

SAP® MaxDB™ Expert Session

SAP® MaxDB™ Performance Optimization
News 2015

Public

The SAP logo is located in the bottom left corner of the image. It consists of the letters 'SAP' in white, bold, sans-serif font, set against a blue rectangular background.



SAP® MaxDB™ Expert Session

SAP® MaxDB™ Performance Optimization

Christiane Hienger
IMS MaxDB/liveCache Development Support
2015



News - General Remarks

- Some functionality might have been changed or expanded over the years.
 - Recommendations might have changed.
- ➔ For each expert session an update with the most important changes is published.
- Check the additional remarks after having heard or read the expert session.
 - Feedback to the responsible author is welcome if you recognize outdated information within the session.



Agenda

1. Performance Analysis: Parameter Setting

2. News: Command Monitor



Performance Analysis: Parameter Setting (1)

EnableIOTimeStatistics

Setting this parameter to 'YES' starts collection of timing information used for monitoring I/O throughput. Since it is online changeable, it is possible to enable or disable collection of time information during runtime. Measuring time of I/O operations adds only a minimal overhead, therefore it is turned on by default.

UseExtendedTimeMeasurement

This parameter allows to enforce extended time measurement with database start. The default value for this database parameter is YES (enable time measurement).

Time measurement:

- is necessary to provide useful statistics regarding command monitoring
- allows the Database Analyzer to collect more detailed statistics

Note that command and resource monitoring is not possible when time measurement is switched off.

Time measurement can be switched off with the console command `TIME DISABLE` or by setting this database parameter to `NO`, which you can do in `ONLINE` mode.

Performance Analysis: Parameter Setting (2)

UseSharedSQL

Enables the use of SharedSQL

'YES': SharedSQL is used for caching SQLCommands

'NO' : SharedSQL will not be used

Note that command and resource monitoring is not possible if Shared SQL is switched off.

SharedSQLCommandCacheSize

Specifies the maximum size (in KB) for SharedSQL statement caching

The lower and upper limits:

32768 (=32MByte) \leq UseSharedSQLCommandCacheSize $<$ 8388608 (=8GByte)

DEFAULT = 262144



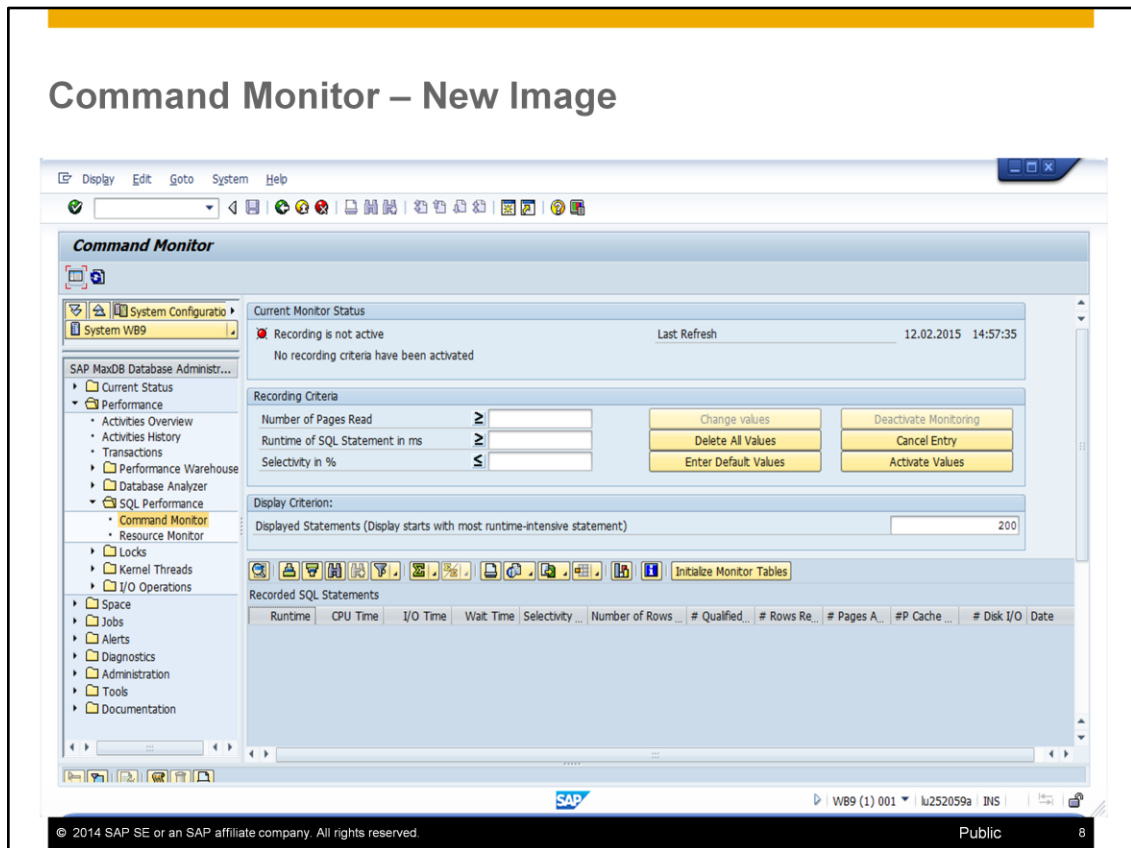
Agenda

1. Performance Analysis: Parameter Setting

2. **News:** Command Monitor



Command Monitor – New Image



As of SAP MaxDB Version 7.9 the image of the Command Monitor has been changed.

Use the *Command Monitor* if the analysis of the database bottlenecks (Database Analyzer) reports inefficient database accesses. With this tool, you can identify long running SQL statements in a systematic way.

The tool should be used for **short-term analysis**. You can filter the SQL statements by first specifying logging criteria. This enables you to concentrate on long-running SQL statements.

Enter the desired recording criteria in the *Change value* display. Now you can determine which SQL statements are to be logged in the command monitor tables. If one of the set criteria is fulfilled the SQL statement is logged in the command monitor.

Number of Pages Read: An SQL statement is logged if the number of specified page accesses is exceeded.

Runtime of SQL Statement in ms: An SQL statement is logged if the specified runtime is exceeded.

Selectivity in %: An SQL statement is logged in the command monitor tables if the ratio of qualified records to read records falls below the specified percentage.

SAP provides default values for these thresholds. Use *Enter Default Values* and then adapt or confirm these values.

You have to explicitly activate the monitoring criteria with *Activate Values*.

Command Monitor – Aggrandised Measured Data

The screenshot displays the SAP Command Monitor interface. At the top, there is a menu bar with 'Display', 'Edit', 'Goto', 'System', and 'Help'. Below the menu is a toolbar with various icons. The main area is titled 'Command Monitor' and contains several sections:

- Current Monitor Status:** Shows 'Recording Active' with a green checkmark and 'Last Refresh' at 12.02.2015 15:16:18. Below it, 'Pay attention to database fill level' is indicated with an 'i' icon, and 'Monitoring active since' is at 12.02.2015 15:14:51.
- Recording Criteria:** A table with columns for criteria, operators, and values.

Number of Pages Read	≥	0	Change values	Deactivate Monitoring
Runtime of SQL Statement in ms	≥	1.000	Delete All Values	Cancel Entry
Selectivity in %	≤	0	Enter Default Values	Activate Values
- Display Criterion:** A dropdown menu set to 'Displayed Statements (Display starts with most runtime-intensive statement)' and a text input field set to '200'.
- Recorded SQL Statements:** A table with columns for various performance metrics and SQL statements.

Runtime	CPU Time	I/O Time	Wait Time	Selectivity	Number of Rows	# Qualified	# Rows Re	# Pages A	#P Cache	# Disk I/O	Date	Time	Abbreviated SQL Statem
3.172.874	1,16	7,69	91,17	2,31	48.518	1.123	103,83	1.166	1.139	27	12.02.2015	15:15:19	SELECT "OBJECTNAME" I
3.153.057	0,98	0,43	98,61	2,31	48.518	1.123	97,42	1.094	1.087	7	12.02.2015	15:15:19	SELECT "OBJECTNAME" I

With the new Command Monitor a more detailed analysis is possible.

You will now get a detailed measurement where the runtime of a command was spent:

CPU-Time: % of CPU usage

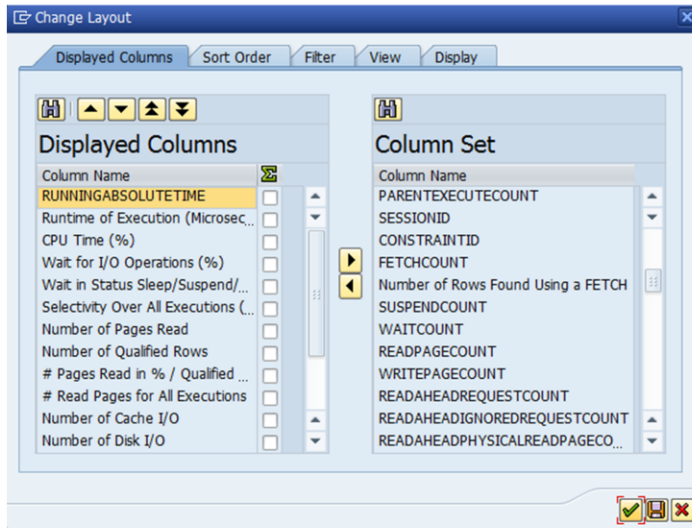
IO Time: wait for IO operations in %

Wait-Time: Wait situations in status: Sleep/Suspend/Wait in %

These new statistic values are displayed by default in the monitoring screen.

There are a lot more detailed statistical values which can be selected and displayed by choosing *Change Layout*.

Command Monitor: More Statistics in Change Layout



IOWAITREADTIME
 IOWAITWRITETIME
 PHYSICALREADS, PHYSICALWRITES, PHYSICALPAGESREAD, PHYSICALPAGESWRITTEN
 FETCHCOUNT, Number of Rows Found Using a FETCH
 SUSPENDCOUNT, WAITCOUNT
 READPAGECOUNT, WRITEPAGECOUNT
 READAHEADREQUESTCOUNT, READAHEADIGNOREDREQUESTCOUNT, READAHEADPHYSICALREADPAGECOUNT
 DISPATCHCOUNT, TASKSELFSUSPENDCOUNT
 REGIONACCESSCOUNT
 SELFIOREADCOUNT
 SELFIOREADPAGECOUNT
 AVGSELFIOREADABSOLUTETIME
 SELFIOREADABSOLUTETIME
 SELFIOWRITECOUNT
 SELFIOWRITEPAGECOUNT
 AVGSELFIOWRITEABSOLUTETIME
 SELFIOWRITEABSOLUTETIME
 IOREADCOUNT
 IOREADPAGECOUNT
 AVGIOREADRELATIVETIME
 IOREADRELATIVETIME
 AVGIOREADABSOLUTETIME
 IOREADABSOLUTETIME
 IOWRITECOUNT
 IOWRITEPAGECOUNT
 AVGIOWRITERELATIVETIME
 IOWRITERELATIVETIME
 AVGIOWRITEABSOLUTETIME
 IOWRITEABSOLUTETIME
 AVGSUSPENDRELATIVETIME
 SUSPENDRELATIVETIME
 AVGSUSPENDABSOLUTETIME
 SUSPENDABSOLUTETIME
 AVGWAITRELATIVETIME
 WAITRELATIVETIME
 AVGWAITABSOLUTETIME
 WAITABSOLUTETIME
 SLEEPCOUNT
 AVGSLEEPRELATIVETIME
 SLEEPRELATIVETIME
 AVGSLEEPABSOLUTETIME
 SLEEPABSOLUTETIME
 RUNNINGCOUNT
 AVGRUNNINGABSOLUTETIME
 RUNNINGABSOLUTETIME
 SCHEDULINGCOUNT
 AVGSCHEDULINGABSOLUTETIME
 SCHEDULINGABSOLUTETIME

Command Monitor: Additional IO Statistics (1)

The screenshot shows the SAP Command Monitor interface. The left sidebar contains a tree view with categories like 'Performance', 'SQL Performance', and 'Resource Monitor'. The main area is titled 'Command Monitor' and includes sections for 'Current Monitor Status', 'Recording Criteria', and 'Recorded SQL Statements'.

Current Monitor Status

Recording Active	Last Refresh	19.02.2015 11:00:53
Pay attention to database fill level	Monitoring active since	19.02.2015 10:42:24

Recording Criteria

Number of Pages Read	≥	0	Change values	Deactivate Monitoring
Runtime of SQL Statement in ms	≥	1.000	Delete All Values	Cancel Entry
Selectivity in %	≤	0	Enter Default Values	Activate Values

Display Criterion:

Displayed Statements (Display starts with most runtime-intensive statement)

Recorded SQL Statements

Runtime	IOWAITREAD	IOWAITWRITETIME	PHYSICALREADS	PHYSICALWRL	PHYSICALPAGESREAD	PHYSICALPAGESWRITTEN	CPU Time	I/O Time
1.762.600	1.687.782	0	225	0	3.263	0	4,38	95,76
3.153.057	13.667	0	7	0	7	0	0,98	0,43
3.172.874	244.065	0	27	0	27	0	1,16	7,69

The following IO statistic values are not part of the default view. It is possible to select them via *Choose Layout*.

In this slide the following columns have been additionally selected:

IOWAITREADTIME, IOWAITWRITETIME, PHYSICALREADS, PHYSICALWRITES, PHYSICALPAGESREAD, PHYSICALPAGESWRITTEN

The unit is microseconds.

You can exactly see if the runtime was spent in IO. In this example we can see the total runtime of the commands in the first column.

The command in the first line ran 1,7 seconds and nearly the total runtime was spent with waiting for IO to read the data: For 225 physical reads, which read 3263 physical pages.

This command did not perform any write IO. The execution time includes 4,38% CPU time but the main part of the execution time (95,76%) was spent with I/O.

=> the high runtime of this command was caused by the time spent for IO

Can we get more details about the time spent for IO? Yes we can – see next slide

Command Monitor: Additional IO Statistics (2)

The screenshot shows the SAP Command Monitor interface. The 'Recording Criteria' section is expanded, showing the following settings:

- Number of Pages Read: \geq 0
- Runtime of SQL Statement in ms: \geq 1.000
- Selectivity in %: \leq 0

The 'Display Criterion' is set to 'Displayed Statements (Display starts with most runtime-intensive statement)' with a value of 200.

The 'Recorded SQL Statements' table is displayed below:

Runtime	IOWAITREAD	IOREADRELATIVETIME	IOREADABSOLUTETIME	AVGIOREADRELATIVETIME	AVGIOAREA	PHYSICALREADS	CPU Time
1.762.600	1.687.782	1.674.877	1.687.782	7.443	7.501	225	4,38
3.153.057	13.667	13.171	13.667	1.881	1.952	7	0,98
3.172.874	244.065	242.352	244.065	8.976	9.039	27	1,16

If we analyze a long running SQL command whose runtime was caused by IO time mainly, we can get more detailed information about where the time exactly was spent.

Background information: SAP MaxDB Kernel is running in one OS process which consists of several threads. Threads which contain user tasks are so called User Kernel Threads (UKT). SQL statements are executed by user tasks. Several user tasks are running in the same User Kernel Thread. Only one task can have the CPU at one time. To avoid a CPU bottleneck in the UKT if a task requests IO, the User Kernel Thread does not execute the IO request by itself but orders special threads, so called IO threads which are executing the IO requests.

If no other task in the same UKT is active e.g. the others are in status Connect Wait, the task is able to do the IO by itself. This action is called self-IO.

If a task has done self-IO the following statistics are of interest:
 SELFIOREADABSOLUTETIME, SELFIOWRITECOUNT, SELFIOWRITEPAGECOUNT,
 AVGSELFIOWRITEABSOLUTETIME, SELFIOWRITEABSOLUTETIME

In most cases the tasks are not doing self-IO. The IO orders are done via the IO threads. Then the following additional columns are of interest and should be selected:
 IOWAITREADTIME,IOREADRELATIVETIME,IOREADABSOLUTETIME,
 AVGIOREADRELATIVETIME, AVGIOREADABSOLUTETIME

The Command Monitor displays values of relative and absolute Read time of each command.

Absolute time: Measurement starts when the task sends an IO request and stops when the IO complete information reaches the user task.

Relative time: Measurement starts when the IO workerthread sends the IO request to the IO sub system.

Command Monitor: Additional CPU Statistics

```
SELECT COUNT ( * ) FROM DOMAIN.SYSMON_US
WHERE UPPER ( STATEDESC ) <> 'CONNECT WAIT'
```

The screenshot shows the SAP Command Monitor interface. The 'Recording Criteria' section is expanded, showing the following settings:

- Number of Pages Read: \geq 0
- Runtime of SQL Statement in ms: \geq 0
- Selectivity in %: \leq 0

The 'Display Criterion' is set to 'Displayed Statements (Display starts with most runtime-intensive statement)' with a value of 200.

The 'Recorded SQL Statements' table is displayed below:

SQL Statement	Runtime	CPU Time	I/O Time	Wait Time	Selectivity	Number of Rows	# Qualified	# Rows Re.	# Pages A.	# P C
2.572.292	161.004.3	1,60	0,53	97,89	0,00	3.630.076	0	0,00	133.277	13
409.019	2.607.413	15,69	84,73	0,13	100,00	35.029	35.029	307,50	107.715	10
47.364	2.258.688	2,10	98,18	0,06	100,00	2.000	2.000	367,15	7.343	7
2.133.655	2.142.135	99,60	0,00	0,00	100,00	31	31	19,35	6	6

In this example there is a command which has no IO time but 99,6 % CPU time. You can get more detailed information e.g. by listing the column: RUNNINGABSOLUTETIME – showing the time the CPU was in use by this command.



Thank you

Contact information:

Heike Gursch
IMS MaxDB / liveCache
Heike.Gursch@sap.com

Christiane Hienger
IMS MaxDB / liveCache
Christiane.Hienger@sap.com

© 2014 SAP SE or an SAP affiliate company. All rights reserved.