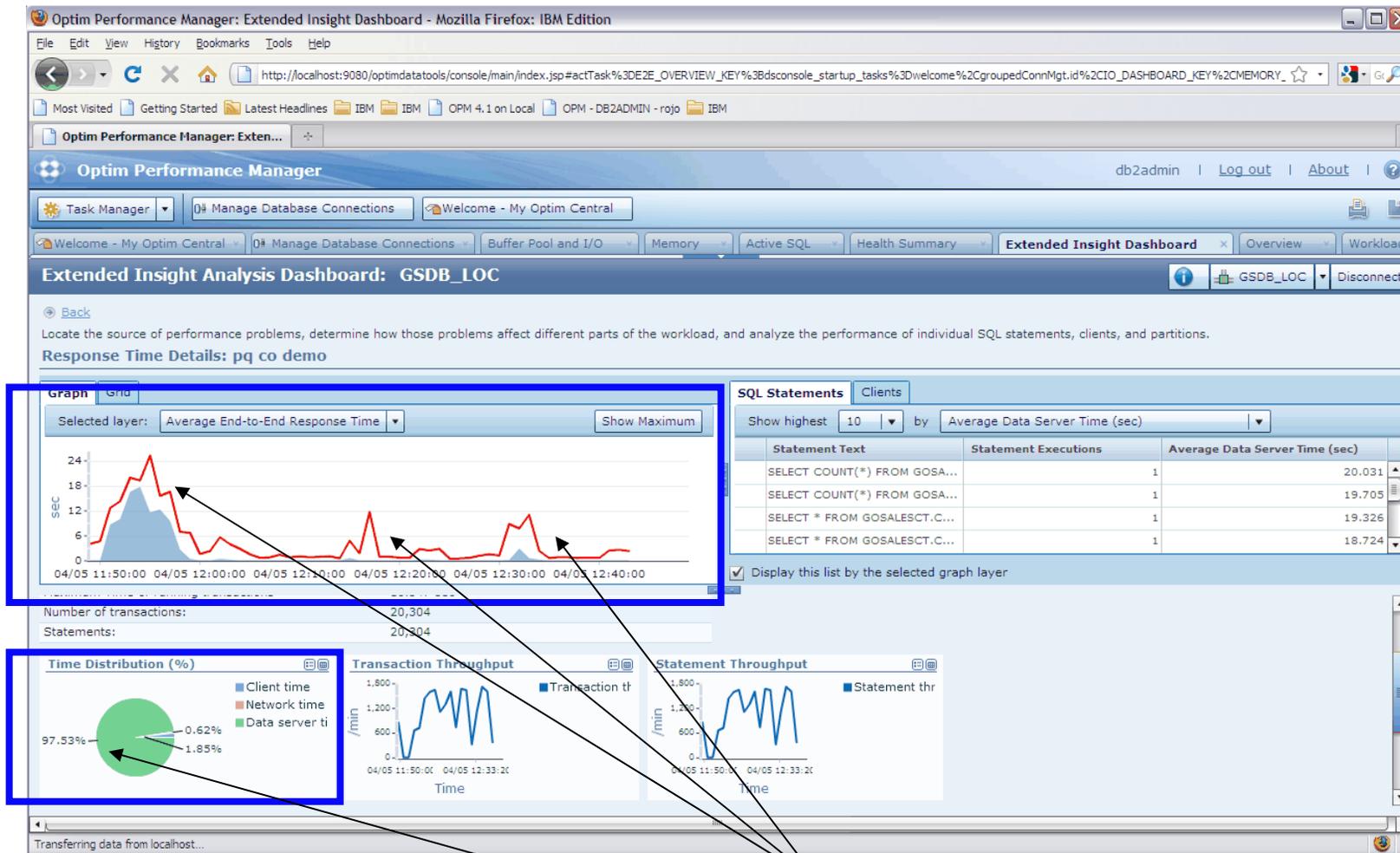
OPM quickly identifies an issue



OPM alerts on the glass and email



Get end-to-end application insight



Contributors but not causal to the slowdown
 Data Server Time Percentage very high

OPM Dashboards Provide Direction

The screenshot shows the IBM Optim Performance Manager Health Summary dashboard. The main table displays various performance metrics for different database components. Two red circles highlight specific alert icons in the 'Alerts' column. A blue arrow points from one of these circles to a detailed 'Alert Summary' popup window. Another arrow points from the 'Alert Summary' popup to the 'Workload Dashboard' link at the bottom of the popup.

Alerts with details

Switch to Workload Dashboard

Group	Name	Alerts	System	Database	Extended In...
GSDB LOC		1	5	24	
DiscoveryDB		N/P	N/P	N/P	N/P
BI DB		N/P	N/P	N/P	N/P

Severity	Alert Name	Last Alert Value	Alert Summary	End Time
Warning	Buffer Pool Hit Ratio	0.95 %	04/05 11:...	N/P
Critical	Rows Read per Fetched Row	77369.74	04/05 11:...	N/P
Critical	Buffer Pool Hit Ratio	9.83 %	04/05 11:...	04/05 11:...
Critical	Rows Read per Fetched Row	43332.03	04/05 11:...	04/05 11:...
Critical	Package Cache Hit Ratio	9.27 %	04/05 11:...	N/P

Severity	Alert Name	Last Alert Value	Start Time	End Time
Warning	Failing Transactions	9 %	04/05 11:...	04/05 11:...
Critical	Rows Read per Fetched Row	63156.08	04/05 11:...	N/P

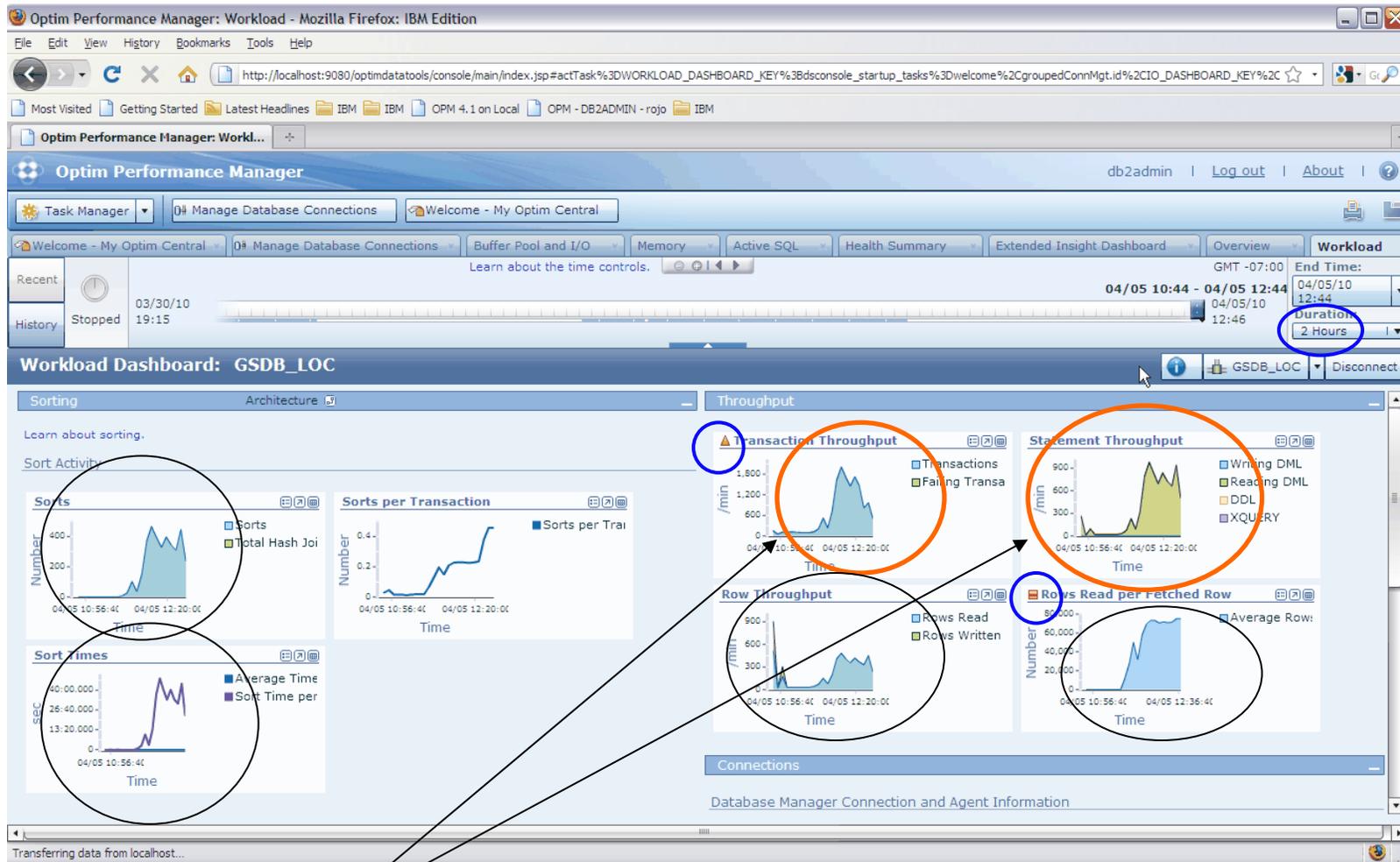
Severity: Warning
 Alert name: Failing Transactions
 Last alert value: 9 %
 Start time: 04/05/10 11:53:03
 End time: 04/05/10 11:53:58
 Partition/Member: 0
 Connection name: GSDB_LOC

Workload Dashboard | View Alert Detail

Analyze... | Configure... | Comment | Send | Delete

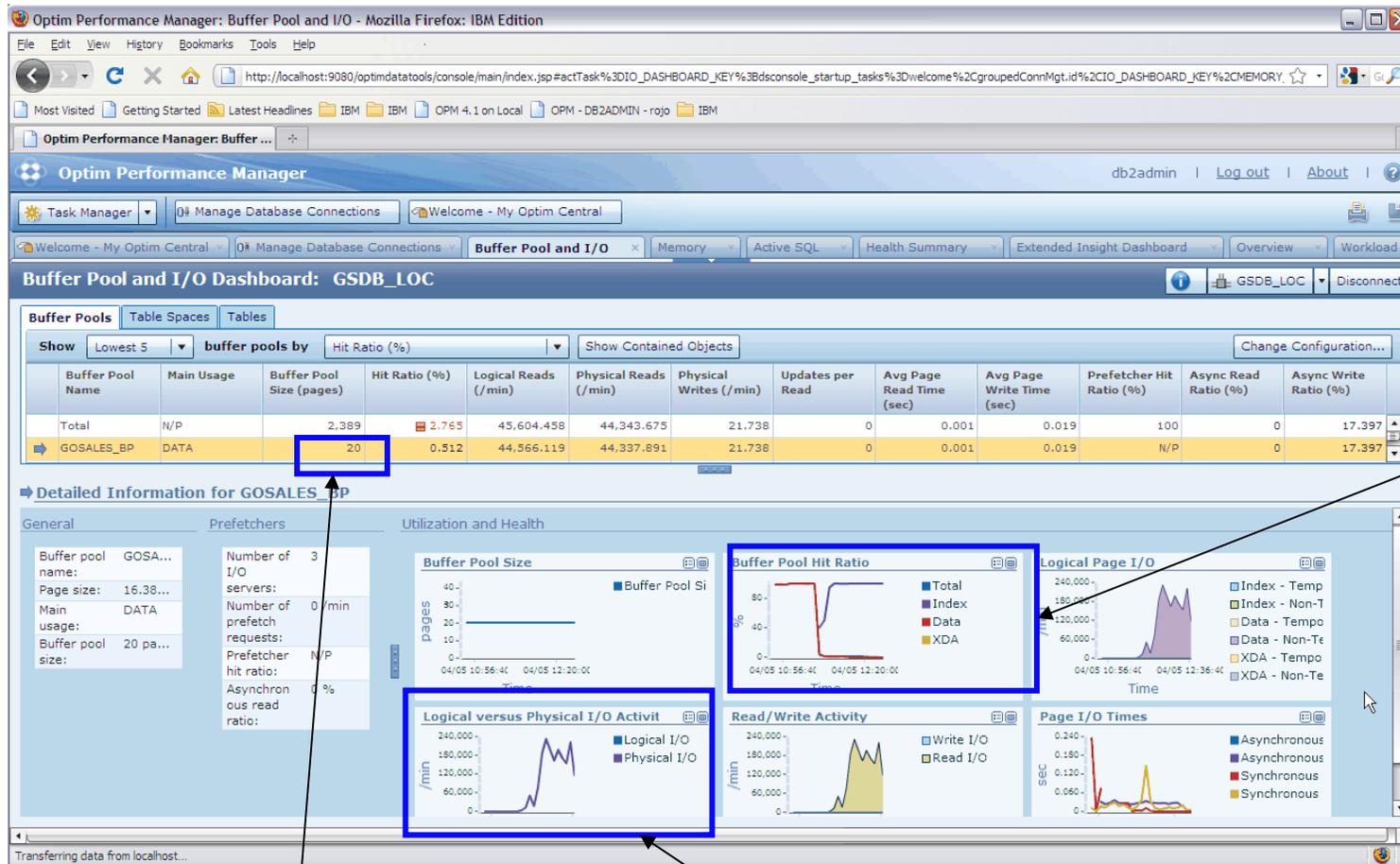
Close

Drilldown Workload To Diagnose Further



Number of transactions dramatically increased, causing KPIs to also increase

Deeper Diagnosis: Bufferpool and I/O Drilldowns

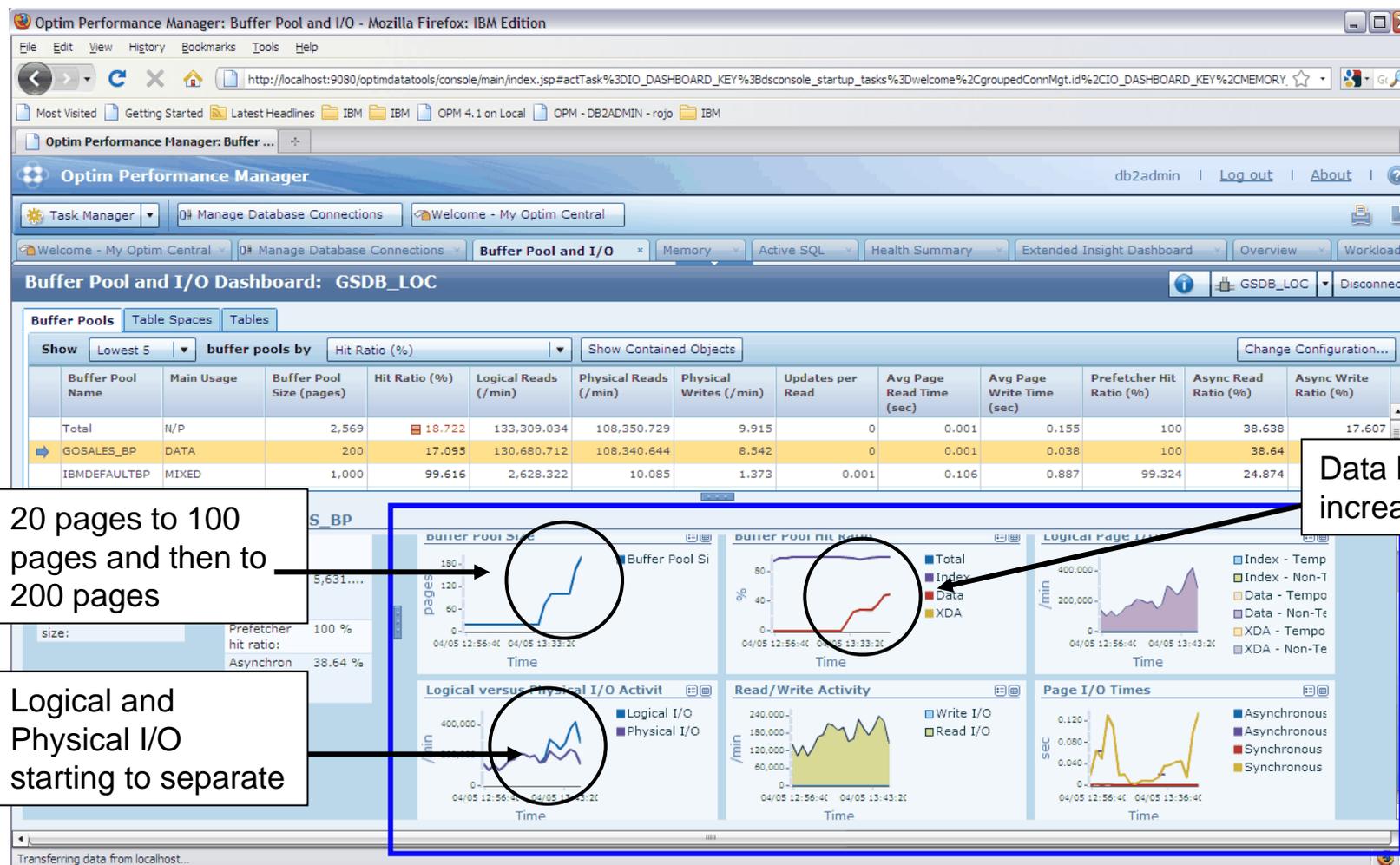


Hit ratios take a dive

Low buffer pool size

Logical and Physical I/O are 1:1 indicating every page read requires an I/O

Iterative changes



Performance Manager

Packaging

Feature	Data Studio Health Monitor (included in DB2)	Optim Performance Manager (included in DB2 AESE)	Optim Performance Manager Extended Edition	DB2 Performance Optimization Feature or AESE
Alerts and notifications	X	X	X	X
Overview health summary	X	X	X	X
Diagnostic dashboards		X	X	X
Standard reporting		X	X	X
OPM privileges, OQT integration		X	X	X
Extended Insight			X	
ITCAM, pureQuery integration			X	
DB2 WLM administration tooling				X
DB2 WLM feature				X

Summary

- **Monitoring in DB2 is changing rapidly**
 - Moving to time spent and time waiting metrics
 - Each release and fixpack typically adds more monitor elements you can leverage

- **Much of the support is targeted at helping tool vendors**
 - However, you can use SQL to get at the same info