

The DB2Night Show™ #136

www.DB2NightShow.com

Twitter.com/DB2NightShow #DB2Night

Educate – Inform – Entertain

“Why WAIT? DB2 LUW Locking!”

Your host:



Scott Hayes
President, DBI
IBM DB2 GOLD Consultant
& IM Champion

Our Special Guest:



Paul Turpin, @pault99
Sr. DBA
BB&T Bank

DISCLAIMERS, ETC

The DB2Night Show sponsors, advertisers, hosts, and guests, and their respective organizations, if any, are not responsible for any liability for any content provided herein. Opinions and DB2 Views are materialized in the viewer's mind only. Use information and ideas at your own risk. Use respect and courtesy when contributing to The DB2Night Show via Tweets, Email, or as a guest. Trademarks and Registered Trademarks belong to their respective organizations. The DB2Night Show is being recorded. Your participation may be recorded via your contributions, if any, and you consent to any and all content being recorded via your ongoing participation. Mind your manners. Recordings, or derivative works thereof, will be made available as replays. Replays, blogs, and related show materials © 2014 DBI. All Rights Reserved.

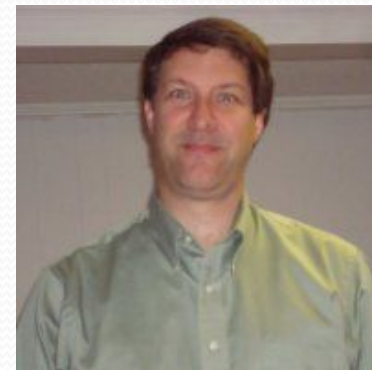
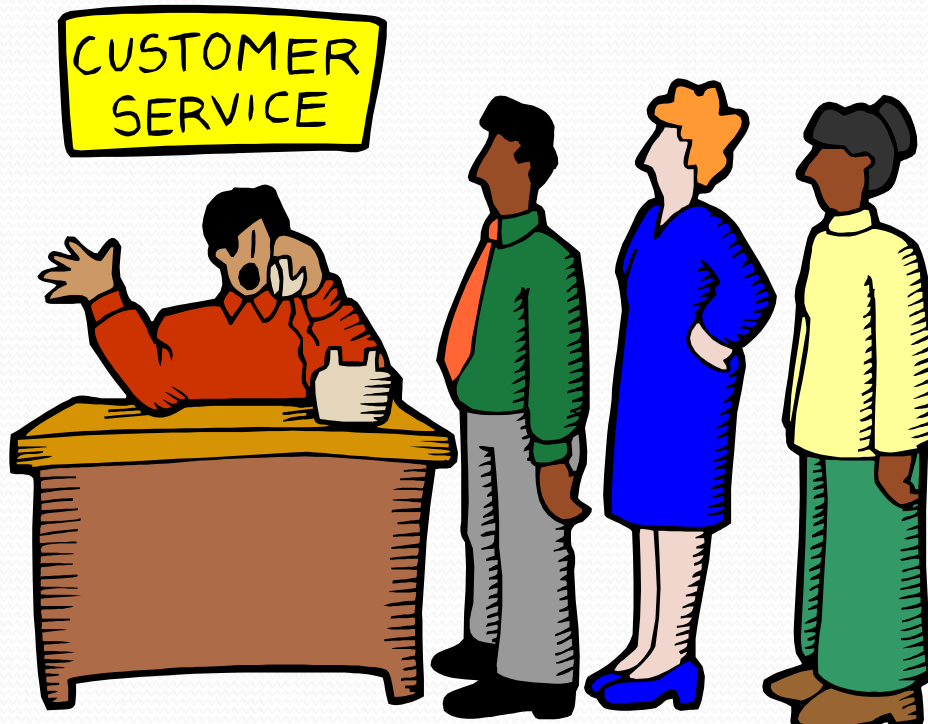
Upcoming Education...

- 19 June 3pm CDT – “How to Tune DB2 LUW in a Minute” by DBI Software
 - <http://bit.ly/dbievents>
- 20 JUNE – Leon Katsnelson, IBM, DB2 in the Cloud Update
- DB2 for z/OS shows:
 - 13 June – Big Data Disaster Recovery Performance w/ Dave Beulke
 - 27 June – Where’s the problem in My z/OS Transaction w/ Martin Hubel

Our Featured Topic!

Why Wait?

DB₂ LUW Locking



Paul Turpin, @pault99
Sr. DBA
BB&T Bank

What Are You Waiting For?

Paul Turpin
BB&T

Session Code: **C13**
Friday, May 3 10:30 AM – 11:30 | Platform: DB2 for Linux, UNIX, Windows



Often we suspect a query is waiting on something because it has a long execution time, but a small CPU time. Is it a lockwait, is it a long compilation time, is it waiting on I/O or something else? It is impossible to resolve the underlying condition until you understand what is wrong. I will show you how to determine what it was waiting on.

This presentation will give high level information on time-spent elements. A deep dive into lockwait analysis will be done using various diagnostic tools. Some tools that will be covered are MONREPORT, MON_GET_PKG_CACHE_STMT, MON_LOCKWAITS and the locking event monitor (and queries to read that data).

Usage of several configuration parameters including mon_lockwait, mon_lw_thresh, mon_deadlock, mon_locktimeout, locktimeout and maxlocks will be discussed.

Why Do We Care About Waits?

- “Time is Money”
- Waits can cause queries to run much longer than expected

How Do You Know Waits Are Bad?

- Extreme example

Total execution time (sec.microsec) = 11720.504066

Total user cpu time (sec.microsec) = 0.174954

Total system cpu time (sec.microsec) = 0.017376

What you will learn

- High level view of time-spent elements
- Configuration parameters
 - maxlocks
 - mon_lockwait
 - mon_lw_thresh
 - mon_deadlock
 - mon_locktimeout
 - locktimeout

High level view of time-spent elements

Tools for looking at wait times using MONREPORT,
MON_GET_PKG_CACHE_STMT

Lockwait analysis using MON_LOCKWAITS and other tools

Locking event monitor usage and locking table queries

Usage of configuration parameters including mon_lockwait, mon_lw_thresh,
mon_deadlock, mon_locktimeout, locktimeout and maxlocks

What you will learn

- **Tools for looking at wait times including MONREPORT, MON_GET_PKG_CACHE_STMT**
- **Lockwait analysis using MON_LOCKWAITS and other tools**
- **Locking event monitor usage and locking table queries**

Sharing Is Good

- **Feel free to tweet or blog anything you find interesting in this session**
- **#IDUGNA is the hashtag for the conference**



Information Resources

- **Information Center for DB2 10.1**
- **Wait time reporting came in 9.7**

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r1/index.jsp>

Time-spent elements

- Details found at:
- <http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0056890.html>
- Database manager monitors time spent:
 - Wait times
 - Component processing times
 - Component elapsed times

10.1

Logical data groups overview

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0059128.html>

9.7

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0056890.html>

Hierarchy of elements

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0055434.html>

total_act_wait_time

- **Total time spent waiting within the database server**
- **The value is given in milliseconds**
- **Reported by many monitoring functions, including MON_GET_PKG_CACHE_STMT and MON_GET_ACTIVITY_DETAILS**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0054078.html>

total_act_time

- Total time spent executing activities
- The value is given in milliseconds
- Reported by many monitoring functions, including `MON_GET_PKG_CACHE_STMT` and `MON_GET_ACTIVITY_DETAILS`

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0054077.html>

High Level Summary of Wait Times

```
select RQST_WAIT_TIME_PERCENT,  
ACT_WAIT_TIME_PERCENT, IO_WAIT_TIME_PERCENT,  
LOCK_WAIT_TIME_PERCENT,  
AGENT_WAIT_TIME_PERCENT,  
NETWORK_WAIT_TIME_PERCENT,  
COMPILE_PROC_TIME_PERCENT,  
CF_WAIT_TIME_PERCENT,  
RECLAIM_WAIT_TIME_PERCENT, ← pureScale  
SPACEMAPPAGE_RECLAIM_WAIT_TIME_PERCENT  
from SYSIBMADM.MON_DB_SUMMARY
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0056513.html>

Configuration Parameters

- locklist
- maxlocks
- mon_lockwait
- mon_lw_thresh
- mon_deadlock
- mon_locktimeout
- locktimeout
- mon_lck_msg_lvl

locklist

- Amount of storage allocated to locklist
- Default is **AUTOMATIC**
- Range 4 – 134217728 (4 KB pages)

- This is related to lockwaits because lock escalation on a table is very likely to cause lockwaits

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000267.html>

maxlocks

- **Maximum percent of lock list before escalation configuration parameter**
- **Percentage (1 – 100)**
- **Default is AUTOMATIC**
- **This is related to lockwaits because lock escalation on a table is very likely to cause lockwaits**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000268.html>

locktimeout

- Lock timeout configuration parameter
- The number of seconds that an application will wait to obtain a lock
- In Seconds with a range of [-1; 0 – 32,767]
- Default is -1
- If the lock is not obtained within the time allowed, a **-911** error will occur

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000329.html>

locktimeout

- **Special cases**

0 If no lock is immediately available the application immediately receives a **-911** error

-1 Lock timeout detection is turned off and the lock will be waited for until

- **The lock is granted**
- **A deadlock occurs**

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000329.html>

CURRENT LOCK TIMEOUT

- For table, row, data partition, and multidimensional clustering (MDC) block locks, an application can override the locktimeout value by changing CURRENT LOCK TIMEOUT special register. Can be used in 9.7, also.
- SET CURRENT LOCK TIMEOUT **120**
- Unset (use value from database configuration parameter)
- SET CURRENT LOCK TIMEOUT **NULL**

CURRENT LOCK TIMEOUT

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.ref.doc/doc/r0011857.html>

SET CURRENT LOCK TIMEOUT

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.ref.doc/doc/r0011874.html>

Special Registers (about 35 in 10.1)

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.ref.doc/doc/r0008404.html>

mon_lockwait

- Controls generation of lock wait events for lock event monitor
- Minimum level of collection
- Settings:
 - NONE
 - WITHOUT_HIST
 - HISTORY
 - HIST_AND_VALUES

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054941.html>

mon_lockwait

- **NONE** – no lock timeout events will be generated **UNLESS** lock timeout event collection enabled on DB2 Workload objects with **COLLECT LOCK TIMEOUT DATA** clause
- **WITHOUT_HIST** – lock events data sent to active locking event monitor. **Does NOT** include history of activities executed by application waiting on lock.

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054939.html>

Creating and altering workloads is a big topic. Some information can be found at:

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.ref.doc/doc/r0050554.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.ref.doc/doc/r0050555.html>

mon_lockwait

- **HISTORY** – Does include history of activities executed by application waiting on lock, until maximum size is reached. Default max size is 250, but can be changed with the **DB2_MAX_INACT_STMTS** registry variable.
- **HIST_AND_VALUES** – Collects history and input data values for the activities that have them.

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054939.html>

For HIST_AND_VALUES data values will NOT have LOB, LONG VARCHAR, LONG VARGRAPHIC or XML data.

DB2_MAX_INACT_STMTS

- **Default 250**
- **Maximum value is 4,000,000,000**
- **Uses system monitor heap**
- **Can exhaust heap if the application contains a large number of statements in a unit of work or if there are a large number of concurrent application**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.regvars.doc/doc/r0005665.html>

Overrides number of inactive statements kept by one application

mon_lw_thresh

- Used in conjunction with `mon_lockwait`
- Measured in microseconds
- Default - 5000000 microseconds (5 seconds)
- If database upgraded from 9.7 to 10.1 default will be 4294967295 (much larger than 10.1 default)
- `num_lw_thresh_exceeded` reports number of times `mon_lw_thresh` was exceeded

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054942.html>

Microsecond to second conversion

<https://www.google.com/#q=microsecond+to+second>

`num_lw_thresh_exceeded`

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0056500.html>

In several of the monitor functions including
`MON_GET_PKG_CACHE_STMT`, `MON_GET_UNIT_OF_WORK` and
`MON_GET_CONNECTION`

mon_deadlock

- Controls generation of deadlock events for lock event monitor
- Similar to mon_lockwait and mon_locktimeout
- Settings:
 - NONE
 - WITHOUT_HIST
 - HISTORY
 - HIST_AND_VALUES

mon_locktimeout

- Controls the generation of lock timeout events for lock event monitor
- Similar to mon_lockwait and mon_deadlock

- Minimum level of collection
- Settings:
 - NONE
 - WITHOUT_HIST
 - HISTORY
 - HIST_AND_VALUES

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054939.html>

mon_lck_msg_lvl

- Enables lock timeout messages to be written to Administration Notification file
- Enable by setting mon_lck_msg_lvl to 3

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.regvars.doc/doc/r0005665.html>

Overrides number of inactive statements kept by one application

mon_lck_msg_lvl

- **Levels** **Default is 1**
- **0** **No notification of lock escalations, deadlocks, time outs**
- **1** **Notification of lock escalations**
- **2** **Notification of lock escalations and deadlocks**
- **3** **Notification of lock escalation, deadlocks and lock timeouts**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.regvars.doc/doc/r0005665.html>

Overrides number of inactive statements kept by one application

lock_escals_maxlocks (monitor element)

- **Number of lock escalations reaching the limit set by maxlocks database configuration parameter.**
- **Reports escalations caused for local lock memory usage**
- **Reported by many monitoring functions, including MON_GET_PKG_CACHE_STMT and MON_GET_ACTIVITY_DETAILS**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0056236>

Tools

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0056236>

MONREPORT

- Set of procedures for retrieving and reporting monitoring data
- The ones most helpful for **wait** information are:
 - **DBSUMMARY** (database summary only)
 - **PKG_CACHE** (SQL level)
 - **LOCKWAIT** (details of a lock holder / requestor)

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

MONREPORT.DBSUMMARY

- Summarized waiting data – good for a high level view of wait times.
- For requests (wait time / total time)
- For activities (wait time / total time)
- Detailed breakdown of TOTAL_WAIT_TIME

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

MONREPORT.DBSUMMARY

	⌘	Total
	---	-----
TOTAL_WAIT_TIME	100	26
I/O wait time		
POOL_READ_TIME	0	0
POOL_WRITE_TIME	0	0
DIRECT_READ_TIME	100	26
DIRECT_WRITE_TIME	0	0
LOG_DISK_WAIT_TIME	0	0
LOCK_WAIT_TIME	0	0
AGENT_WAIT_TIME	0	0
Network and FCM		
TCPIP_SEND_WAIT_TIME	0	0
TCPIP_RECV_WAIT_TIME	0	0
IPC_SEND_WAIT_TIME	0	0
IPC_RECV_WAIT_TIME	0	0
FCM_SEND_WAIT_TIME	0	0
FCM_RECV_WAIT_TIME	0	0
WLM_QUEUE_TIME_TOTAL	0	0
CF_WAIT_TIME	0	0
RECLAIM_WAIT_TIME	0	0
SMP_RECLAIM_WAIT_TIME	0	0

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

MONREPORT.PKGCACHE

- To run the report:

```
db2 "call monreport.pkgcache "
```

or

```
db2 "call MONREPORT.pkgcache"
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056372.html>

MONREPORT.PKGCACHE

Summaries by 'top' metrics

Top 10 statements by TOTAL_CPU_TIME

Top 10 statements by TOTAL_CPU TIME per exec

Top 10 statements by TOTAL_ACT_WAIT_TIME

Top 10 statements by TOTAL_ACT_WAIT_TIME per exec

Top 10 statements by ROWS_READ + ROWS_MODIFIED

Top 10 statements by ROWS_READ + ROWS_MODIFIED per exec

Top 10 statements by number of executions

Top 10 statements by I/O wait time

Top 10 statements by I/O wait time per exec

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056372.html>

MONREPORT.PKGCACHE

Top 10 statements by TOTAL_ACT_WAIT_TIME

#	TOTAL_ACT_WAIT_TIME	LOCK_WAIT_TIME	STMT_TEXT
11	17	0	SELECT ARRAY_AGG(P.EXECUTABLE_ID ORDER BY M.CPU TI
12	14	0	CALL SAVE_EXEC_INFO (CAST(:HV00040 :HI00040 AS "
3	0	0	SELECT ARRAY_AGG(VALUE) INTO :HV00029 :HI00029 F

Top 10 statements by TOTAL_ACT_WAIT_TIME per exec

#	TOTAL_ACT_WAIT_TIME	LOCK_WAIT_TIME	STMT_TEXT
12	1	0	CALL SAVE_EXEC_INFO (CAST(:HV00040 :HI00040 AS "
3	0	0	SELECT ARRAY_AGG(VALUE) INTO :HV00029 :HI00029 F
16	0	0	VALUES (1, CARDINALITY(CAST(:HV00040 :HI00040 AS
15	0	0	SET :HV00017 :HI00017 = RPAD(VARCHAR(:HV00035 :
22	0	0	VALUES (:HV00026 :HV00026 + 1 :HV00024 :HI0002

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056368.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.apdv.sqlpl.doc/doc/r0056372.html>

MONREPORT.LOCKWAIT

- Reports shows detailed lockwait information for lock holder and lock requestor
- Report will **ONLY** contain data if there are lockwaits when it runs (no historical data)

db2 "call monreport.lockwait "

MON_GET_PKG_CACHE_STMT

```
SELECT * FROM  
TABLE(MON_GET_PKG_CACHE_STMT(null,null,null,-2));
```

Parameters are **optional**

1st is section type of "D" or "S" (dynamic or static). You get both by default (specify NULL)

2nd is executable id

3rd search_args, specify a modified time

4th member (in the database) [-1 is current member, -2 is all members]

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

MON_GET_PKG_CACHE_STMT

- Common parts of query

```
SELECT SUBSTR(STMT_TEXT,1,500) AS STMT,
```

WAIT COLUMNS

```
FROM TABLE(MON_GET_PKG_CACHE_STMT('',NULL,NULL,-2))
```

```
WHERE TOTAL_ACT_WAIT_TIME > 0
```

```
order by TOTAL_ACT_WAIT_TIME DESC fetch first 10 rows only;
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

MON_GET_PKG_CACHE_STMT

- Lots of data elements – but focusing on the WAIT elements

```
SELECT SUBSTR(STMT_TEXT,1,500) AS STMT,  
SECTION_TYPE AS TYPE, NUM_EXECUTIONS,  
TOTAL_ACT_TIME AS TOTAL_ACTIVITY_TIME,  
TOTAL_ACT_WAIT_TIME AS TOTAL_ACTIVITY_WAIT_TIME,  
LOCK_WAIT_TIME, LOCK_WAITS,  
FROM TABLE(MON_GET_PKG_CACHE_STMT('',NULL,NULL,-2))  
WHERE TOTAL_ACT_WAIT_TIME > 0  
order by TOTAL_ACT_WAIT_TIME DESC fetch first 10 rows only;
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

MON_GET_PKG_CACHE_STMT

```
SELECT SUBSTR(STMT_TEXT,1,500) AS STMT,  
  
LOG_BUFFER_WAIT_TIME, LOG_DISK_WAIT_TIME,  
LOG_DISK_WAITS_TOTAL, FCM_RECV_WAIT_TIME,  
FCM_SEND_WAIT_TIME,  
FCM_MESSAGE_RECV_WAIT_TIME,  
FCM_MESSAGE_SEND_WAIT_TIME,  
FCM_TQ_RECV_WAIT_TIME, FCM_TQ_SEND_WAIT_TIME,  
LOCK_WAITS_GLOBAL, RECLAIM_WAIT_TIME
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

MON_GET_PKG_CACHE_STMT

```
SELECT SUBSTR(STMT_TEXT,1,500) AS STMT,
```

```
SPACEMAPPAGE_RECLAIM_WAIT_TIME,
```

```
CF_WAIT_TIME
```

```
AUDIT_FILE_WRITE_WAIT_TIME,
```

```
AUDIT_SUBSYSTEM_WAIT_TIME
```

```
DIAGLOG_WRITE_WAIT_TIME
```

```
EVMON_WAIT_TIME,
```

```
TOTAL_EXTENDED_LATCH_WAIT_TIME,
```

```
PREFETCH_WAIT_TIME
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

MON_LOCKWAITS

Shows ONLY current lockwait information (just like MONREPORT), so it is helpful if the lockwait stays for some length of time.

Shows requestor and holder information

Replaces the SNAPLOCKWAIT (deprecated) admin view

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0056601.html>

MON_LOCKWAITS

```
Select LOCK_NAME, LOCK_OBJECT_TYPE, LOCK_WAIT_ELAPSED_TIME,  
SUBSTR(TABSCHEMA,1,10) as TABSCHEMA, SUBSTR(TABNAME,1,30) as  
TABNAME,  
DATA_PARTITION_ID,  
LOCK_MODE, LOCK_CURRENT_MODE, LOCK_MODE_REQUESTED, REQ_MEMBER,  
SUBSTR(REQ_APPLICATION_NAME,1,30) as REQ_APPLICATION_NAME,  
SUBSTR(REQ_USERID,1,30) as REQ_USERID,  
SUBSTR(REQ_STMT_TEXT,1,80) as REQ_STMT_TEXT,  
HLD_APPLICATION_HANDLE, HLD_MEMBER,  
SUBSTR(HLD_APPLICATION_NAME,1,30) as HLD_APPLICATION_NAME,  
SUBSTR(HLD_USERID,1,30) as HLD_USERID,  
SUBSTR(HLD_CURRENT_STMT_TEXT,1,80) as HLD_CURRENT_STMT_TEXT,  
REQ_APPLICATION_HANDLE, REQ_AGENT_TID  
from SYSIBMADM.MON_LOCKWAITS;
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0056601.html>

Lock Event Monitors

- Captures locking information
- Can collect lock timeout, lock wait, and deadlock information
- Data can be collected in unformatted event monitors (data has to be formatted later) or in DB2 tables

Data generated by locking event monitors

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0059025.html>

Collecting lock event data and generating reports

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

Lock Event Monitors - Creation

- **Two types**
- **Unformatted – format collected data later**
 - **Low overhead**
- **Write to table – locking data written to DB2 tables immediately**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

Lock Event Monitors - Creation

CREATE EVENT MONITOR **lockevtmon** FOR LOCKING
WRITE TO **UNFORMATTED** EVENT TABLE

Activate the event monitor

SET EVENT MONITOR **lockevtmon** STATE 1

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

Create event monitor DDL with db2evtbl

```
db2evtbl -schema test -evm eventmon1 locking
```

This will generate the create event monitor DDL that includes the five locking tables:

- CONTROL
- LOCK
- LOCK_PARTICIPANTS
- LOCK_PARTICIPANT_ACTIVITIES
- LOCK_ACTIVITY_VALUES

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.cmd.doc/doc/r0008959.html>

Lock Event Monitors - Creation

```
CREATE EVENT MONITOR eventmon1  
FOR LOCKING WRITE TO TABLE  
CONTROL (TABLE test.CONTROL_eventmon1),  
LOCK (TABLE test.LOCK_eventmon1),  
LOCK_PARTICIPANTS (TABLE  
test.LOCK_PARTICIPANTS_eventmon1),  
LOCK_PARTICIPANT_ACTIVITIES (TABLE  
test.LOCK_PARTICIPANT_ACTIVITIES_eventmon1),  
LOCK_ACTIVITY_VALUES (TABLE  
test.LOCK_ACTIVITY_VALUES_eventmon1);
```

Lock Event Monitors - Thresholds

ALTERS of Workloads another method to do this

Set threshold for LOCKWAITS

update db cfg for test using **mon_lockwait** hist_and_values

update db cfg for test using **mon_lw_thresh** 30000000 **30 secs**

Set what to collect for LOCK TIMEOUTS

update db cfg for test using **mon_locktimeout** hist_and_values

mon_lockwait

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054941.html>

mon_lw_thresh

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054942.html>

mon_locktimeout

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.config.doc/doc/r0054939.html>

Lock Event Monitors - Thresholds

ALTERS of Workloads another method to do this

Set what to collect for DEADLOCKS

**update db cfg for sample using `mon_deadlock`
`hist_and_values`**

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

Lock Event Monitors - Collecting

Rerun the workload that is having trouble

Format the event locking report using

java program `db2evmonfmt`

Or

`EVMON_FORMAT_UE_TO_XML` table function to create XML

`EVMON_FORMAT_UE_TO_TABLES` procedure put data into table

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/t0055093.html>

Set up db2evmonfmt

- These give information for how to get db2evmonfmt set up:

<http://db2forum.wordpress.com/2011/10/17/lock-event-monitor-in-db2-9-7-to-solve-concurrency-issues/>

<http://www.ibm.com/developerworks/data/library/techarticle/dm-1004lockeventmonitor/>

<http://db2commerce.com/2012/01/23/analyzing-deadlocks-the-new-way/>

Using db2evmonfmt

To obtain a formatted text output for all events of type **LOCKTIMEOUT** that have occurred in last **12** hours from unformatted event table **LOCKEVTMON** in database **TESTDB**

```
java db2evmonfmt -d TESTDB -ue LOCKEVTMON -ftext -hours 12 -  
type locktimeout
```

Produces an easy to read report

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0053977.html>

Using `EVMON_FORMAT_UE_TO_XML`

Convert unformatted events to XML

Selects ALL data from **LOCKEVTMON**

```
SELECT evtmon.* FROM TABLE (  
  EVMON_FORMAT_UE_TO_XML ( NULL, FOR EACH ROW  
  OF ( select * from LOCKEVTMON order by  
  EVENT_TIMESTAMP ))) AS evtmon;
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0054909.html>

Using EVMON_FORMAT_UE_TO_XML

Selects LOCKWAIT data from **LOCKEVTMON** for the last 5 hours

```
SELECT evtmon.* FROM TABLE (  
  EVMON_FORMAT_UE_TO_XML ( NULL, FOR EACH ROW  
  OF ( select * from LOCKEVTMON order by  
  EVENT_TIMESTAMP where EVENT_TYPE = 'LOCKWAIT'  
  and EVENT_TIMESTAMP >= CURRENT_TIMESTAMP - 5  
  hours ))) AS evtmon;
```

EVENT_TYPE can be LOCKTIMEOUT, LOCKWAIT or DEADLOCK

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0054909.html>

Using EVMON_FORMAT_UE_TO_TABLES

Move unformatted events data to DB2 table

```
CALL EVMON_FORMAT_UE_TO_TABLES (evmon_type, xsrschema,  
xsobjectname, xmlschemafile, tabschema, tbsp_name, options,  
commit_count, fullselect)
```

```
CALL EVMON_FORMAT_UE_TO_TABLES ('LOCKING', NULL, NULL,  
NULL, NULL, NULL, 'RECREATE_ONERROR', -1, 'SELECT * FROM  
LOCKEVTMON ORDER BY event_timestamp')
```

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0054910.html>

evmon_type – Type of data stored in unformatted event table. The possible values are:

LOCKING - Data stored in the unformatted event table is from a locking event monitor.

PKG_CACHE - Data stored in the unformatted event table is from PACKAGE CACHE event monitor.

UOW - Data stored in the unformatted event table is from UOW event monitor.

Locking Table Queries

<u>Lock Mode Number</u>	<u>Lock Mode Type</u>
0	None
1	Intent Share (IS)
2	Intent Exclusive (IX)
3	Share (S)
4	Share with Intent Exclusive (SIX)
5	Exclusive (X)
6	Intent None (IN)
7	Super Exclusive (Z)
8	Update (U)
9	Scan Share (NS)
10	Next Key Weak Exclusive Lock

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0001286.html>

10 is

Locking Table Queries

Status

2	Waiting
4	Converting

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/r0001286.html>

Locking Table Queries

```
select * from LOCKEVENTS fetch first 5 rows only
```

```
select * from LOCK_ACTIVITY_VALUES fetch first 5 rows  
only
```

```
select * from LOCK_EVENT fetch first 5 rows only
```

```
select * from LOCK_PARTICIPANTS fetch first 5 rows only
```

```
select * from LOCK_PARTICIPANT_ACTIVITIES fetch first 5  
rows only
```

Locking Table Queries

- LOCK_PARTICIPANT_ACTIVITIES

EXPORT TO

```
'/db2data/exports/lock_participant_activities_March9.csv' of del
select STMT_FIRST_USE_TIME,STMT_LAST_USE_TIME,
package_name, effective_isolation, stmt_type, rtrim(stmt_text)
from test.LOCK_PARTICIPANT_ACTIVITIES
order by STMT_LAST_USE_TIME desc;
```

Locking Table Queries

- LOCK_PARTICIPANTS

```
EXPORT TO '/db2data/exports/lock_participants_March9.csv' OF DEL  
SELECT *  
FROM test.LOCK_PARTICIPANTS;
```


Locking Table Queries

```
select e.EVENT_ID, e.DEADLOCK_TYPE , SUBSTR(e.EVENT_TYPE,1,12) as
event_type
,p.LOCK_MODE_REQUESTED
,substr(p.TABLE_NAME,1,30) as TABLE_NAME
, CASE
    WHEN LOCK_MODE = 0 THEN 'None'
    WHEN LOCK_MODE = 1 THEN 'Intent Share'
END as LOCK_MODE
, substr(a.STMT_TEXT,1,80)
from LOCK_EVENT e, LOCK_PARTICIPANTS p, LOCK_PARTICIPANT_ACTIVITIES a
where e.EVENT_ID = p.EVENT_ID and e.EVENT_ID = a.EVENT_ID
and p.TABLE_NAME ^= ''
```

Locking Table Queries

```
, CASE  
  WHEN LOCK_MODE = 0 THEN 'None'  
  WHEN LOCK_MODE = 1 THEN 'Intent Share'  
  WHEN LOCK_MODE = 2 THEN 'Intent Exclusive'  
  WHEN LOCK_MODE = 3 THEN 'Share'  
  WHEN LOCK_MODE = 4 THEN 'Share with Intent Exclusive'  
  WHEN LOCK_MODE = 5 THEN 'Exclusive'  
  WHEN LOCK_MODE = 6 THEN 'Intent None'  
  WHEN LOCK_MODE = 7 THEN 'Super Exclusive'  
  WHEN LOCK_MODE = 8 THEN 'Update'  
  WHEN LOCK_MODE = 9 THEN 'Scan Share'  
  WHEN LOCK_MODE = 10 THEN 'Next Key Weak Exclusive'  
END as LOCK_MODE
```

Could use a similar case statement for LOCK_MODE_REQUESTED column in the LOCK_PARTICIPANTS table

What We Covered

- High level view of time-spent elements
- Configuration parameters
 - maxlocks
 - mon_lockwait
 - mon_lw_thresh
 - mon_deadlock
 - mon_locktimeout
 - locktimeout

What We Covered

- **Tools for looking at wait times using MONREPORT, MON_GET_PKG_CACHE_STMT**
- **Lockwait analysis using MON_LOCKWAITS and other tools**
- **Locking event monitor usage and locking table queries**

Additional Information

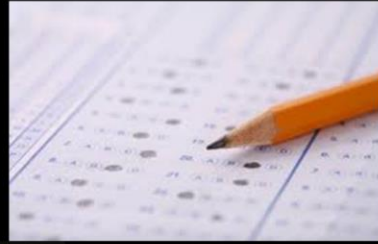
- **Appendix of this presentation contains a number of items that could not be covered**
- **Appendix has links to a number of other presentations that have more information**

THANK YOU!

Paul Turpin

BB&T

pturpin@carolina.rr.com



C13

What Are you Waiting For?

*Please fill out your session
evaluation before leaving!*



Paul Turpin is a consulting database administrator for a large financial institution. He specializes in DB2 for LUW on large systems. He enjoys exploring new features and functionality in DB2. He has spoken at IDUG North America, IDUG Europe, SHARE, IBM's Information on Demand conference, and several RUGs. Paul currently serves on the IDUG Board of Directors. He was one of IBM's inaugural Information Champions.

Appendix

- **Additional information resources**
- **db2fopt information**
- **Information for researching**
 - **Lock waits**
 - **Lock timeouts**
 - **Deadlocks**
 - **Lock escalation**

Additional Information

- Alexander Kopac presented
- C08 "Secrets of the Dead - Locks, Waits, Escalations & Semaphores" earlier in the week
- Where my request spending time? --- First taste DB2 9.7 Time Spent metrics (C04) Pak Kwan May 11, 2010
- <http://www.idug.org/p/do/sd/topic=103&sid=1539>

Additional Information

- **C15: Diagnosing Your Key Performance Bottlenecks in DB2 for LUW** David Kalmuk
- <http://www.idug.org/p/do/sd/topic=299&sid=3242>
- **Eliminating Performance Bottlenecks in DB2 9.7 and PureScale (D02)** Steve Rees
- <http://www.idug.org/p/do/sd/topic=101&sid=421>

Radesh Kumar

<http://radheshk.blogspot.com/2010/03/where-db-request-spends-time-in-db2-97.html>

<http://www.db2guide.blogspot.com/>

Additional Information

- Radesh Kumar blog
- <http://radheshk.blogspot.com/2010/03/where-db-request-spends-time-in-db2-97.html>

<http://www.db2guide.blogspot.com/>

db2fopt

- Specify query optimizer parameters command
- Can be used when setting up test system which has less physical resources than production system
- “Tricks” optimizer
- Example of use:
- `db2fopt SAMPLE update opt_sorheap 50000 opt_locklist 10000`

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.cmd.doc/doc/r0057067>

Lock Wait Diagnosis

- Need lock wait baseline to understand if situation is abnormal
- Can use the following tools to gather diagnostic information about active agents waiting on locks
 - application snapshot
 - lock administrative views (SNAPLOCK administrative view and SNAP_GET_LOCK table function)
 - lock wait option of db2pd -wlocks

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055234.html>

SNAPLOCK administrative view and SNAP_GET_LOCK table function

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0021992.html>

Lock Wait Diagnosis

- **For ideas on resolving lock waits**
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055073.html>
- **Information about lock monitoring**
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.mon.doc/doc/c0054136.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055234.html>

Lock Timeout Diagnosis

- Best to capture the locking data over time
- Create locking event monitor to capture lock timeout data for a workload or database
- Set `mon_lck_msg_lvl` database configuration parameter to 3 to write lock timeout messages to Administration Notification Log file
- For ideas on resolving lock time
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055072.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055235.html>

Deadlock Diagnosis

- All deadlocks are abnormal
- Cost of deadlocks can be high as work must be re-done
- Look at number of deadlocks over time
- For ideas on resolving deadlocks:
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055074.html>

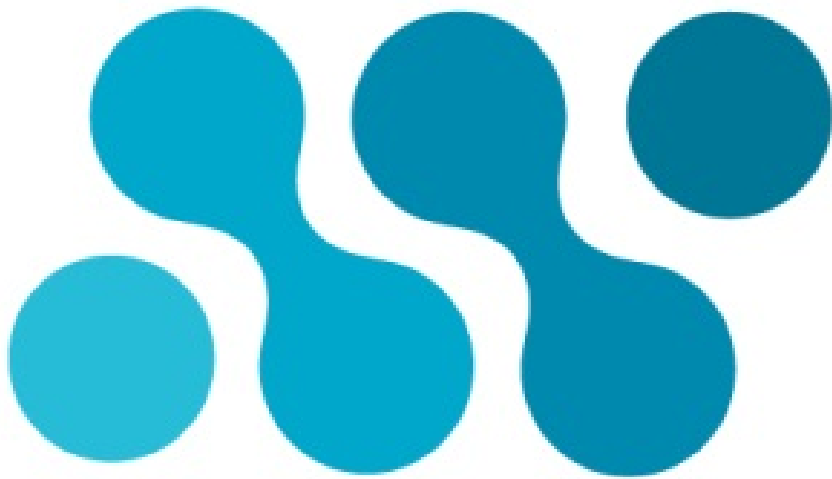
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055236.html>

Lock Escalation Diagnosis

- Lock escalation will occur if the lock list runs out of space. The application that causing the lock list to fill has its locks forced to escalate, even if not holder of most locks
- For ideas on resolving lock escalations:
<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055238.html>

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.trb.doc/doc/t0055237.html>

A Commercial Break from
today's show sponsor...



dbi

Smarter Tools for Smarter People

www.DBISoftware.com

SAP/DB2/Windows w/ DBI

3X Faster, 1.5% more CPU

Compare Database Workload													
Before: 3/14/14 9:00 AM to 3/14/14 3:00 PM										Last Refresh: 4/29/14 4:47 PM			
After: 4/29/14 9:00 AM to 4/29/14 3:00 PM										Rows: 1			
Server	Instance	Database	IX Read Efficiency After	IX Read Efficiency Change	% CPU Busy Improved	% CPU Busy Before	% CPU Busy After	% CPU Busy Change	SQL Sync Read % Improved	SQL Sync Read % Before	SQL Sync Read % After	SQL Sync Read % Change	Logi Rea Imp
SAP.PRIM...	db2per	PER	2.568	-27.055	N	28.760%	30.320%	1.560%	Y	57.452%	62.423%	4.971%	

Compare Database Workload													
Before: 3/14/14 9:00 AM to 3/14/14 3:00 PM										Last Refresh: 4/29/14 4:47 PM			
After: 4/29/14 9:00 AM to 4/29/14 3:00 PM										Rows: 1			
Server	Instance	Database	Tx Avg Outside Time (ms) After	Tx Avg Outside Time (ms) Change	SQL/Sec Improved	SQL/Sec Before	SQL/Sec After	SQL/Sec Change	IX Read Efficiency Improved	IX Read Efficiency Before	IX Read Efficiency After	IX Read Efficiency Change	% CF Impr
SAP.PRIM...	db2per	PER	947.46475	285.39264	Y	1,226.930	3,765.232	2,538.302	Y	29.623	2.568	-27.055	

Compare Database Workload													
Before: 3/14/14 9:00 AM to 3/14/14 3:00 PM										Last Refresh: 4/29/14 4:47 PM			
After: 4/29/14 9:00 AM to 4/29/14 3:00 PM										Rows: 1			
Server	Instance	Database	SQL Sync Read % Change	Logical IX Reads/Tx Improved	Logical IX Reads/Tx Before	Logical IX Reads/Tx After	Logical IX Reads/Tx Change	Logical Reads/Tx Improved	Logical Reads/Tx Before	Logical Reads/Tx After	Logical Reads/Tx Change	Tmp Logical Reads/Tx Improved	Tmp Read Befo
SAP.PRIM...	db2per	PER	4.971%	Y	742.37	193.40	-548.97	Y	6,152.67	550.79	-5,601.88	Y	

Today's Show was sponsored by...



Smarter Tools for Smarter People

www.DBISoftware.com

Contact Sales@DBISoftware.com

Toll Free: 1-866-773-8789 x101

International: 1-512-249-2324

**“Your trusted partner for
Breakthrough DB2 Performance
Solutions that DELIVER INVALUABLE
RESULTS for Organizations having
the most Demanding Requirements
and Discriminating Preferences”**