Do more with less

Resource and cost saving approach for
DB2 SQL workloads on z/OS

Ulf Heinrich
SEGUS, Inc
u.heinrich@segus.com

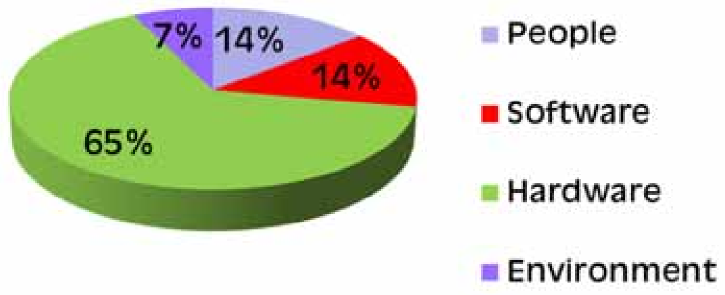
Agenda

- Past and future directions
- The mainframe from a costs perspective
- SQL workload : capture and analyze
- Typical approaches
- **SQL WorkloadExpert** for DB2 z/OS
  - GUI filtering and sorting
  - Data input
  - GUI views
- Types of applications
- Use cases
- Appendix

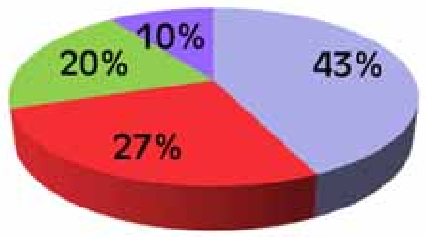
Past and future directions

- Changing cost drivers:
  Hardware → Software → Staff
  However a steady increase of overall operation costs even though less $ per MSU, there are dramatically more MSU in use.

1995 – TCO Allocation



2004 – TCO Allocation



Past and future directions

- Changing performance and differentiation of market players

- 1994 fastest hardware counts
- 2004 best software (home grown/bought) counts
- 2014 z/Enterprize EC12, BIG data, Business Analytics & High Performance. Fastest Hardware, more and data, best evaluations required

The from a costs perspective?

Ways of facing this dilemma:

- Moving applications from z/OS to AIX
- Shifting workload to ZIIPs and ZAAPs
- Implementing NETEZZA (Accelerators)
- SQL Workload Tuning

Goal: Reducing the 4 hours rolling average

- Quality Assurance metrics to stabilize resource consumption and performance

- Avoiding or minimizing peaks, performance bottlenecks and growing staff

- Increase automation and controlling changes
  - system maintenance and system changes
  - access path management of applications
  - monitoring recovery prerequisites and SLAs

SQL workload : capture and analyze

- The different flavors:
  - (Production) monitoring

    - monitors (OMPE;Strobe,FreezeFrame..)
      but often not economically possible for 24/7
      even in its most efficient monitoring class
    - SQL Collectors
      but complex and time consuming

  - DB2 10 resource-friendly IFCID technology getting dynamic & static SQL

NEW

SQL WorkloadExpert for DB2 z/OS

The architecture:

SOURCES

Real Time Statistics
RTS

DB2 Catalog
& Explain Data
for static SQL

DSN MSTR System
Services Address Space
DSC + EDM + IFCIDs

STC Started Task
24 x 7

Processing

- Explain for dynamic SQL
- Aggregation
- DB2 Objects
  - tables
  - indexes

Repository

SQL Workload Warehouse

Type 4 ⟷ JAVA Connection

Interface

Eclipse native
Rational
DataStudio

WLX reduces capturing costs by efficient IFCIDs

- The enhanced IFCIDs delivers all the important info to enable smart analysis of SQL workload

- The required IFCIDs are cheap and can be processed very fast via a STC (24x7 to catch all thrown IFCIDs)

- These IFCIDs can be snapped periodically, allowing big time frames (month, quarter, year)

- Redundant data can be summarized and aggregated

WLX provides what matters for any needs

Counters # EXECUTIONS OF THE STATEMENT. FOR A CURSOR STATEMENT, THIS IS THE # OF OPENS. # OF SYNCHRONOUS BUFFER READS PERFORMED FOR STATEMENT # OF GETPAGE OPERATIONS PERFORMED FOR STATEMENT. # OF ROWS EXAMINED FOR STATEMENT. # OF ROWS PROCESSED FOR STATEMENT - FOR EXAMPLE, THE # OF ROWS RETURNED FOR A SELECT, OR THE NUMBER OF ROWS AFFECTED BY AN INSERT, UPDATE, OR DELETE. # OF SORTS PERFORMED FOR STATEMENT. # OF INDEX SCANS PERFORMED FOR STATEMENT. # OF TABLESPACE SCANS PERFORMED FOR STATEMENT. * # OF PARALLEL GROUPS CREATED FOR STATEMENT. # OF SYNCHRONOUS BUFFER WRITE OPERATIONS PERFORMED FOR STATEMENT.

O Counters # OF TIMES THAT A RID LIST WAS NOT USED BECAUSE [illegible] # OF RIDS EXCEEDED ONE OR MORE INTERNAL DB2 LIMITS, AND THE # OF RID BLOCKS EXCEEDED THE VALUE OF SUBSYSTEM PARAMETER MAXTEMPS_RID. # OF TIMES THAT A RID LIST WAS NOT USED BECAUSE NOT ENOUGH STORAGE WAS AVAILABLE TO HOLD THE RID LIST, OR WORK FILE STORAGE OR RESOURCES WERE NOT AVAILABLE. # OF TIMES THAT A RID LIST OVERFLOWED TO A WORK FILE BECAUSE NO RID POOL STORAGE WAS AVAILABLE TO HOLD THE LIST OF RIDS*. # OF TIMES A THAT RID LIST OVERFLOWED TO A WORK FILE BECAUSE THE NUMBER OF RIDS EXCEEDED ONE OR MORE INTERNAL LIMITS*. # OF TIMES THAT APPENDING TO A RID LIST FOR A HYBRID JOIN WAS INTERRUPTED BECAUSE NO RID POOL STORAGE WAS AVAILABLE TO HOLD THE LIST OF RIDS*. # OF TIMES THAT APPENDING TO A RID LIST FOR A HYBRID JOIN WAS INTERRUPTED BECAUSE [illegible] NUMBER OF RIDS EXCEEDED ONE OR MORE INTERNAL LIMITS*. # OF TIMES THAT RID LIST RETRIEVAL FOR MULTIPLE INDEX ACCESS WAS NOT DONE BECAUSE DB2 COULD [illegible]RMINE THE OUTCOME OF INDEX ANDING OR ORING*.

TIMINGS ACCUMULATED CPU TIME. THIS VALUE INCLUDES CPU TIME THAT IS C[illegible] AN IBM SPECIALTY ENGINE. ACCUMULATED ELAPSED TIME USED FOR STATEMENT. ACCUMULATED WAIT TIME FOR LATCH REQUESTS*. ACCUMULATED WAIT TIME FOR PAGE LATCHES*. ACCUMULATED WAIT TIME FOR DRAIN LOCKS*. ACCUMULATED WAIT TIME FOR DRAINS DURING WAITS FOR CLAIMS TO BE RELEASED*. ACCUMULATED WAIT TIME FOR LOG WRITERS. ACCUMULATED WAIT TIME FOR SYNCHRONOUS I/O. ACCUMULATED WAIT TIME FOR LOCK REQUESTS. ACCUMULATED WAIT TIME FOR A SYNCHRONOUS EXECUTION UNIT SWITCH. ACCUMULATED WAIT TIME FOR GLOBAL LOCKS. ACCUMULATED WAIT TIME FOR READ ACTIVITY THAT IS DONE BY ANOTHER THREAD. ACCUMULATED WAIT TIME FOR WRITE ACTIVITY THAT IS DONE BY ANOTHER THREAD.

IDENTIFICATION DATA SHARING MEMBER THAT CACHED THE SQL STATEMENT*. PROGRAM NAME. PROGRAM NAME IS THE NAME OF THE PACKAGE OR DBRM THAT PERFORMED THE PREPARE/SQL. PRECOMPILER LINE NUMBER FOR THE PREPARE STATEMENT OR SQL STATEMENT. TRANSACTION NAME. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON. END USER ID*. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON. WORKSTATION NAME*. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON. USERID. USERID IS THE PRIMARY AUTH. ID OF THE USER WHO DID THE INITIAL PREPARE. USER GROUP. USER GROUP IS THE CURRENT SQLID OF THE USER WHO DID THE INITIAL PREPARE. USER-PROVIDED IDENTIFICATION STRING.

ENVIRONMENTAL REFERENCED TABLE NAME. FOR STATEMENTS THAT REFERENCE MORE THAN ONE TABLE, ONLY THE NAME OF THE FIRST TABLE THAT IS REFERENCED IS REPORTED. (ALL REFERENCED OBJECTS ARE STORED IN THE WLLDATA MODEL) LITERAL REPLACEMENT FLAG* CURRENT SCHEMA. QUALIFIER THAT IS USED FOR UNQUALIFIED TABLE NAMES. BIND OPTIONS: ISOLATION, CURRENTDATA, AND DYNAMICRULES. SPECIAL REGISTER VALUES: CURRENT DEGREE, CURRENT RULES, AND CURRENT PRECISION. WHETHER THE STATEMENT CURSOR IS A HELD CURSOR. TIMESTAMP WHEN STATISTICS COLLECTION BEGAN. DATA COLLECTION BEGINS WHEN A TRACE FOR IFCID 318 IS STARTED. DATE AND TIME WHEN THE STATEMENT WAS INSERTED INTO THE CACHE IN STORE CLOCK FORMAT. DATE AND TIME WHEN THE STATEMENT WAS UPDATED, IN STORE CLOCK FORMAT. DATE AND TIME WHEN THE STATEMENT WAS UPDATED, IN INTERNAL FORMAT.

WLX correlates elapsed time and access paths

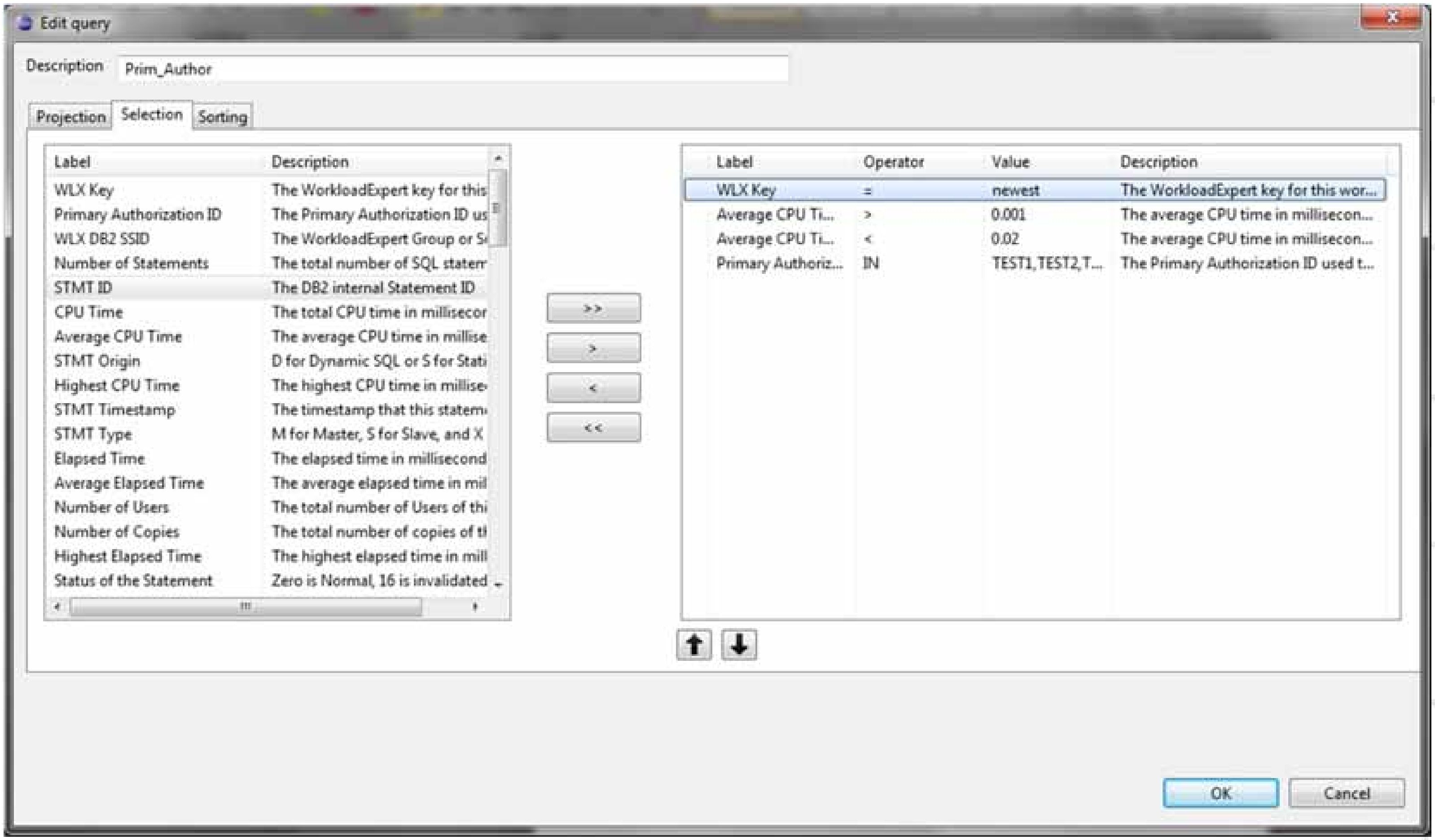
- It catches all SQL enterprise wide
  - All static (EDMPOOL=SSC) + dynamic (DSC) data is processed, including SQL thrown out of the caches
  - All statements are explained for access path determination and to match objects related to the SQL
- A „before/after“ comparison allows to identify and review access paths
- It does trending of KPIs for resource/costs considerations

Typical approaches

- Who does what, how, how often, and when ?
- Examples : CPU/timeframe -User ,-group
  -per table(s),-group
  -per statement
  Getpages/timeframe -User ,-group
  -per table(s),-group
- Presentation layer
  - INDIGO
  - Data Studio 3.2
  - Rational 8.2
- All KPIs are available on drill down levels (database, table, application etc.)

SQL WorkloadExpert for DB2 z/OS

- SQL Workload analysis – filter options



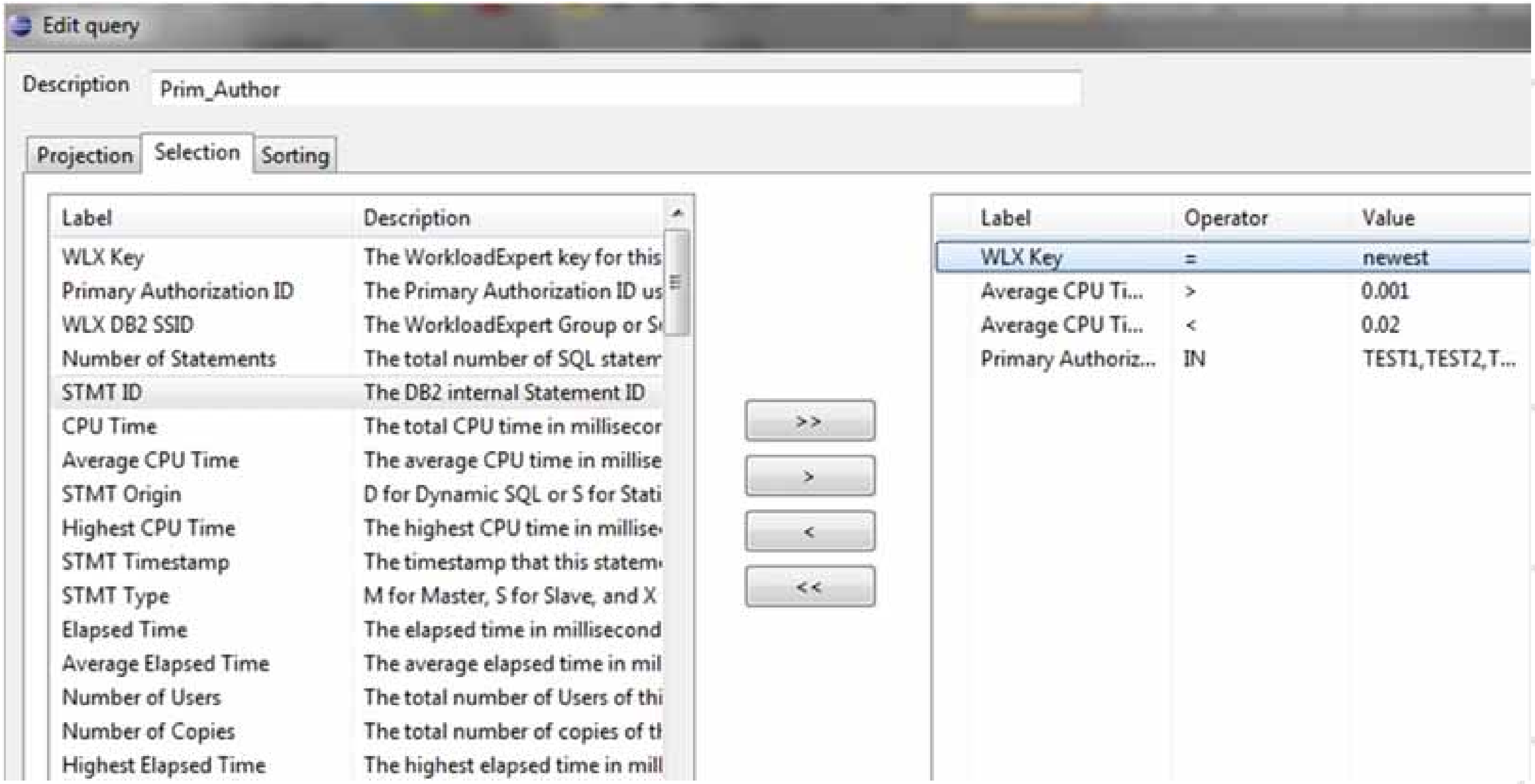
Description: Prim_Author

Label	Description
WLX Key	The WorkloadExpert key for this
Primary Authorization ID	The Primary Authorization ID us
WLX DB2 SSID	The WorkloadExpert Group or S
Number of Statements	The total number of SQL staterr
STMT ID	The DB2 internal Statement ID
CPU Time	The total CPU time in millisecor
Average CPU Time	The average CPU time in millise
STMT Origin	D for Dynamic SQL or S for Stati
Highest CPU Time	The highest CPU time in millise
STMT Timestamp	The timestamp that this statem
STMT Type	M for Master, S for Slave, and X
Elapsed Time	The elapsed time in millisecond
Average Elapsed Time	The average elapsed time in mil
Number of Users	The total number of Users of thi
Number of Copies	The total number of copies of tl
Highest Elapsed Time	The highest elapsed time in mill
Status of the Statement	Zero is Normal, 16 is invalidated

Label	Operator	Value	Description
WLX Key	=	newest	The WorkloadExpert key for this wor...
Average CPU Ti...	>	0.001	The average CPU time in millisecon...
Average CPU Ti...	<	0.02	The average CPU time in millisecon...
Primary Authoriz...	IN	TEST1,TEST2,T...	The Primary Authorization ID used t...

SQL WorkloadExpert for DB2 z/OS

- SQL Workload analysis – filter options



Edit query

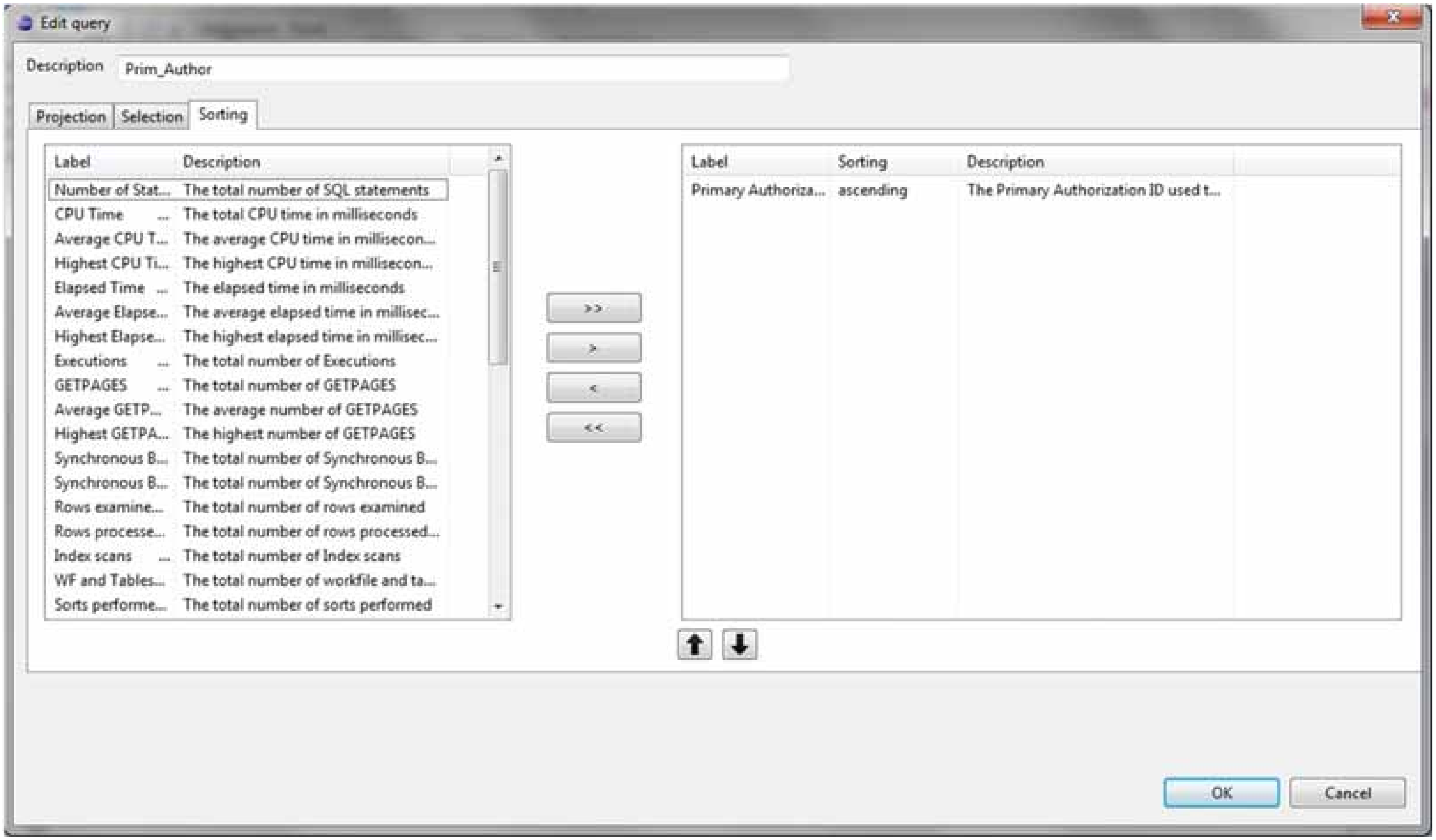
Description: Prim_Author

Label	Description
WLX Key	The WorkloadExpert key for this
Primary Authorization ID	The Primary Authorization ID us
WLX DB2 SSID	The WorkloadExpert Group or S
Number of Statements	The total number of SQL statem
STMT ID	The DB2 internal Statement ID
CPU Time	The total CPU time in millisecor
Average CPU Time	The average CPU time in millise
STMT Origin	D for Dynamic SQL or S for Stati
Highest CPU Time	The highest CPU time in millise
STMT Timestamp	The timestamp that this statem
STMT Type	M for Master, S for Slave, and X
Elapsed Time	The elapsed time in millisecond
Average Elapsed Time	The average elapsed time in mil
Number of Users	The total number of Users of thi
Number of Copies	The total number of copies of tl
Highest Elapsed Time	The highest elapsed time in mill

Label	Operator	Value
WLX Key	=	newest
Average CPU Ti...	>	0.001
Average CPU Ti...	<	0.02
Primary Authoriz...	IN	TEST1,TEST2,T...

SQL WorkloadExpert for DB2 z/OS

- SQL Workload analysis – sort options



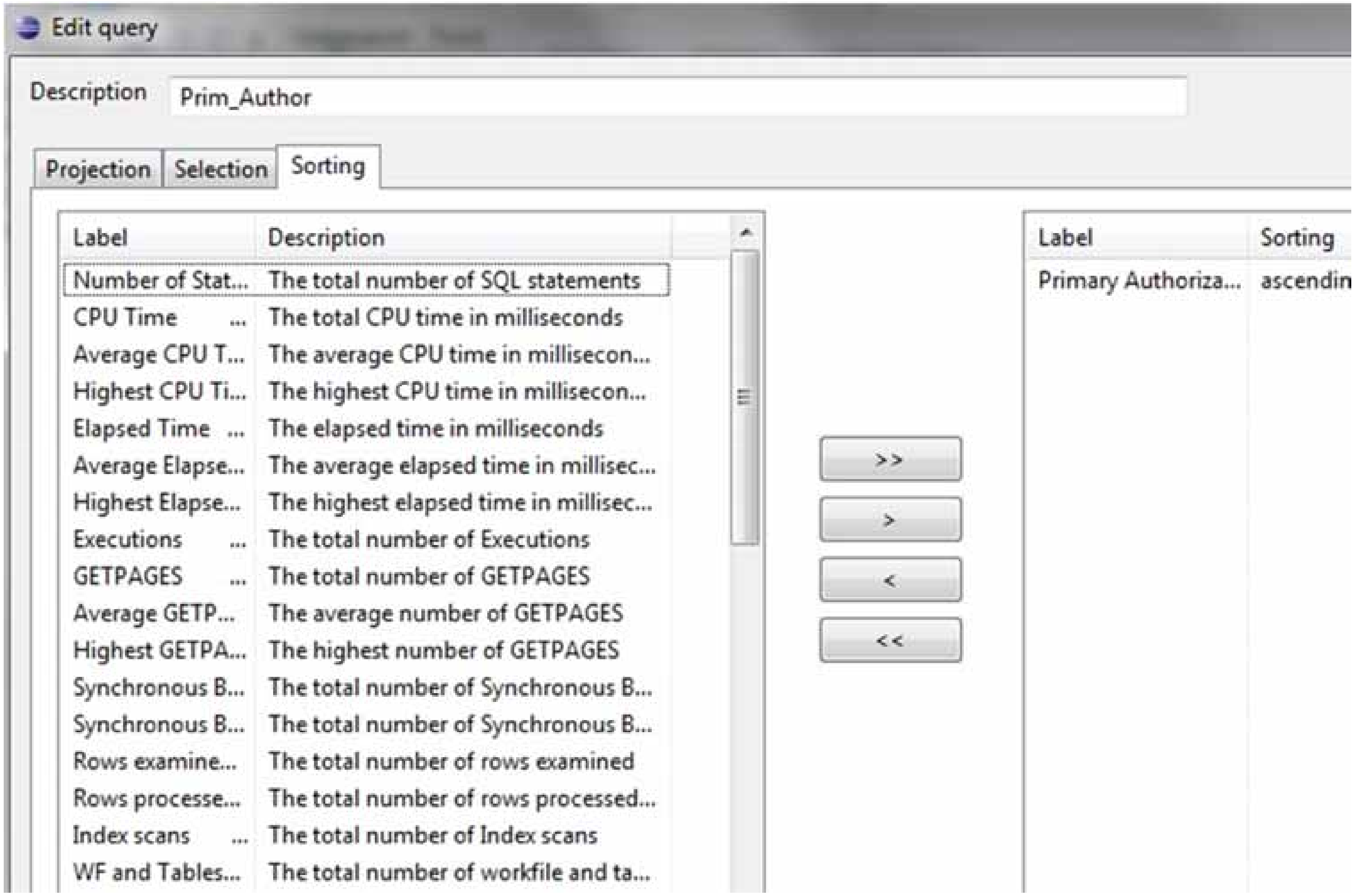
Description: Prim_Author

Label	Description
Number of Stat...	The total number of SQL statements
CPU Time ...	The total CPU time in milliseconds
Average CPU T...	The average CPU time in millisecon...
Highest CPU Ti...	The highest CPU time in millisecon...
Elapsed Time ...	The elapsed time in milliseconds
Average Elapse...	The average elapsed time in millisec...
Highest Elapse...	The highest elapsed time in millisec...
Executions ...	The total number of Executions
GETPAGES ...	The total number of GETPAGES
Average GETP...	The average number of GETPAGES
Highest GETPA...	The highest number of GETPAGES
Synchronous B...	The total number of Synchronous B...
Synchronous B...	The total number of Synchronous B...
Rows examine...	The total number of rows examined
Rows processe...	The total number of rows processed...
Index scans ...	The total number of Index scans
WF and Tables...	The total number of workfile and ta...
Sorts performe...	The total number of sorts performed

Label	Sorting	Description
Primary Authoriza...	ascending	The Primary Authorization ID used t...

SQL WorkloadExpert for DB2 z/OS

- SQL Workload analysis – sort options



Edit query

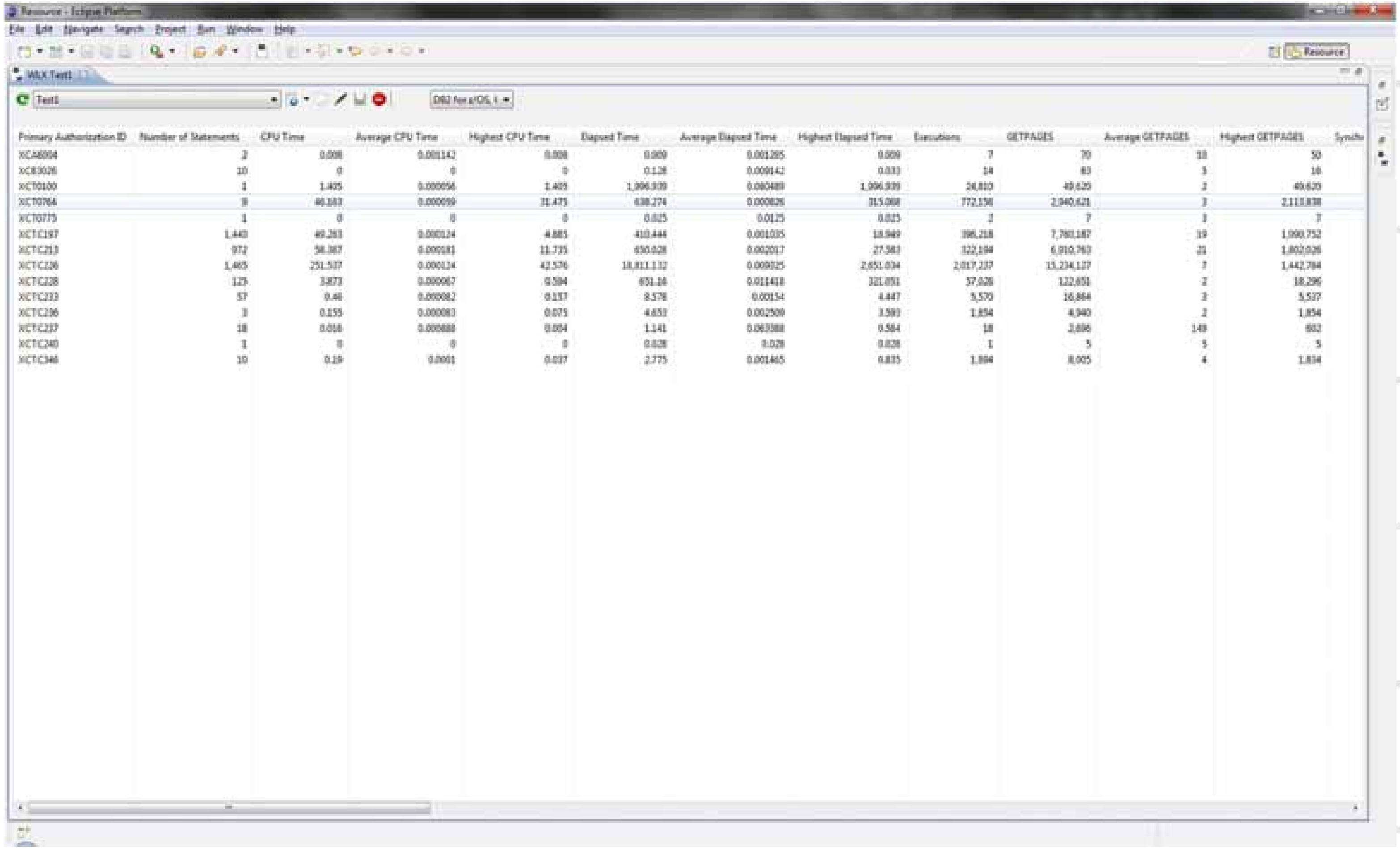
Description: Prim_Author

Label	Description
Number of Stat...	The total number of SQL statements
CPU Time ...	The total CPU time in milliseconds
Average CPU T...	The average CPU time in millisecon...
Highest CPU Ti...	The highest CPU time in millisecon...
Elapsed Time ...	The elapsed time in milliseconds
Average Elapse...	The average elapsed time in millisec...
Highest Elapse...	The highest elapsed time in millisec...
Executions ...	The total number of Executions
GETPAGES ...	The total number of GETPAGES
Average GETP...	The average number of GETPAGES
Highest GETPA...	The highest number of GETPAGES
Synchronous B...	The total number of Synchronous B...
Synchronous B...	The total number of Synchronous B...
Rows examine...	The total number of rows examined
Rows processe...	The total number of rows processed...
Index scans ...	The total number of Index scans
WF and Tables...	The total number of workfile and ta...

Label	Sorting
Primary Authoriza...	ascendin

SQL WorkloadExpert for DB2 z/OS

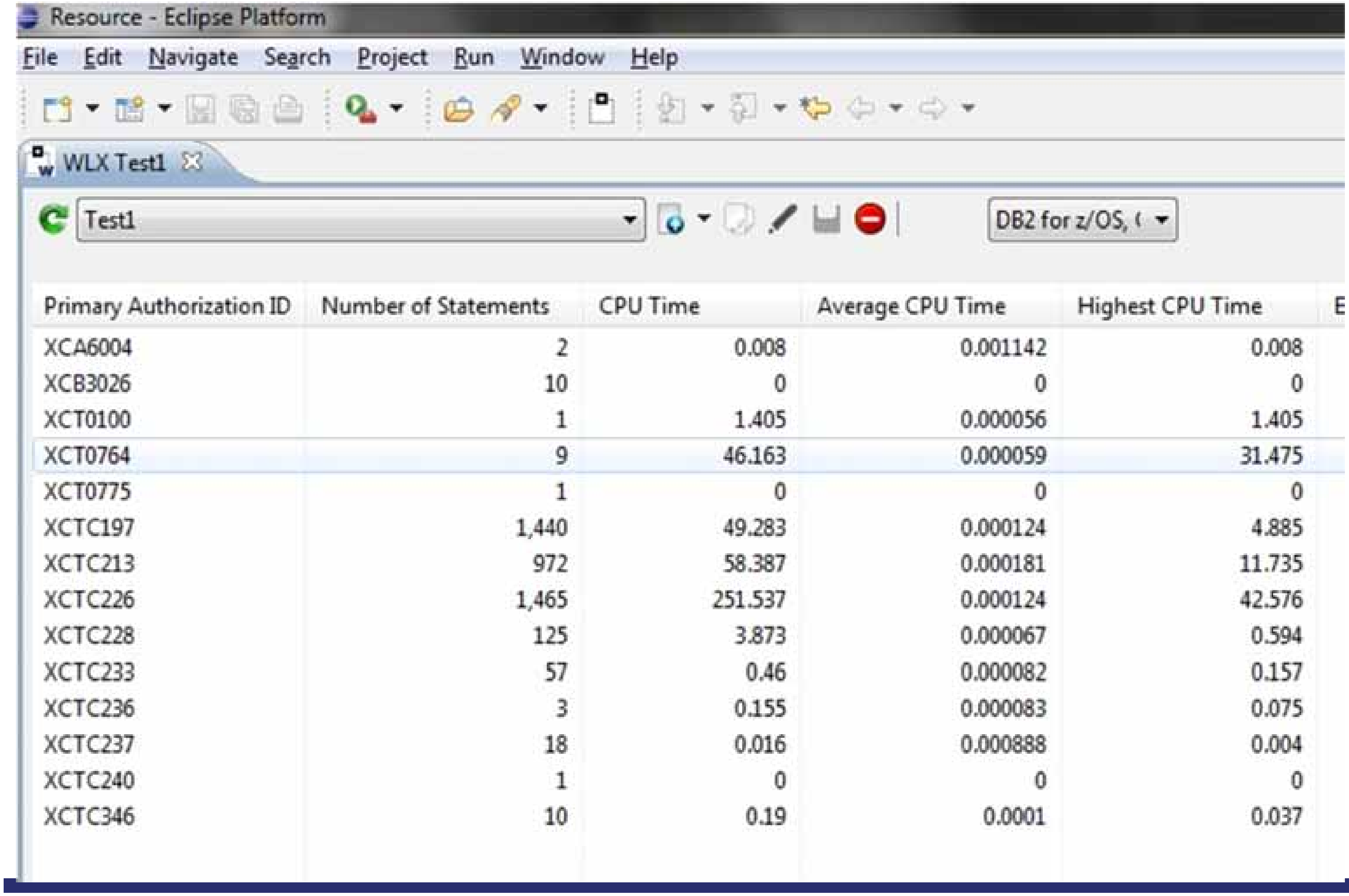
- SQL Workload analysis – results


Resource - Eclipse Platform
File Edit Navigate Search Project Run Window Help
Resource
WLX Test1
Test1
DB2 for z/OS, I

Primary Authorization ID	Number of Statements	CPU Time	Average CPU Time	Highest CPU Time	Elapsed Time	Average Elapsed Time	Highest Elapsed Time	Executions	GETPAGES	Average GETPAGES	Highest GETPAGES	Synch
XCA6004	2	0.008	0.001142	0.008	0.009	0.001285	0.009	7	70	10	50	
XCB3026	10	0	0	0	0.128	0.009142	0.033	14	83	3	16	
XCT0100	1	1.405	0.000056	1.405	1,996.939	0.080489	1,996.939	24,810	49,620	2	49,620	
XCT0764	9	46.163	0.000059	31.475	638.274	0.000826	315.068	772,156	2,940,621	3	2,111,838	
XCT0775	1	0	0	0	0.025	0.0125	0.025	2	7	3	7	
XCTC197	1,440	49.263	0.000124	4.885	410.444	0.001035	18.949	396,218	7,780,187	19	1,090,752	
XCTC213	972	58.387	0.000181	11.735	650.028	0.002017	27.583	322,194	6,910,763	21	1,802,026	
XCTC226	1,465	251.537	0.000124	42.576	18,811.132	0.009325	2,651.034	2,017,237	15,234,127	7	1,442,784	
XCTC228	125	3.873	0.000067	0.594	651.16	0.011418	321.051	57,026	122,651	2	18,296	
XCTC233	57	0.46	0.000082	0.137	8.578	0.00154	4.447	5,570	16,864	3	5,537	
XCTC236	3	0.155	0.000083	0.075	4.653	0.002509	3.593	1,854	4,940	2	1,854	
XCTC237	18	0.016	0.000888	0.004	1.141	0.063388	0.584	18	2,696	149	602	
XCTC240	1	0	0	0	0.028	0.028	0.028	1	5	5	5	
XCTC346	10	0.19	0.0001	0.037	2.775	0.001465	0.835	1,894	8,005	4	1,834	

SQL WorkloadExpert for DB2 z/OS

- SQL Workload analysis – results


Resource - Eclipse Platform
File Edit Navigate Search Project Run Window Help
WLX Test1
Test1
DB2 for z/OS, (

Primary Authorization ID	Number of Statements	CPU Time	Average CPU Time	Highest CPU Time	E
XCA6004	2	0.008	0.001142	0.008	
XCB3026	10	0	0	0	
XCT0100	1	1.405	0.000056	1.405	
XCT0764	9	46.163	0.000059	31.475	
XCT0775	1	0	0	0	
XCTC197	1,440	49.283	0.000124	4.885	
XCTC213	972	58.387	0.000181	11.735	
XCTC226	1,465	251.537	0.000124	42.576	
XCTC228	125	3.873	0.000067	0.594	
XCTC233	57	0.46	0.000082	0.157	
XCTC236	3	0.155	0.000083	0.075	
XCTC237	18	0.016	0.000888	0.004	
XCTC240	1	0	0	0	
XCTC346	10	0.19	0.0001	0.037	

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – data

All data is categorized in four general groups

- Counters
- Timings
- Identification
- Environmental

SQL Workload analysis – Counters:

- NUMBER OF EXECUTIONS OF THE STATEMENT. FOR A CURSOR STATEMENT, THIS IS THE NUMBER OF OPENS.
- NUMBER OF SYNCHRONOUS BUFFER READS PERFORMED FOR STATEMENT.
- NUMBER OF GETPAGE OPERATIONS PERFORMED FOR STATEMENT.
- NUMBER OF ROWS EXAMINED FOR STATEMENT.
- NUMBER OF ROWS PROCESSED FOR STATEMENT - FOR EXAMPLE, THE NUMBER OF ROWS RETURNED FOR A SELECT, OR THE NUMBER OF ROWS AFFECTED BY AN INSERT, UPDATE, OR DELETE.
- NUMBER OF SORTS PERFORMED FOR STATEMENT.
- NUMBER OF INDEX SCANS PERFORMED FOR STATEMENT.
- NUMBER OF TABLESPACE SCANS PERFORMED FOR STATEMENT.
- NUMBER OF PARALLEL GROUPS CREATED FOR STATEMENT.
- NUMBER OF SYNCHRONOUS BUFFER WRITE OPERATIONS PERFORMED FOR STATEMENT.
- NUMBER OF TIMES THAT A RID LIST WAS NOT USED BECAUSE THE NUMBER OF RIDS EXCEEDED ONE OR MORE INTERNAL DB2 LIMITS, AND THE NUMBER OF RID BLOCKS EXCEEDED THE VALUE OF SUBSYSTEM PARAMETER MAXTEMPS_RID.
- NUMBER OF TIMES THAT A RID LIST WAS NOT USED BECAUSE NOT ENOUGH STORAGE WAS AVAILABLE TO HOLD THE RID LIST, OR WORK FILE STORAGE OR RESOURCES WERE NOT AVAILABLE.
- NUMBER OF TIMES THAT A RID LIST OVERFLOWED TO A WORK FILE BECAUSE NO RID POOL STORAGE WAS AVAILABLE TO HOLD THE LIST OF RIDS.
- NUMBER OF TIMES A THAT RID LIST OVERFLOWED TO A WORK FILE BECAUSE THE NUMBER OF RIDS EXCEEDED ONE OR MORE INTERNAL LIMITS.
- NUMBER OF TIMES THAT APPENDING TO A RID LIST FOR A HYBRID JOIN WAS INTERRUPTED BECAUSE NO RID POOL STORAGE WAS AVAILABLE TO HOLD THE LIST OF RIDS.
- NUMBER OF TIMES THAT APPENDING TO A RID LIST FOR A HYBRID JOIN WAS INTERRUPTED BECAUSE THE NUMBER OF RIDS EXCEEDED ONE OR MORE INTERNAL LIMITS.
- NUMBER OF TIMES THAT RID LIST RETRIEVAL FOR MULTIPLE INDEX ACCESS WAS NOT DONE BECAUSE DB2 COULD DETERMINE THE OUTCOME OF INDEX ANDING OR ORING.

SQL Workload analysis – Timings:

- ACCUMULATED CPU TIME. THIS VALUE INCLUDES CPU TIME THAT IS CONSUMED ON AN IBM SPECIALTY ENGINE.
- ACCUMULATED ELAPSED TIME USED FOR STATEMENT.
- ACCUMULATED WAIT TIME FOR LATCH REQUESTS.
- ACCUMULATED WAIT TIME FOR PAGE LATCHES.
- ACCUMULATED WAIT TIME FOR DRAIN LOCKS.
- ACCUMULATED WAIT TIME FOR DRAINS DURING WAITS FOR CLAIMS TO BE RELEASED.
- ACCUMULATED WAIT TIME FOR LOG WRITERS.
- ACCUMULATED WAIT TIME FOR SYNCHRONOUS I/O.
- ACCUMULATED WAIT TIME FOR LOCK REQUESTS.
- ACCUMULATED WAIT TIME FOR A SYNCHRONOUS EXECUTION UNIT SWITCH.
- ACCUMULATED WAIT TIME FOR GLOBAL LOCKS.
- ACCUMULATED WAIT TIME FOR READ ACTIVITY THAT IS DONE BY ANOTHER THREAD.
- ACCUMULATED WAIT TIME FOR WRITE ACTIVITY THAT IS DONE BY ANOTHER THREAD.

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Identification:

- DATA SHARING MEMBER THAT CACHED THE SQL STATEMENT.
- PROGRAM NAME. PROGRAM NAME IS THE NAME OF THE PACKAGE OR DBRM THAT PERFORMED THE PREPARE/SQL.
- PRECOMPILER LINE NUMBER FOR THE PREPARE STATEMENT OR SQL STATEMENT.
- TRANSACTION NAME. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON.
- END USER ID. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON.
- WORKSTATION NAME. THIS VALUE IS PROVIDED DURING RRS SIGNON OR RE-SIGNON.
- USER ID. USER ID IS THE PRIMARY AUTH. ID OF THE USER WHO DID THE INITIAL PREPARE.
- USER GROUP. USER GROUP IS THE CURRENT SQLID OF THE USER WHO DID THE INITIAL PREPARE.
- USER-PROVIDED IDENTIFICATION STRING.

End User Id and Workstation Name are optional

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Environmental:

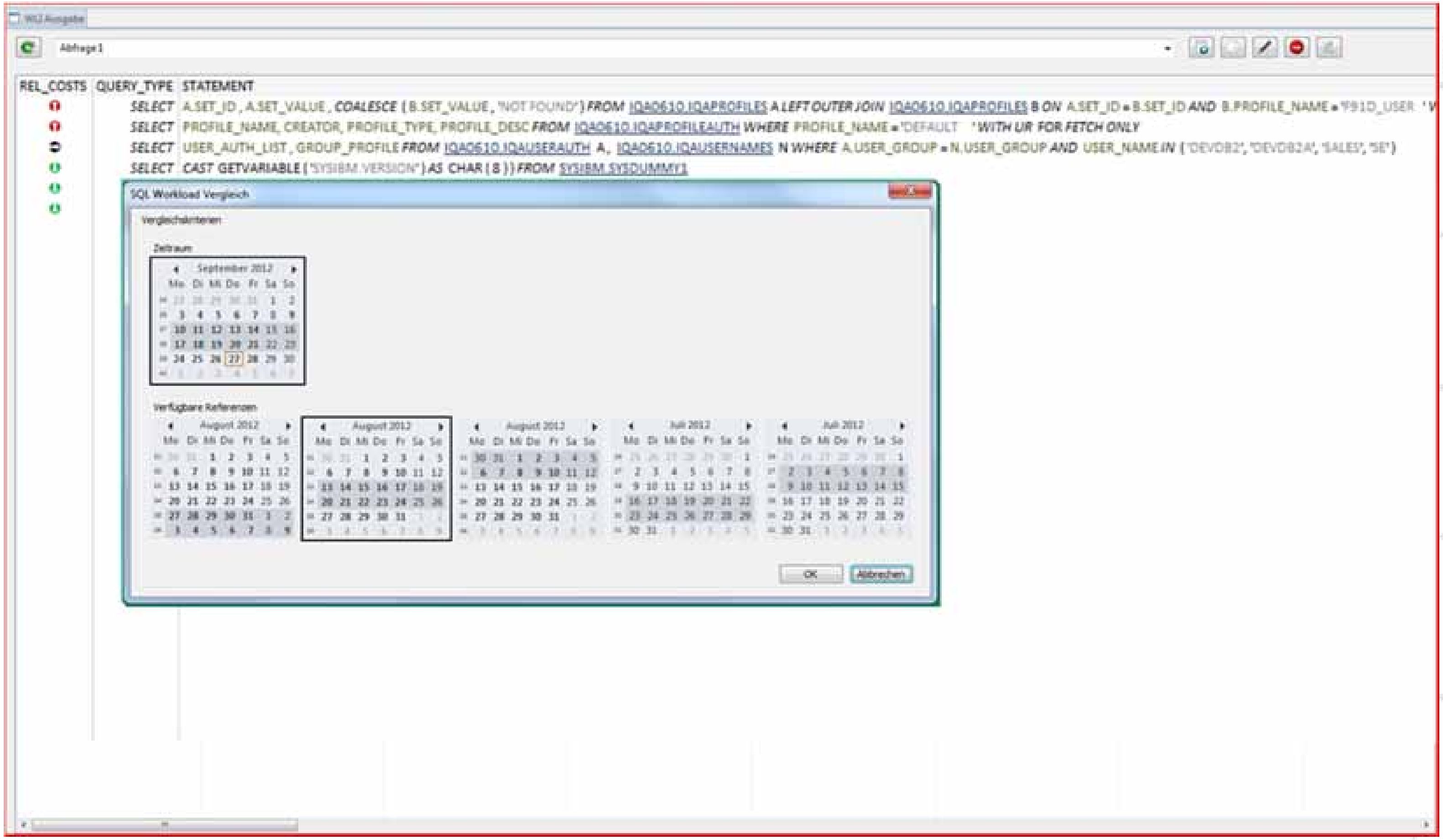
- REFERENCED TABLE NAME. FOR STATEMENTS THAT REFERENCE MORE THAN ONE TABLE, ONLY THE NAME OF THE FIRST TABLE THAT IS REFERENCED IS REPORTED.

- CURRENT SCHEMA.
- QUALIFIER THAT IS USED FOR UNQUALIFIED TABLE NAMES.
- BIND OPTIONS: ISOLATION, CURRENTDATA, AND DYNAMICRULES.
- SPECIAL REGISTER VALUES: CURRENT DEGREE, CURRENT RULES, AND CURRENT PRECISION.
- WHETHER THE STATEMENT CURSOR IS A HELD CURSOR.

- TIMESTAMP WHEN STATISTICS COLLECTION BEGAN. DATA COLLECTION BEGINS WHEN A TRACE FOR IFCID 318 IS STARTED.
- DATE AND TIME WHEN THE STATEMENT WAS INSERTED INTO THE CACHE IN STORE CLOCK FORMAT.
- DATE AND TIME WHEN THE STATEMENT WAS UPDATED, IN STORE CLOCK FORMAT.
- DATE AND TIME WHEN THE STATEMENT WAS UPDATED, IN INTERNAL FORMAT.

SQL WorkloadExpert for DB2 z/OS

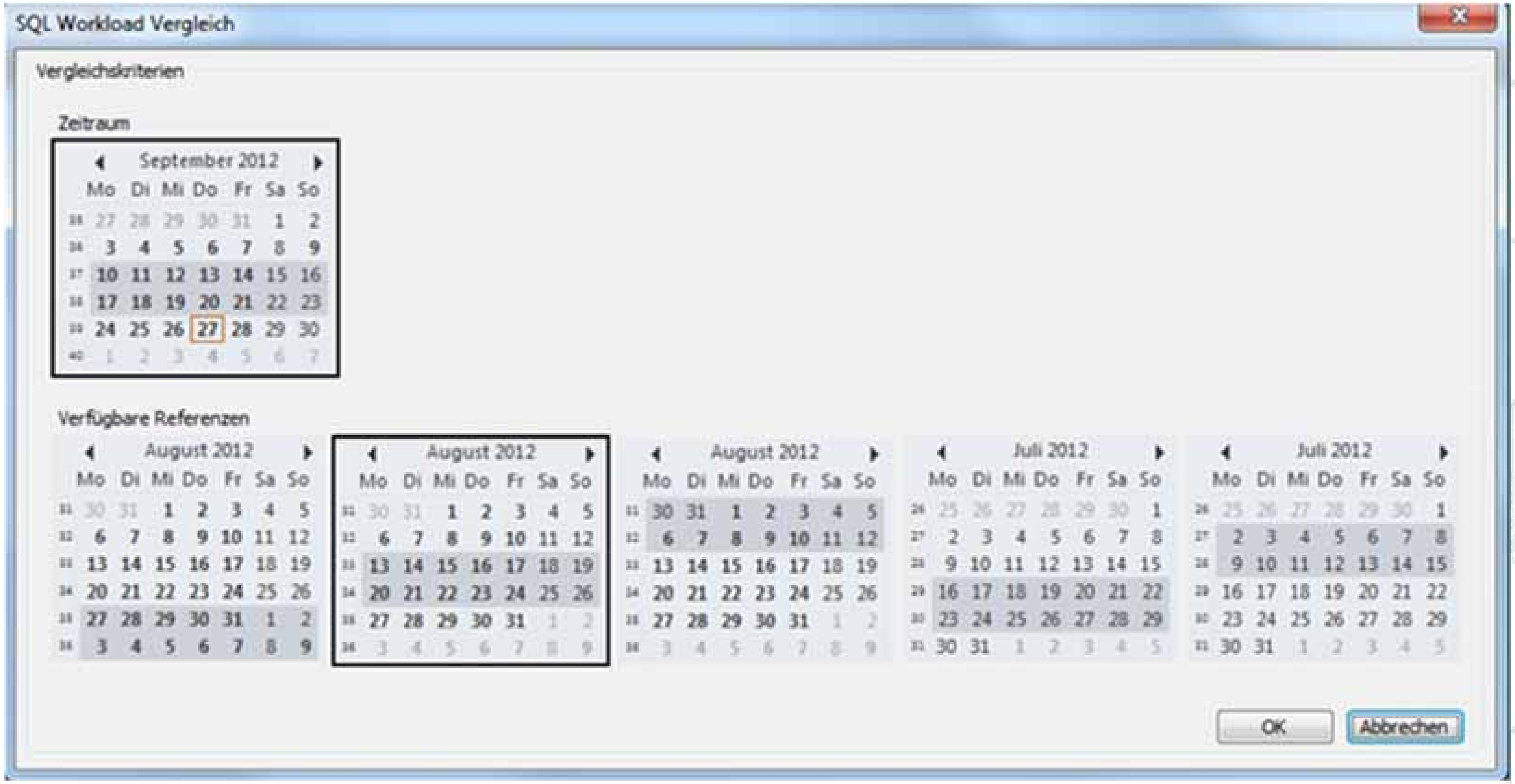
SQL Workload analysis – Comparison & Trending:



REL_COSTS	QUERY_TYPE	STATEMENT
	SELECT	A.SET_ID , A.SET_VALUE , COALESCE (B.SET_VALUE , 'NOT FOUND') FROM IQA0610.IQAPROFILES A LEFT OUTER JOIN IQA0610.IQAPROFILES B ON A.SET_ID = B.SET_ID AND B.PROFILE_NAME = 'F91D_USER ' W
	SELECT	PROFILE_NAME, CREATOR, PROFILE_TYPE, PROFILE_DESC FROM IQA0610.IQAPROFILEAUTH WHERE PROFILE_NAME = 'DEFAULT ' WITH UR FOR FETCH ONLY
	SELECT	USER_AUTH_LIST , GROUP_PROFILE FROM IQA0610.IQAUSERAUTH A , IQA0610.IQAUSERNAMES N WHERE A.USER_GROUP = N.USER_GROUP AND USER_NAME IN ('DEVDB2', 'DEVDB2A', 'SALES', 'SE')
	SELECT	CAST GETVARIABLE ('SYSIBM.VERSION') AS CHAR (8)) FROM SYSIBM.SYSDUMMY1

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Commparison & Trending:



SQL Workload Vergleich

Vergleichskriterien

Zeitraum

September 2012

	Mo	Di	Mi	Do	Fr	Sa	So
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9
37	10	11	12	13	14	15	16
38	17	18	19	20	21	22	23
39	24	25	26	27	28	29	30
40	1	2	3	4	5	6	7

Verfügbare Referenzen

August 2012

	Mo	Di	Mi	Do	Fr	Sa	So
31	30	31	1	2	3	4	5
32	6	7	8	9	10	11	12
33	13	14	15	16	17	18	19
34	20	21	22	23	24	25	26
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9

August 2012

	Mo	Di	Mi	Do	Fr	Sa	So
31	30	31	1	2	3	4	5
32	6	7	8	9	10	11	12
33	13	14	15	16	17	18	19
34	20	21	22	23	24	25	26
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9

August 2012

	Mo	Di	Mi	Do	Fr	Sa	So
31	30	31	1	2	3	4	5
32	6	7	8	9	10	11	12
33	13	14	15	16	17	18	19
34	20	21	22	23	24	25	26
35	27	28	29	30	31	1	2
36	3	4	5	6	7	8	9

Juli 2012

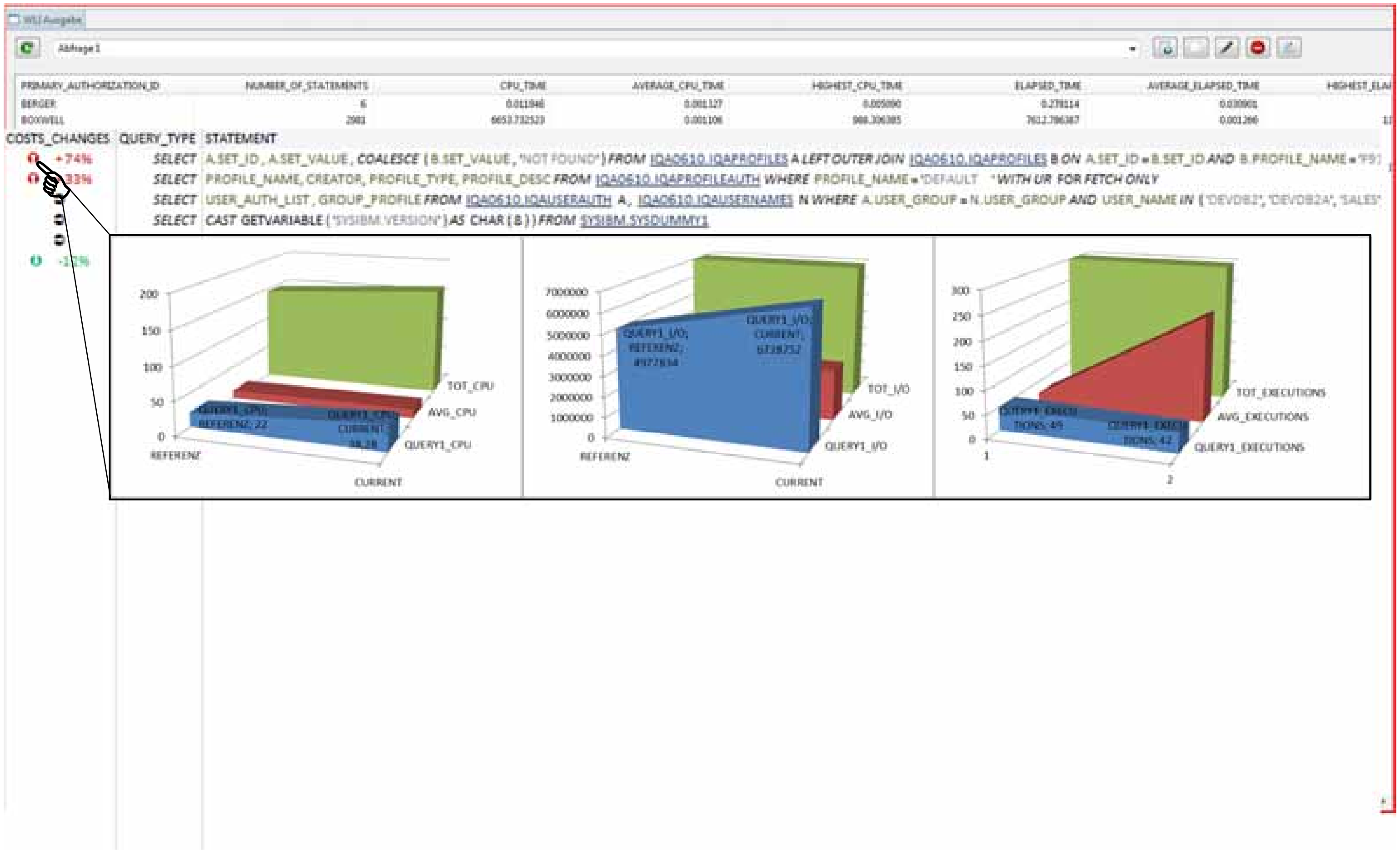
	Mo	Di	Mi	Do	Fr	Sa	So
26	25	26	27	28	29	30	1
27	2	3	4	5	6	7	8
28	9	10	11	12	13	14	15
29	16	17	18	19	20	21	22
30	23	24	25	26	27	28	29
31	30	31	1	2	3	4	5

Juli 2012

	Mo	Di	Mi	Do	Fr	Sa	So
26	25	26	27	28	29	30	1
27	2	3	4	5	6	7	8
28	9	10	11	12	13	14	15
29	16	17	18	19	20	21	22
30	23	24	25	26	27	28	29
31	30	31	1	2	3	4	5

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Comparison & Trending:

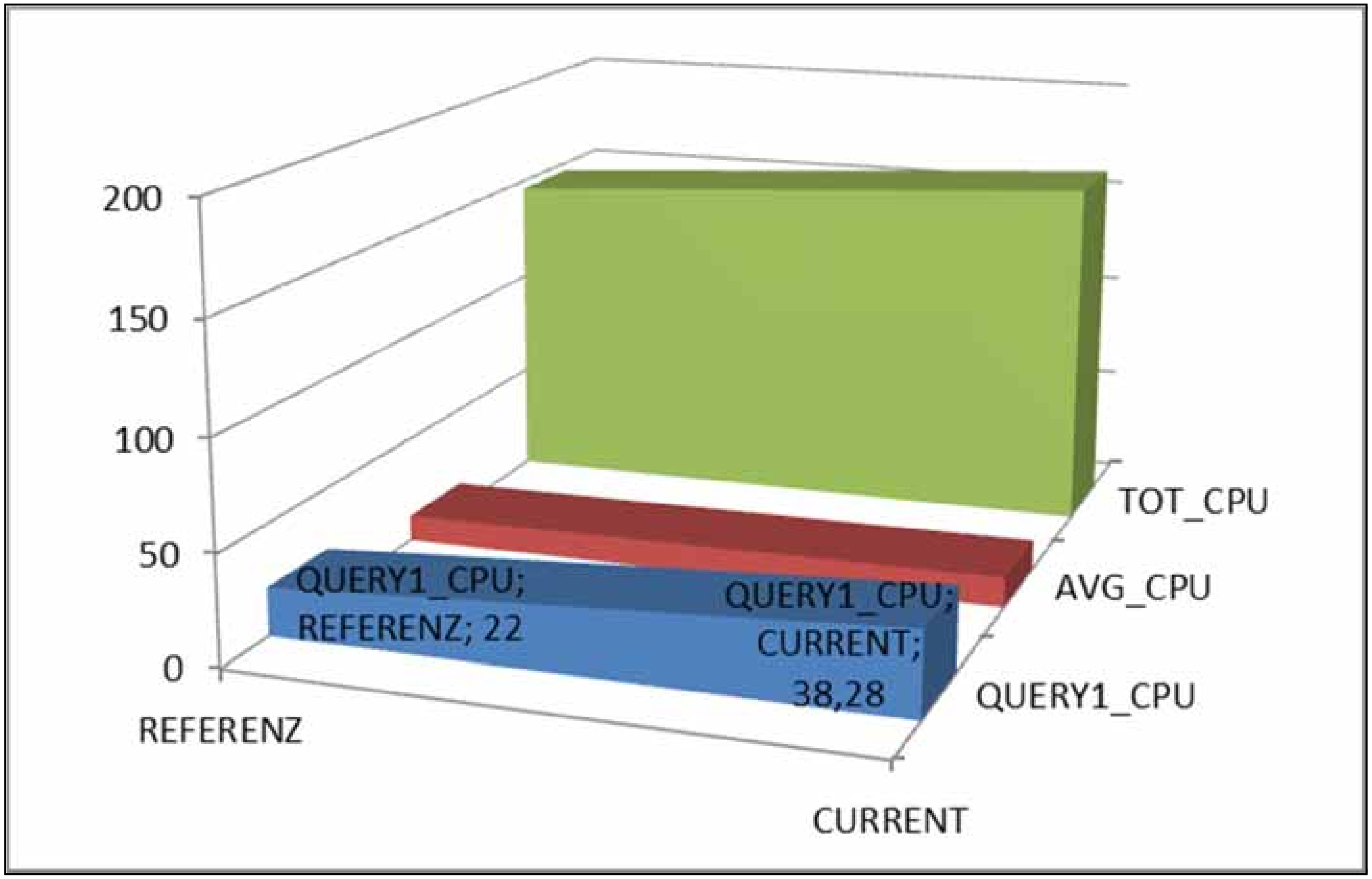


PRIMARY_AUTHORIZATION_ID	NUMBER_OF_STATEMENTS	CPU_TIME	AVERAGE_CPU_TIME	HIGHEST_CPU_TIME	ELAPSED_TIME	AVERAGE_ELAPSED_TIME	HIGHEST_ELA
BERGER	6	0.011946	0.001127	0.005090	0.278114	0.030901	
BOXWELL	2981	6653.732523	0.001106	988.306385	7612.786387	0.001266	1

COSTS_CHANGES	QUERY_TYPE	STATEMENT
+74%	SELECT	A.SET_ID , A.SET_VALUE , COALESCE (B.SET_VALUE , 'NOT FOUND') FROM IQA0610.IQAPROFILES A LEFT OUTER JOIN IQA0610.IQAPROFILES B ON A.SET_ID = B.SET_ID AND B.PROFILE_NAME = 'F9
+33%	SELECT	PROFILE_NAME, CREATOR, PROFILE_TYPE, PROFILE_DESC FROM IQA0610.IQAPROFILEAUTH WHERE PROFILE_NAME = 'DEFAULT ' WITH UR FOR FETCH ONLY
	SELECT	USER_AUTH_LIST , GROUP_PROFILE FROM IQA0610.IQAUSERAUTH A , IQA0610.IQAUSERNAMES N WHERE A.USER_GROUP = N.USER_GROUP AND USER_NAME IN ('DEVDB2', 'DEVDB2A', 'SALES'
	SELECT	CAST GETVARIABLE ('SYSIBM.VERSION') AS CHAR (8)) FROM SYSIBM.SYSDUMMY1
-12%		

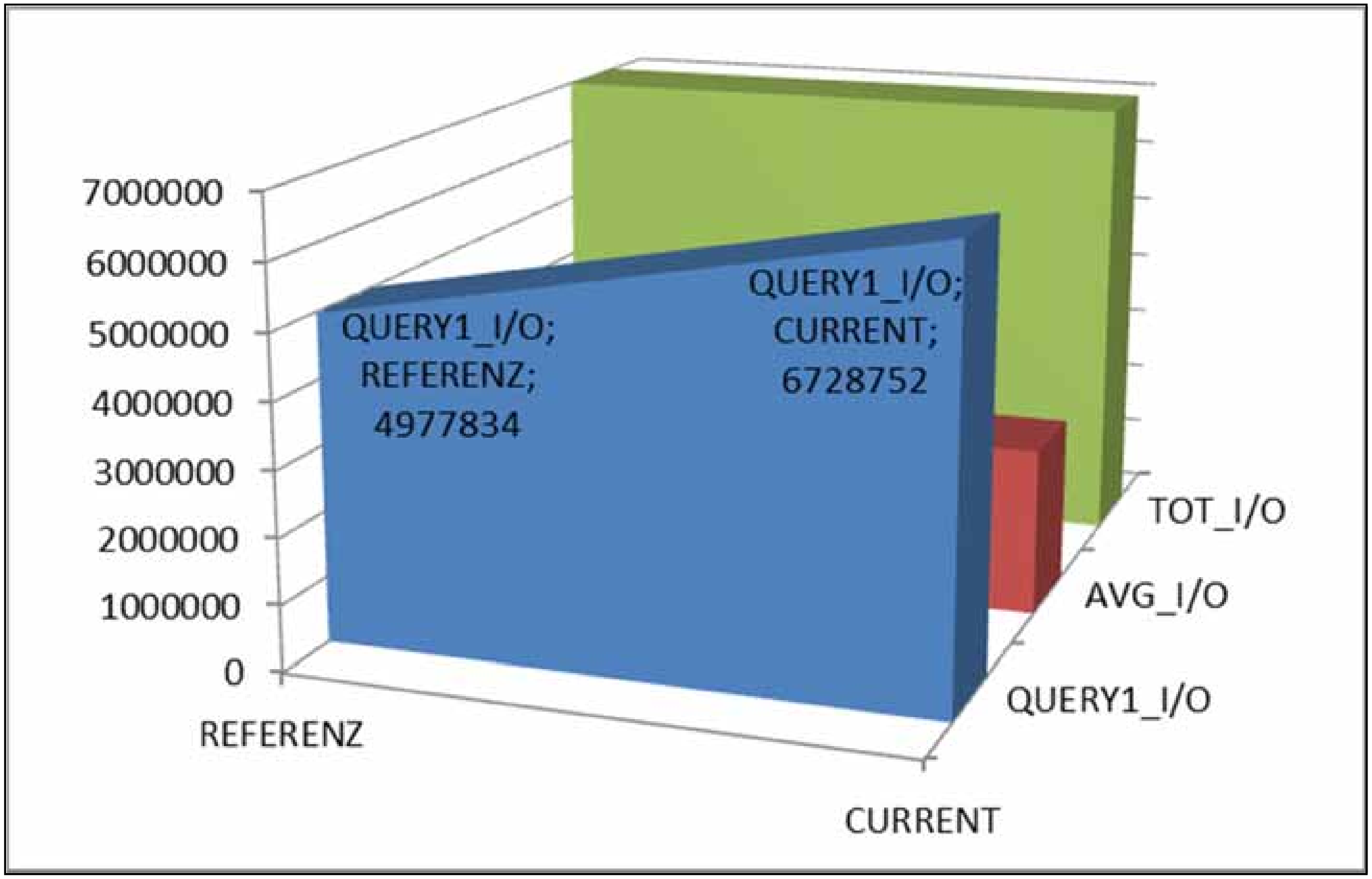
SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Comparison & Trending:



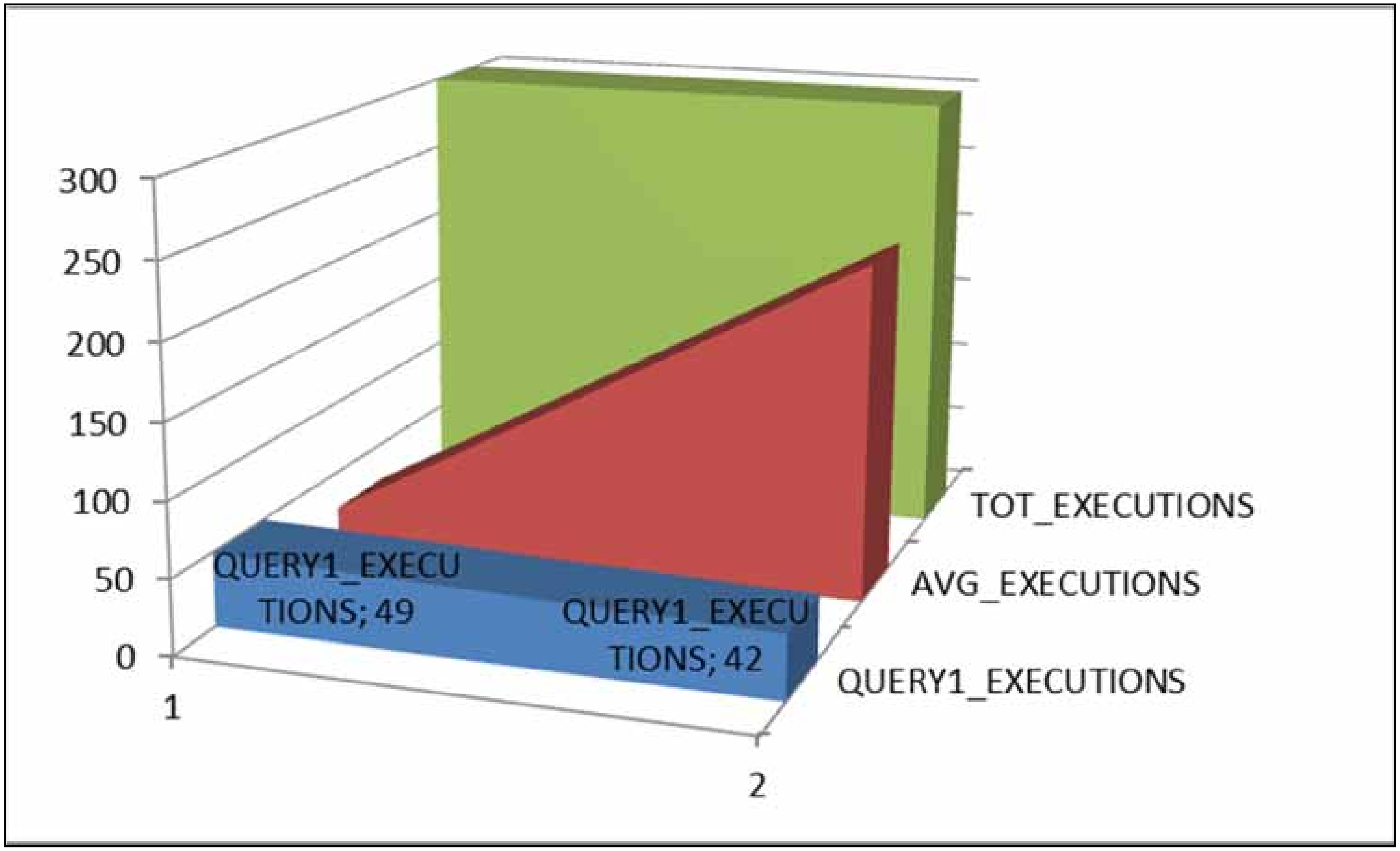
SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Commparison & Trending:



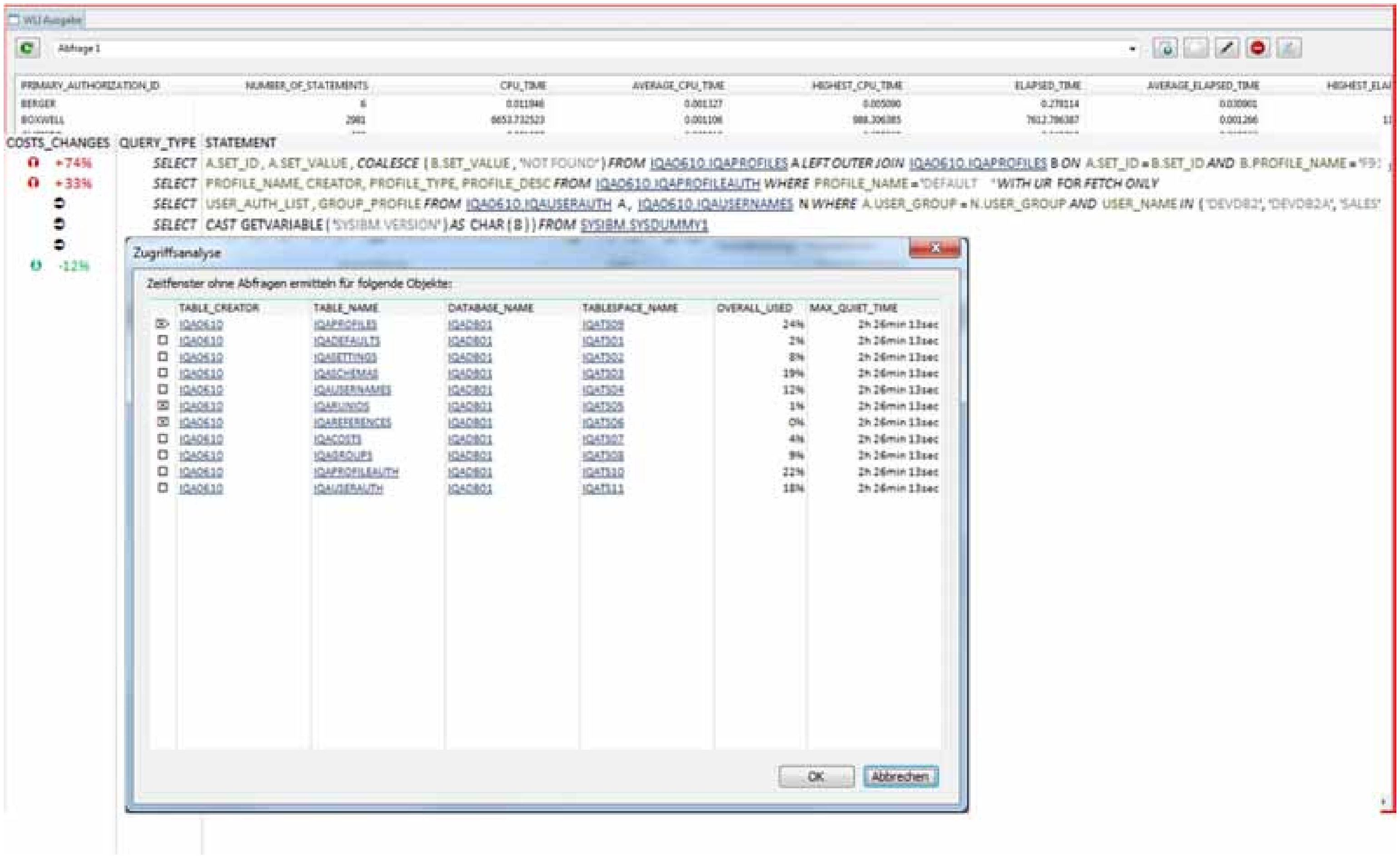
SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Comparison & Trending:



SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Objects Access & Usage:



PRIMARY_AUTHORIZATION_ID	NUMBER_OF_STATEMENTS	CPU_TIME	AVERAGE_CPU_TIME	HIGHEST_CPU_TIME	ELAPSED_TIME	AVERAGE_ELAPSED_TIME	HIGHEST_ELA
BERGER	6	0.011946	0.001127	0.005090	0.278114	0.030901	
BOXWELL	2981	8653.732523	0.001108	988.306385	7612.786387	0.001266	11

COSTS_CHANGES	QUERY_TYPE	STATEMENT
+74%	SELECT	A.SET_ID , A.SET_VALUE , COALESCE (B.SET_VALUE , 'NOT FOUND') FROM IQA0610.IQAPROFILES A LEFT OUTER JOIN IQA0610.IQAPROFILES B ON A.SET_ID = B.SET_ID AND B.PROFILE_NAME = '93:
+33%	SELECT	PROFILE_NAME, CREATOR, PROFILE_TYPE, PROFILE_DESC FROM IQA0610.IQAPROFILEAUTH WHERE PROFILE_NAME = 'DEFAULT' WITH UR FOR FETCH ONLY
●	SELECT	USER_AUTH_LIST , GROUP_PROFILE FROM IQA0610.IQAUSERAUTH A , IQA0610.IQAUSERNAMES N WHERE A.USER_GROUP = N.USER_GROUP AND USER_NAME IN ('DEVDB2', 'DEVDB2A', 'SALES'
●	SELECT	CAST GETVARIABLE ('SYSIBM.VERSION') AS CHAR (8)) FROM SYSIBM.SYSDUMMY1
●		
-12%		

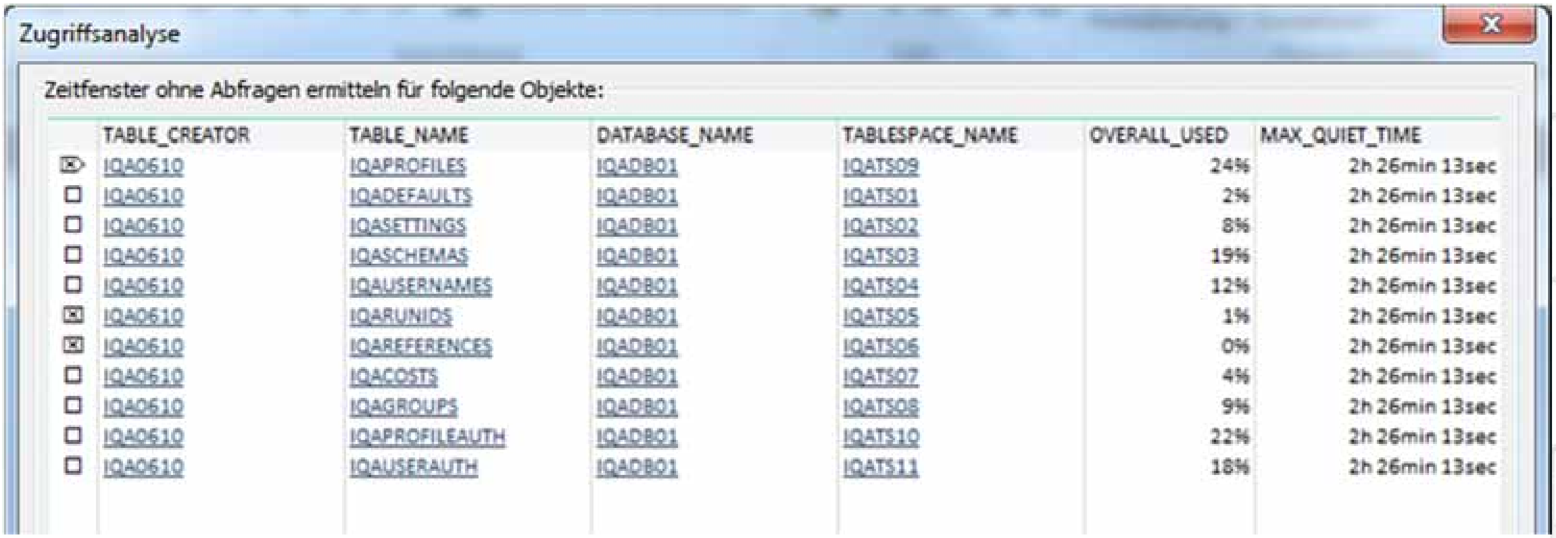
Zugriffsanalyse

Zeitfenster ohne Abfragen ermitteln für folgende Objekte:

	TABLE_CREATOR	TABLE_NAME	DATABASE_NAME	TABLESPACE_NAME	OVERALL_USED	MAX_QUIET_TIME
☒	IQA0610	IQAPROFILES	IQAD801	IQAT309	24%	2h 26min 13sec
□	IQA0610	IQADEFAULTS	IQAD801	IQAT301	2%	2h 26min 13sec
□	IQA0610	IQASETTINGS	IQAD801	IQAT302	8%	2h 26min 13sec
□	IQA0610	IQASCHEMAS	IQAD801	IQAT303	19%	2h 26min 13sec
□	IQA0610	IQAUSERNAMES	IQAD801	IQAT304	12%	2h 26min 13sec
☒	IQA0610	IQARUNIDS	IQAD801	IQAT305	1%	2h 26min 13sec
☒	IQA0610	IQAREFERENCES	IQAD801	IQAT306	0%	2h 26min 13sec
□	IQA0610	IQACOSTS	IQAD801	IQAT307	4%	2h 26min 13sec
□	IQA0610	IQAGROUPS	IQAD801	IQAT308	9%	2h 26min 13sec
□	IQA0610	IQAPROFILEAUTH	IQAD801	IQAT310	22%	2h 26min 13sec
□	IQA0610	IQAUSERAUTH	IQAD801	IQAT311	18%	2h 26min 13sec

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Objects Access & Usage:

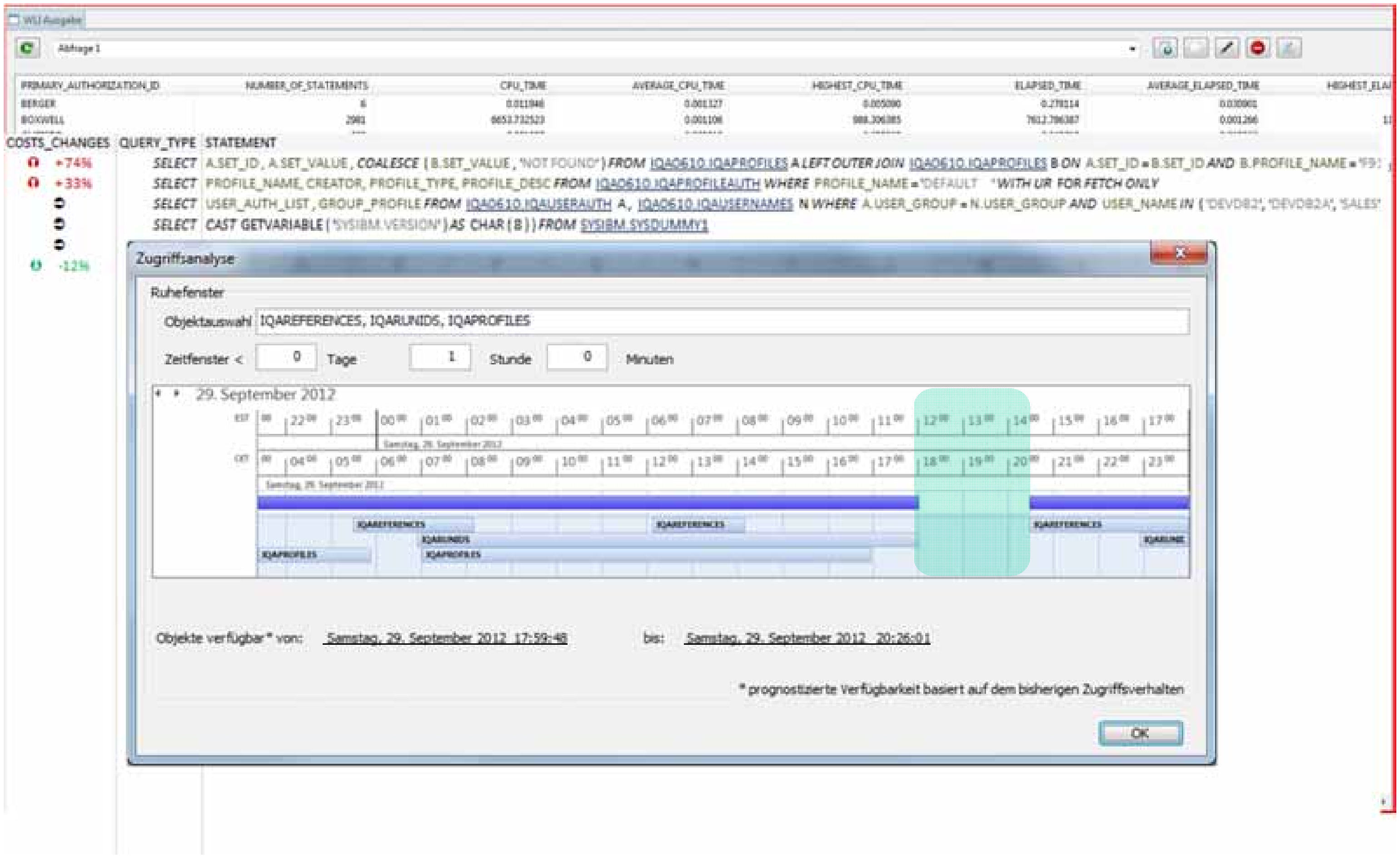


Zeitfenster ohne Abfragen ermitteln für folgende Objekte:

	TABLE_CREATOR	TABLE_NAME	DATABASE_NAME	TABLESPACE_NAME	OVERALL_USED	MAX_QUIET_TIME
☒	IQA0610	IQAPROFILES	IQADB01	IQATS09	24%	2h 26min 13sec
□	IQA0610	IQADEFAULTS	IQADB01	IQATS01	2%	2h 26min 13sec
□	IQA0610	IQASETTINGS	IQADB01	IQATS02	8%	2h 26min 13sec
□	IQA0610	IQASCHEMAS	IQADB01	IQATS03	19%	2h 26min 13sec
□	IQA0610	IQAUSERNAMES	IQADB01	IQATS04	12%	2h 26min 13sec
☒	IQA0610	IQARUNIDS	IQADB01	IQATS05	1%	2h 26min 13sec
☒	IQA0610	IQAREFERENCES	IQADB01	IQATS06	0%	2h 26min 13sec
□	IQA0610	IQACOSTS	IQADB01	IQATS07	4%	2h 26min 13sec
□	IQA0610	IQAGROUPS	IQADB01	IQATS08	9%	2h 26min 13sec
□	IQA0610	IQAPROFILEAUTH	IQADB01	IQATS10	22%	2h 26min 13sec
□	IQA0610	IQAUSERAUTH	IQADB01	IQATS11	18%	2h 26min 13sec

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Objects Access & Usage:



WLI Ausgabe

Abfrage 1

PRIMARY_AUTHORIZATION_ID	NUMBER_OF_STATEMENTS	CPU_TIME	AVERAGE_CPU_TIME	HIGHEST_CPU_TIME	ELAPSED_TIME	AVERAGE_ELAPSED_TIME	HIGHEST_ELA
BERGER	6	0.011946	0.001127	0.005090	0.278114	0.030901	
BOXWELL	2981	6653.732523	0.001106	988.306385	7612.786387	0.001266	1

COSTS_CHANGES	QUERY_TYPE	STATEMENT
+74%	SELECT	A.SET_ID, A.SET_VALUE, COALESCE (B.SET_VALUE, 'NOT FOUND') FROM IQA0610.IQAPROFILES A LEFT OUTER JOIN IQA0610.IQAPROFILES B ON A.SET_ID = B.SET_ID AND B.PROFILE_NAME = '99:
+33%	SELECT	PROFILE_NAME, CREATOR, PROFILE_TYPE, PROFILE_DESC FROM IQA0610.IQAPROFILEAUTH WHERE PROFILE_NAME = 'DEFAULT ' WITH UR FOR FETCH ONLY
	SELECT	USER_AUTH_LIST, GROUP_PROFILE FROM IQA0610.IQAUSERAUTH A, IQA0610.IQAUSERNAMES N WHERE A.USER_GROUP = N.USER_GROUP AND USER_NAME IN ('DEVDB2', 'DEVDB2A', 'SALES'
	SELECT	CAST GETVARIABLE ('SYSIBM.VERSION') AS CHAR (8)) FROM SYSIBM.SYSDUMMY1
-12%		

Zugriffsanalyse

Ruhefenster

Objektauswahl: IQAREFERENCES, IQARUNIDS, IQAPROFILES

Zeitfenster < 0 Tage 1 Stunde 0 Minuten

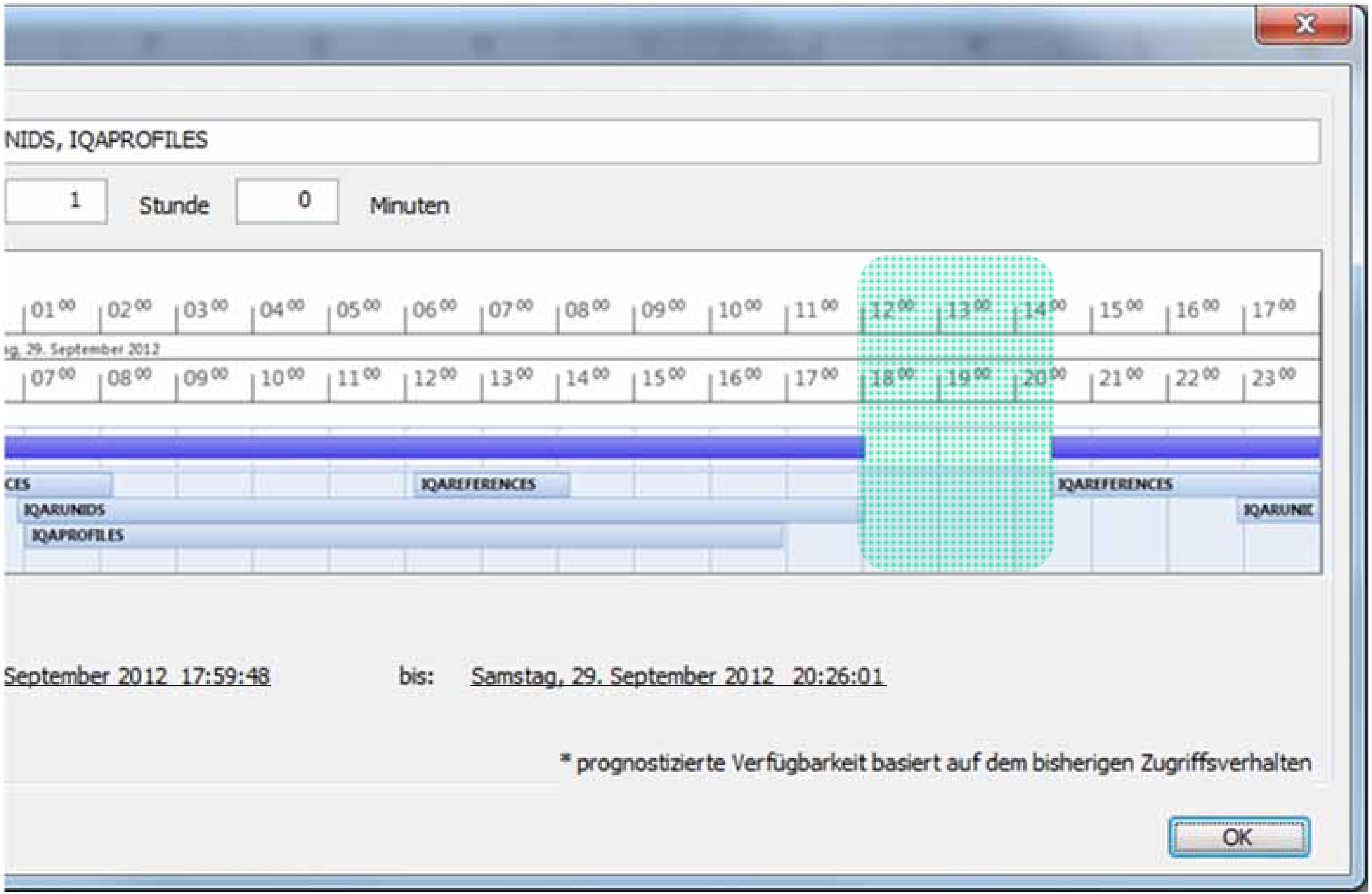
29. September 2012

Objekte verfügbar* von: Samstag, 29. September 2012 17:59:48 bis: Samstag, 29. September 2012 20:26:01

* prognostizierte Verfügbarkeit basiert auf dem bisherigen Zugriffsverhalten

SQL WorkloadExpert for DB2 z/OS

SQL Workload analysis – Objects Access & Usage:



NIDS, IQAPROFILES

1 Stunde 0 Minuten

September 2012 17:59:48 bis: Samstag, 29. September 2012 20:26:01

* prognostizierte Verfügbarkeit basiert auf dem bisherigen Zugriffsverhalten

WLX - Areas of use

- Enhance (classical) SQL performance tuning with lowest overhead
- Costs estimation and accounting even on the statement level
  - delivered with UDF for accounting integration
- Identifying the availability of objects for DB2 system and DB2 application maintenance
  - Which timeframe allows taking objects offline without compromising users access
- Identifying access patterns
  - When do we have certain activities from whom on which objects (audit purposes)

Summary

- This exciting new technology offers answers to the questions like:
  - Which queries represent my core business?
  - Where are bottlenecks?
  - What needs to be improved?
  - How do I control proactively changes in:
    - Applications
    - DB2 versions (8->10, 9->10, 10->11)
    - DB2 system maintenance (PTFs /APARs)
    - before REBINDs and PREPARES
    - DB2 SLAs for recovery purposes
  - Where and how could trends be discovered?

SQL WorkloadExpert on Trial

- Taste the „use case“ … you can‘t resist
- Appetite comes with eating. It‘s moreish.

Appendix

Problems list and sample customer timings:

Open APAR PM67255 – Difference between IFCID 401 and IFCID 58

UK70891 - Reset statistics when STOP/START MONITOR TRACE

UK72630 – Incorrect when executing same statement from different threads

UK73903 – Storage leak in code leading to abend

UK78414 – Storage overlay in code leading to abend

UK81878 - Storage leak

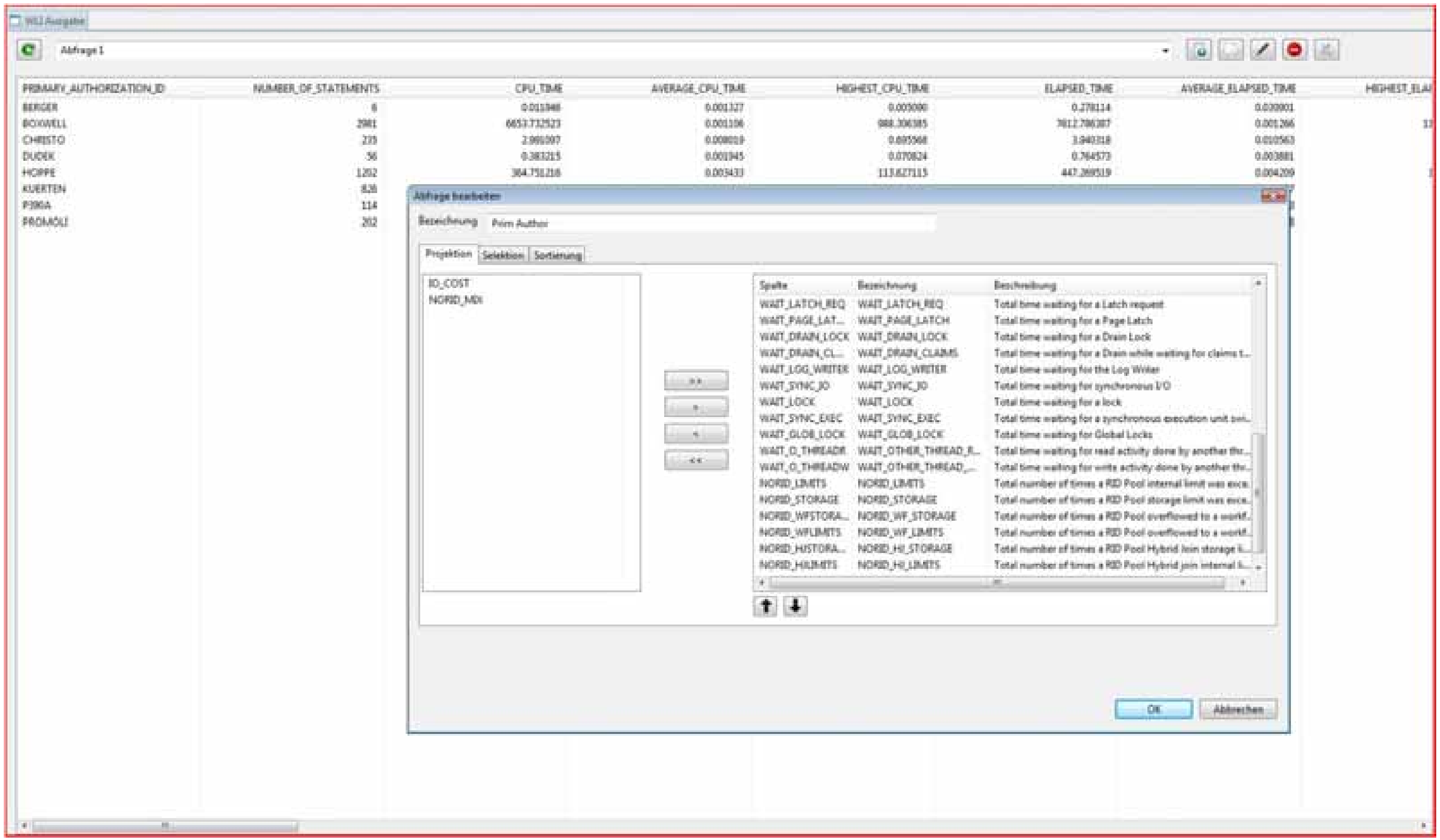
Timings:

Externalizing every 5 mins for 24 hours cost 300 secs cpu

(No EXPLAIN of course in this scenario!)

SQL WorkloadExpert on Trial

Use Case 1 : Application Workload Analysis – determine the resource consumption of users, applications, ...



PRIMARY_AUTHORIZATION_ID	NUMBER_OF_STATEMENTS	CPU_TIME	AVERAGE_CPU_TIME	HIGHEST_CPU_TIME	ELAPSED_TIME	AVERAGE_ELAPSED_TIME	HIGHEST_ELA
BERGER	6	0.011948	0.001327	0.003090	0.278114	0.030001	
BOXWELL	2981	6653.732523	0.001106	988.306385	7812.786387	0.001266	11
CHRISTO	275	2.991007	0.008019	0.695568	3.940318	0.010563	
DUDEK	56	0.383215	0.001945	0.070824	0.764573	0.003881	
HOPPE	1202	384.751216	0.003433	113.627115	447.269519	0.004209	1
KUERTEN	826						
P390A	114						
PROMOLI	202						

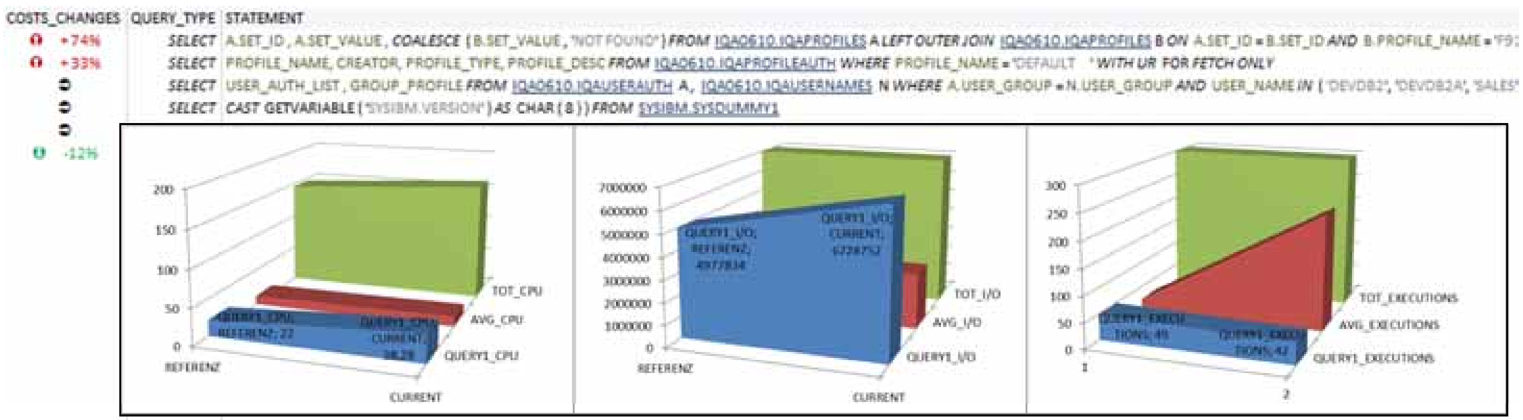
Bezeichnung: Prim Author

IO_COST
NORID_MIX

Spalte	Bezeichnung	Beschreibung
WAIT_LATCH_REQ	WAIT_LATCH_REQ	Total time waiting for a Latch request
WAIT_PAGE_LAT...	WAIT_PAGE_LATCH	Total time waiting for a Page Latch
WAIT_DRAIN_LOCK	WAIT_DRAIN_LOCK	Total time waiting for a Drain Lock
WAIT_DRAIN_CL...	WAIT_DRAIN_CLAIMS	Total time waiting for a Drain while waiting for claims t...
WAIT_LOG_WRITER	WAIT_LOG_WRITER	Total time waiting for the Log Writer
WAIT_SYNC_IO	WAIT_SYNC_IO	Total time waiting for synchronous I/O
WAIT_LOCK	WAIT_LOCK	Total time waiting for a lock
WAIT_SYNC_EXEC	WAIT_SYNC_EXEC	Total time waiting for a synchronous execution unit swi...
WAIT_GLOB_LOCK	WAIT_GLOB_LOCK	Total time waiting for Global Locks
WAIT_O_THREADR	WAIT_OTHER_THREAD_R...	Total time waiting for read activity done by another thr...
WAIT_O_THREADW	WAIT_OTHER_THREAD_...	Total time waiting for write activity done by another thr...
NORID_LIMITS	NORID_LIMITS	Total number of times a RID Pool internal limit was exce...
NORID_STORAGE	NORID_STORAGE	Total number of times a RID Pool storage limit was exce...
NORID_WFSTORA...	NORID_WF_STORAGE	Total number of times a RID Pool overflowed to a workf...
NORID_WFLIMITS	NORID_WF_LIMITS	Total number of times a RID Pool overflowed to a workf...
NORID_HISTORA...	NORID_HI_STORAGE	Total number of times a RID Pool Hybrid Join storage li...
NORID_HILIMITS	NORID_HI_LIMITS	Total number of times a RID Pool Hybrid join internal li...

SQL WorkloadExpert on Trial

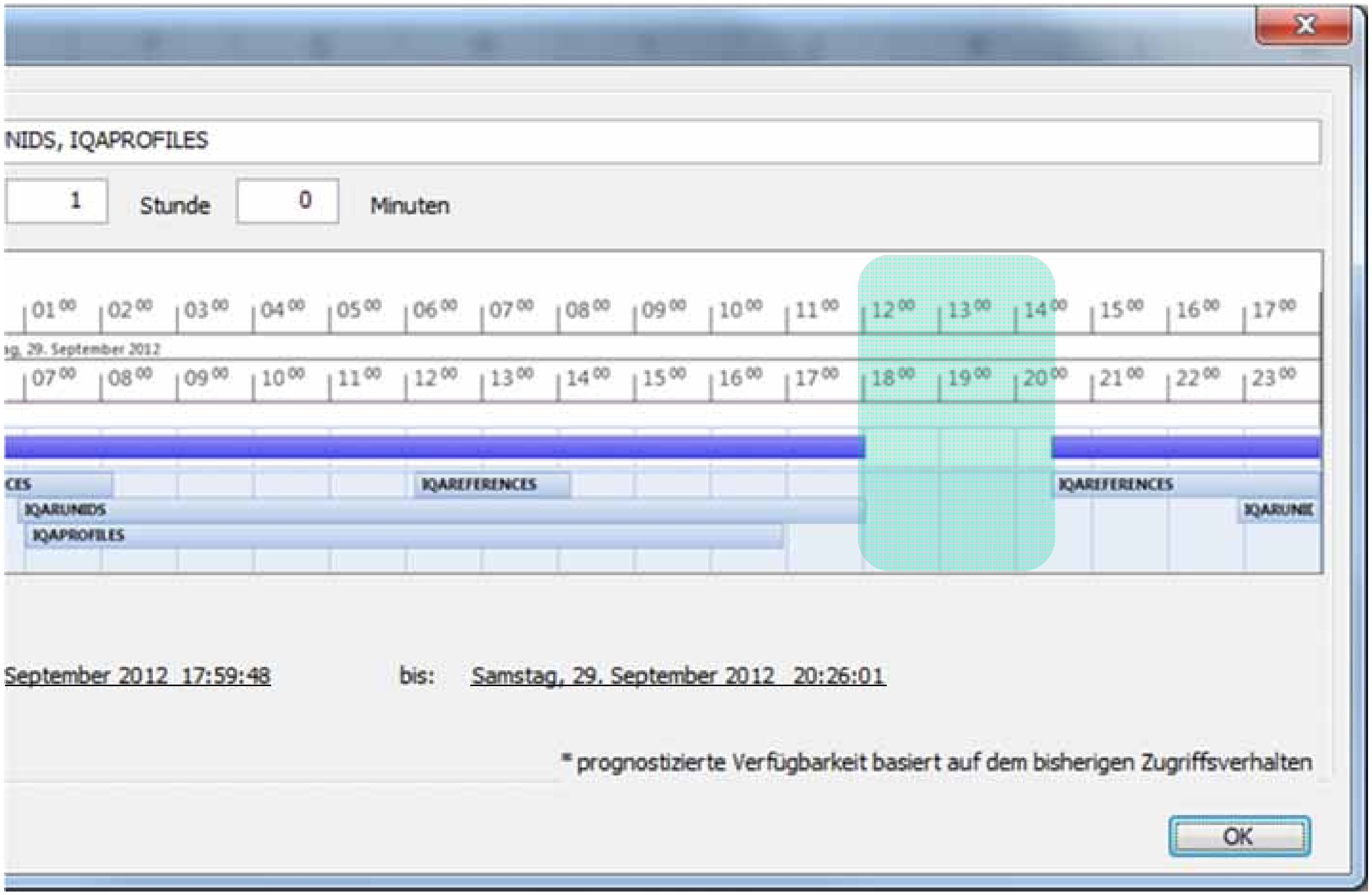
Use Case 2 : Trending of Applications, CPU, I/O, execution



COSTS_CHANGES	QUERY_TYPE	STATEMENT
+74%	SELECT	A.SET_ID , A.SET_VALUE , COALESCE (B.SET_VALUE , 'NOT FOUND') FROM IQA0610.IQAPROFILES A LEFT OUTER JOIN IQA0610.IQAPROFILES B ON A.SET_ID = B.SET_ID AND B.PROFILE_NAME = 'F9:
+33%	SELECT	PROFILE_NAME, CREATOR, PROFILE_TYPE, PROFILE_DESC FROM IQA0610.IQAPROFILEAUTH WHERE PROFILE_NAME = 'DEFAULT ' WITH UR FOR FETCH ONLY
●	SELECT	USER_AUTH_LIST , GROUP_PROFILE FROM IQA0610.IQAUSERAUTH A , IQA0610.IQAUSERNAMES N WHERE A.USER_GROUP = N.USER_GROUP AND USER_NAME IN ('DEVDB2', 'DEVDB2A', 'SALES'
●	SELECT	CAST GETVARIABLE ('SYSIBM.VERSION') AS CHAR (8)) FROM SYSIBM.SYSDUMMY1
●		
-12%		

SQL WorkloadExpert on Trial

Use Case 3 : Object Quiet Times for maintenance (REORG)



NIDS, IQAPROFILES

1 Stunde 0 Minuten

September 2012 17:59:48 bis: Samstag, 29. September 2012 20:26:01

* prognostizierte Verfügbarkeit basiert auf dem bisherigen Zugriffsverhalten

SQL WorkloadExpert on Trial

Use Case 4 : Audit (Parallel user access etc.)

User SMITH manages to run three SQLs at the same time from three different servers in New York, London and Bangalore.

How is this possible? Has there been a password leak?

SQL WorkloadExpert on Trial

Use Case 5: Find never used objects (Plans, Collections, Packages, Tables, MQTs, VIEWs and Indexes)

With a representative timeframe of data (e.g. quarter, year) you gain great insights about objects being used, or not used.

SQL WorkloadExpert on Trial

Use Case 6 : Never executed static SQL

Avoid having DBAs correcting „bad“ SQL in pre-production, or wasting time „correcting“, or even creating/altering indexes tuning SQL that is never executed.

SQL WorkloadExpert on Trial

Use Case 7 : Forecast the possible performance improvements in dynamic SQL by changing literals to parameter markers.

SQL WorkloadExpert on Trial

Use case 8 : Detect DASD problems, determine I/O rates

- The data allows analyzing how long it takes for synchronous I/Os, or calculating I/O throughput
- The data allows alerting when:
    - Any wait time per synchronous IO is over two milliseconds
    - For OLTP any application that has more than one synchronous IO per statement

SQL WorkloadExpert on Trial

Use Case 9 : Analyse bufferpools – hit ratios, VPSEQT-tuning (Virtual Pool Sequential Threshold) random/sequential access

The following values are calculated:

- System and application hit ratios
- Three residency times (system, random page and sequential page)
- Two bufferpool write efficiency measures (page updates per page written and pages written per write I/O)
- VPSEQT
- Bufferpool intensity. Which is a good pointer to use PAGEFIX=YES

Use Case 10: Detect Multi-row fetch candidates

Since DB2 V8 we have multi-row FETCH (as well as INSERT and UPDATE). But the problem is no-one changed the existing programs.

Benchmarking shows savings about 50% for FETCH related CPU.

- Determine top 10 CURSORs that would save most
- Find all non-ROWSET defined cursors and calculate the No. Fetch's / No. of Execution's

SQL WorkloadExpert on Trial

Use Case 11: SQL WLX KPIs – Discover the background noise as well as exceptions

Graphs shows the averaged background work and also the highest peaks for CPU and IO and Elapsed etc.

Use Case 12: Detect SELECT only tables

Objects are determined that *only* have SELECT SQLs running against them to check if these

- typically look-up
- cross reference style tables
- should be moved to their „own“ Bufferpool
- etc.

SQL WorkloadExpert on Trial

Use Case 13: Detect long delays

“Scroll back in time” to the point of a delay to check inflight SQL that references any of the same objects in the current SQL.

SQL WorkloadExpert on Trial

Use Case 14: Detect deadlock, lock escalation and index page splits

By using three other IFCIDS we trap any, or all of the above occurrences to enable full detection of:

- DEADLOCK reasons (all holders)
- Lock escalation reasons

For index page splits:

1. Find the right candidates
2. Redesign the index, or change it’s definition
3. REORG to stop the splits causing problems

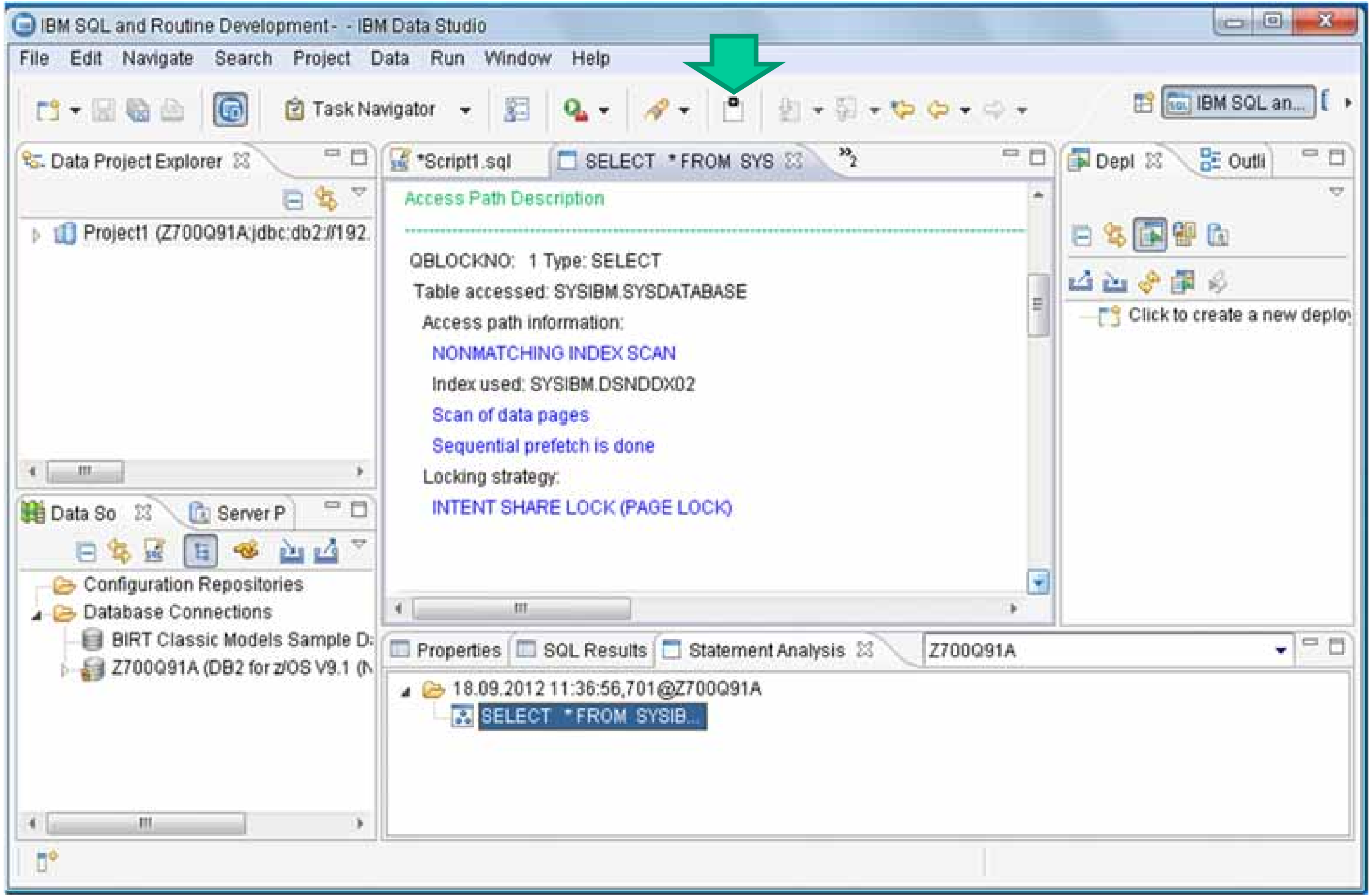
SQL WorkloadExpert on Trial

Use Case 15: Multi-snap

Provides a very granular view of the resource consumption of SQL running on the system.

SQL WorkloadExpert on Trial

Use Case 16: SPX (SQLPerformanceExpert) GUI link

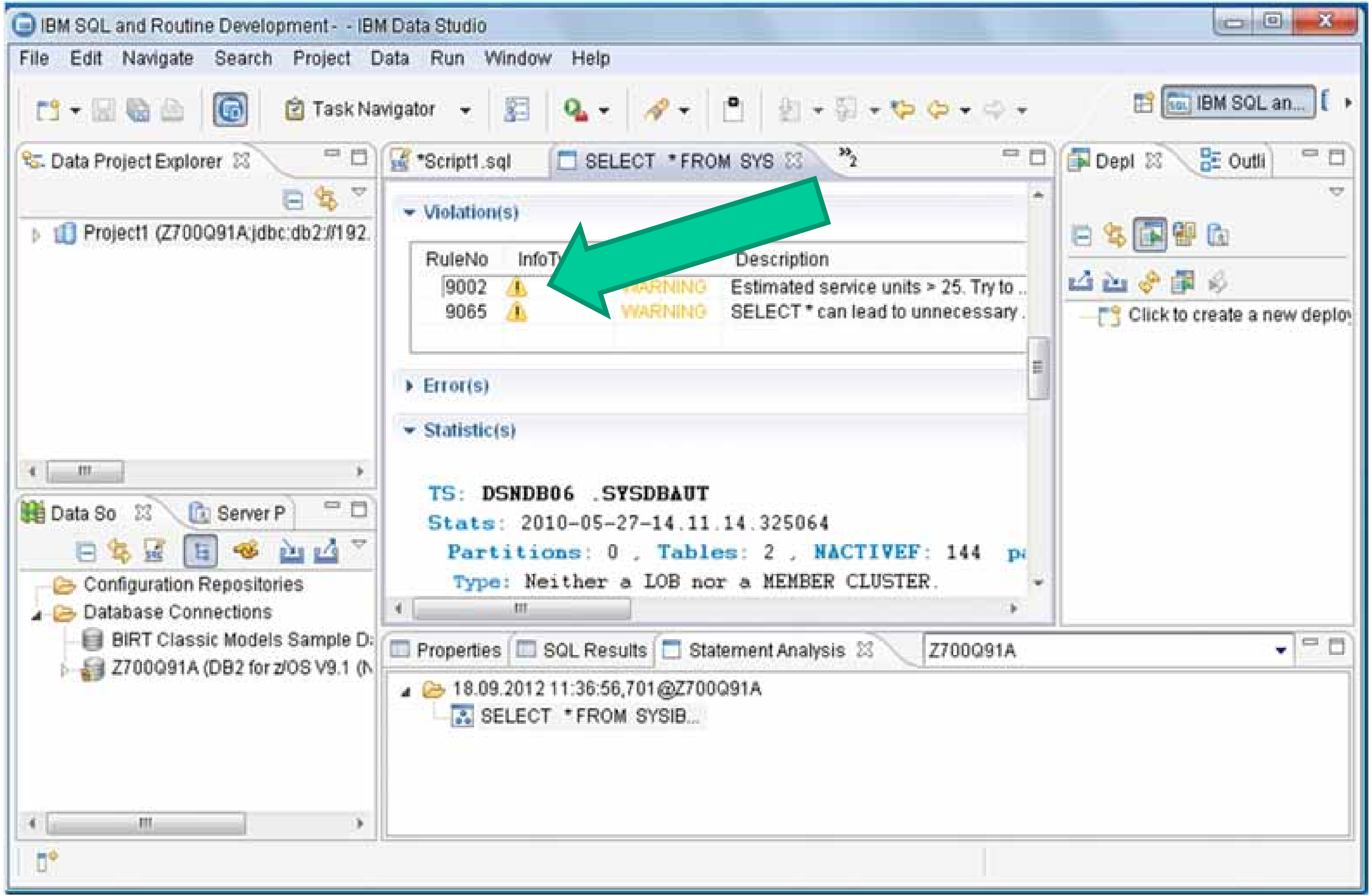


Access Path Description

QBLOCKNO: 1 Type: SELECT
Table accessed: SYSIBM.SYSDATABASE
Access path information:
NONMATCHING INDEX SCAN
Index used: SYSIBM.DSNDDX02
Scan of data pages
Sequential prefetch is done
Locking strategy:
INTENT SHARE LOCK (PAGE LOCK)

SQL WorkloadExpert on Trial

Use Case 16: SPX (SQLPerformanceExpert) GUI link



IBM SQL and Routine Development - - IBM Data Studio
File Edit Navigate Search Project Data Run Window Help
Task Navigator
IBM SQL an...
Data Project Explorer
Project1 (Z700Q91A:jdbc:db2://192.
*Script1.sql
SELECT *FROM SYS
Depl
Outli
Click to create a new deploy
Data So
Server P
Configuration Repositories
Database Connections
BIRT Classic Models Sample D:
Z700Q91A (DB2 for z/OS V9.1 (N
Properties
SQL Results
Statement Analysis
Z700Q91A

Violation(s)

RuleNo	InfoTy		Description
9002	⚠	WARNING	Estimated service units > 25. Try to ..
9065	⚠	WARNING	SELECT * can lead to unnecessary .

Error(s)

Statistic(s)

```
TS: DSNDB06 .SYSDBAUT
Stats: 2010-05-27-14.11.14.325064
  Partitions: 0 , Tables: 2 , NACTIVEF: 144  pa
   Type: Neither a LOB nor a MEMBER CLUSTER.
```

```
18.09.2012 11:36:56,701@Z700Q91A
  SELECT *FROM SYSIB...
```

SQL WorkloadExpert on Trial

Use Case 17: REORG Detector & Suppressor

Detect and verifies REORGs and there effect on performance, I/O, …

Interfaces with database maintenance solutions like RealTime DBAEpert for OnDemand REORGs

SQL WorkloadExpert on Trial

Use Case 18: Detect Eager vs. Lazy loader (JPA – Java Persistence API)

Various JPAs are used and some have Eager Loading or Lazy Loading as an option. Eager instantiates everything for everywhere - which can of course be overkill! Lazy delays the instantiation until used.

Making recommendations of whether or not eager is better than lazy helps managing Java workload more efficiently.

SQL WorkloadExpert on Trial

Use Case 19: Determine object usage by application, including service naming by object (enhanced RECOVER prioritization)

Allows to see the KPIs for all of the tables with a free form text interface to define

- What is this object?
- Which service uses this object?
- How important is this object?

SQL WorkloadExpert on Trial

Use Case 20: Exploit offline workload repository databases

Exploits DB2 LUW, or DB2 Express-C to hold an unloaded version of the z/OS WLX workload repository for offline data mining etc.

Enabling to even keep years of data on cheap PC disks.

SQL WorkloadExpert – Eclipse support

Codename	Date	Platform version	
N/A	21 June 2004	3.0	Old versions
N/A	28 June 2005	3.1	
Callisto	30 June 2006	3.2	
Europa	29 June 2007	3.3	
Ganymede	25 June 2008	3.4	
Galileo	24 June 2009	3.5	
Helios	23 June 2010	3.6	Older versions, still supported
Indigo	22 June 2011	3.7	
Juno	27 June 2012	4.2	Latest version
Kepler	26 June 2013 (planned)	4.3	Future release

SQL WorkloadExpert

- Collects all static and dynamic SQL cost effectively
- Aggregates, condenses and EXPLAINs the data dynamically
- Compares 2, or more timefframes for resource trending of (CPU_Time, Elapsed Time, I/O)
- Establish a SQL Workload Warehouse
- Comes with more than 20 predefined use cases
- Eclipse, DataStudio, or Rational integration as presentation layer

→ Increase your DB2 Workload insight