Expert Stored Procedure Monitoring, Analysis and Tuning on System z

Steve Fafard, Product Manager, IBM
OMEGAMON XE for DB2 Performance Expert on z/OS

August 16, 2013
13824

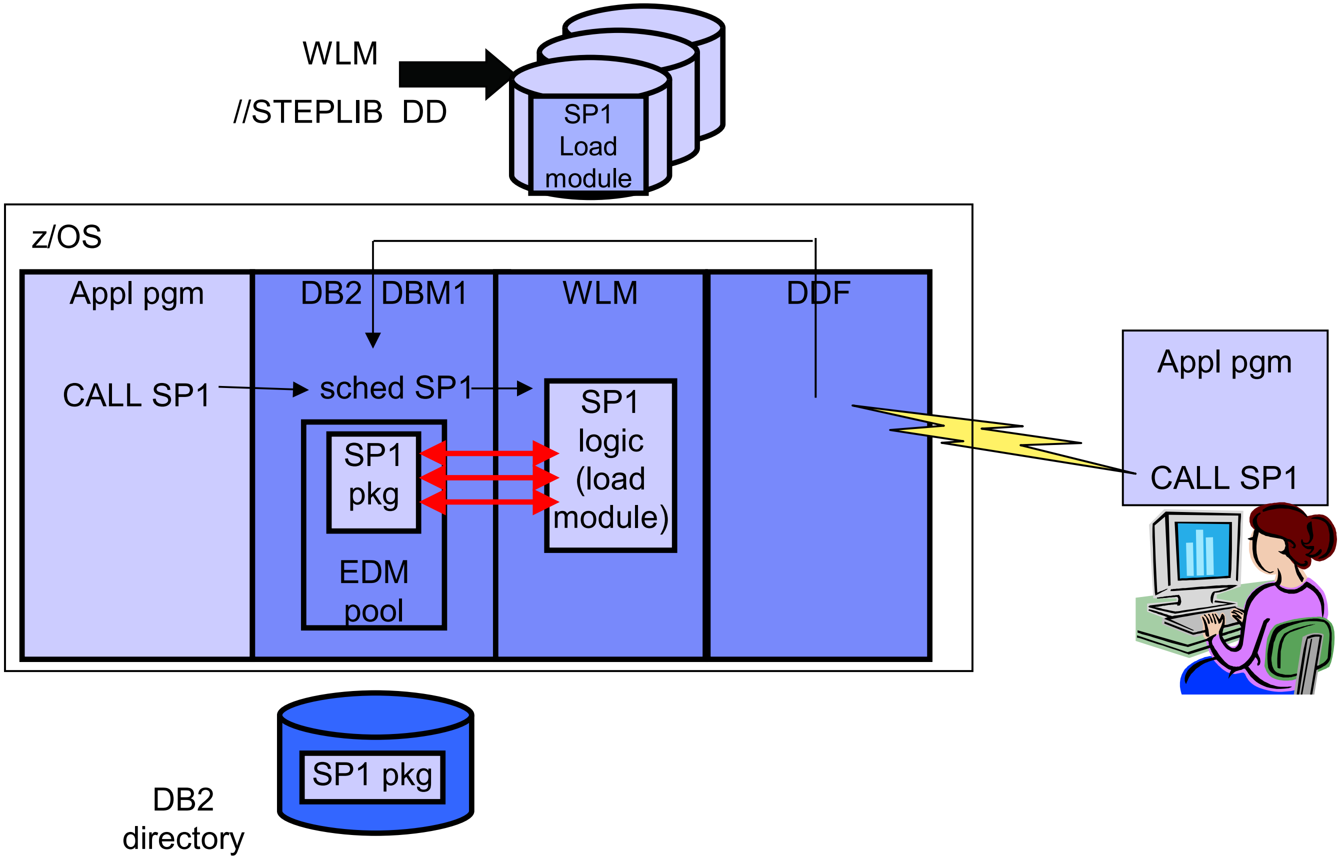
Agenda

- What are stored procedures?
  - Benefits of stored procedures
  - Stored procedure analysis – Issues and solutions
- Monitoring stored procedures using OMEGAMON DB2 Performance Expert
- Isolating and tuning poorly performing stored procedures

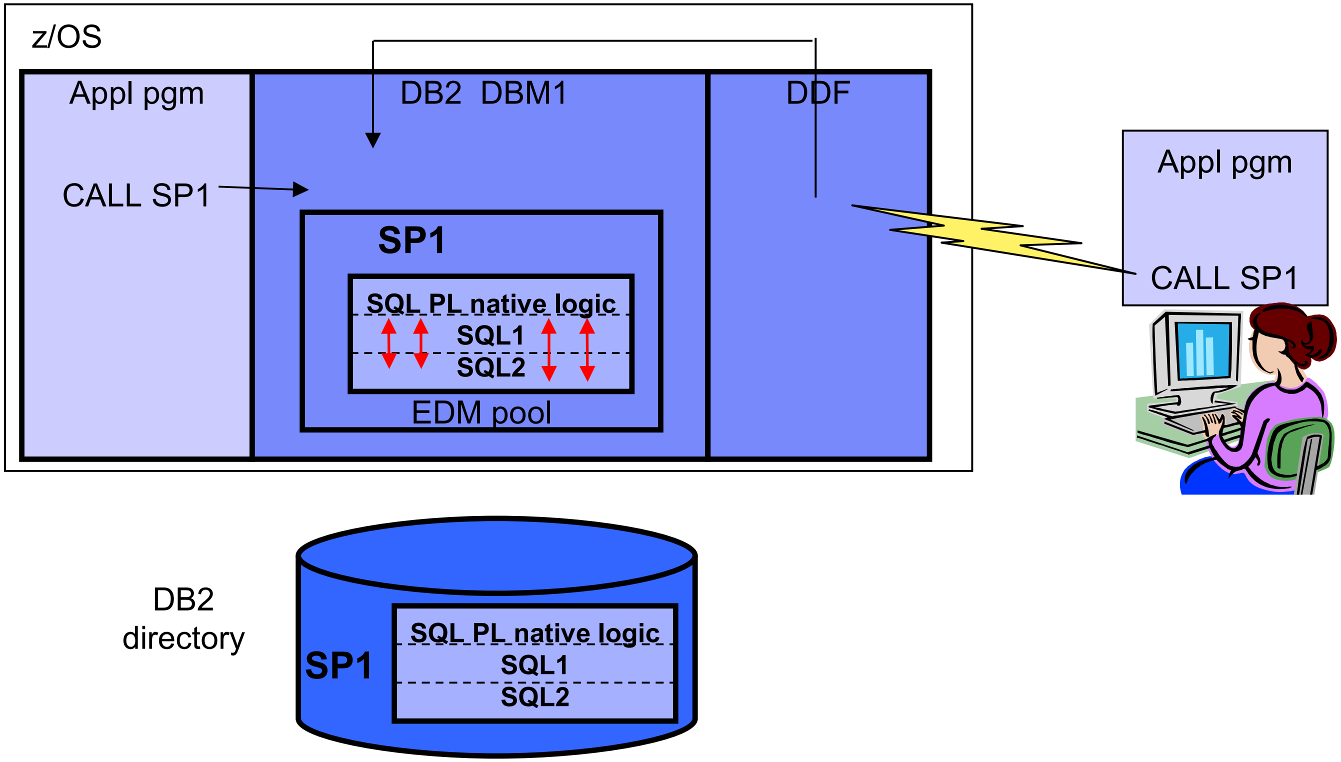
What are stored procedures?

- A stored procedure is a user-written program that can be called by an application with an SQL CALL statement.
- It is a compiled program that is stored at a DB2 server
- It can execute business logic and SQL statements
- Stored procedure types
  - External high level language procedures COBOL, PL/I, C, C++, Assembler, REXX, and Java
  - External SQL procedures
  - Native SQL procedures introduced by DB2 9 for z/OS

DB2 z/OS stored procedure processing (External)



Native SQL procedure processing (Internal)



Programming benefits of stored procedures

- Modularity in application development
  - Data will be processed always in a consistent way according to the rules defined in the stored procedure
- Enforcement of business rules
  - Use stored procedures to define business rules that are common to several applications.
  - Can be an alternative to using constraints and triggers.
- Improved application security
  - Sensitive business logic runs on the DB2 server
  - End users are authorized to execute a stored procedure, and do not need table privileges. (Similar to static authorization model.)
- Application integration solutions
  - Can access non-DB2 resources, e.g. VSAM files, MQ queues, IMS or CICS transactions
  - Stored procedures can have access to commands that run only on the server.

Total cost of ownership benefits of stored procedures

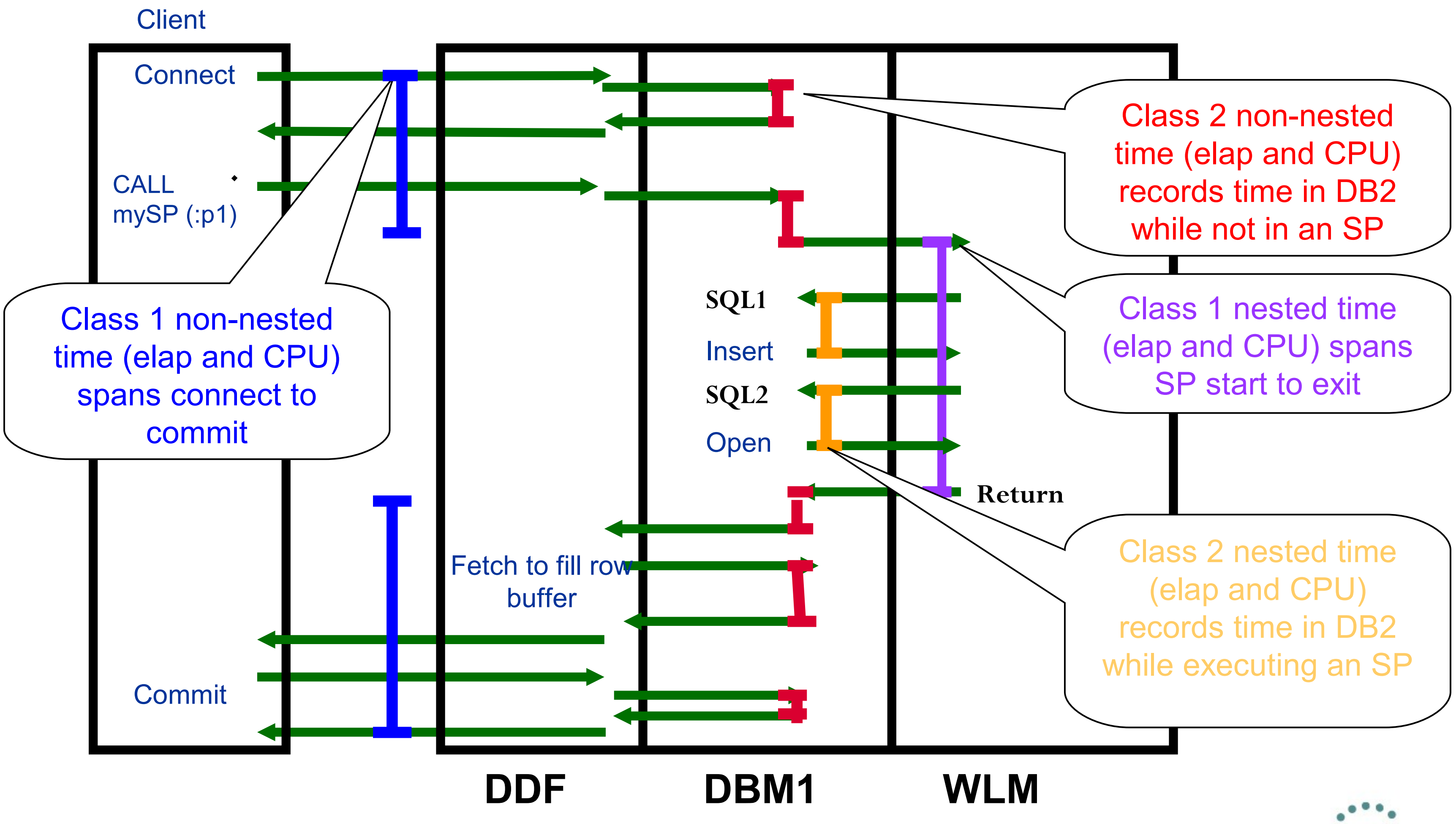
- Reduced network traffic for distributed applications
  - Grouping SQL statements into a stored procedure results in two trips across the network for each group of statement, resulting in better performance for applications
- Cost of ownership reduction
  - If stored procedure is called from distributed client via DRDA, a portion is eligible for zIIP redirect.
    - Including: Call statement processing; Result set processing; Commit processing
  - Stored procedures written in Java can take advantage of zAAP engines
  - Native SQL procedures run as enclave SRB in DBM1 address space and the stored procedure execution itself is zIIP off-loadable with DB2 9 for z/OS.
  - For WLM managed stored procedures:
    - SQL processing runs under a TCB hence not eligible for zIIP redirect
- As of now, there is NO performance benefit for calling a stored procedure from a local application

Stored procedure language / API CPU Cost comparison - Update

- IRWW workload (OLTP workload consisting of 7 transactions)
- Called from distributed JCC type 4 client

Language / API	Base CPU / Tran Cost	Billable CPU / Tran Cost after zIIP and / or zAAP redirect
COBOL stored procedure	1X (BASE)	0.80x (Some zIIP)
C stored procedure	1.02x	0.82x (Some zIIP)
SQLJ stored procedure	2.01x	1.11x (zAAP+ some zIIP)
JDBC stored procedure	2.97x	1.84x (zAAP+ some zIIP)
Native SQL stored procedure	1.09x	0.59x (Significant zIIP)

Performance reporting – External stored procedure



Class 2 non-nested time (elap and CPU) records time in DB2 while not in an SP

Class 1 nested time (elap and CPU) spans SP start to exit

Class 1 non-nested time (elap and CPU) spans connect to commit

Class 2 nested time (elap and CPU) records time in DB2 while executing an SP

External stored procedure performance summary – plan level

- DB2 Accounting class 1 and 2 needed (3 is recommended)

```
AVERAGE         APPL(CL.1)  DB2 (CL.2)
------------    ----------  ----------
ELAPSED TIME      0.003212    0.002575
 NONNESTED        0.000714    0.000694
 STORED PROC      0.002498    0.001881
 UDF              0.000000    0.000000
 TRIGGER          0.000000    0.000000
CP CPU TIME       0.000715    0.000654
 AGENT            0.000715    0.000654
  NONNESTED       0.000149    0.000129
  STORED PRC      0.000567    0.000525
  UDF             0.000000    0.000000
  TRIGGER         0.000000    0.000000
 PAR.TASKS        0.000000    0.000000
```

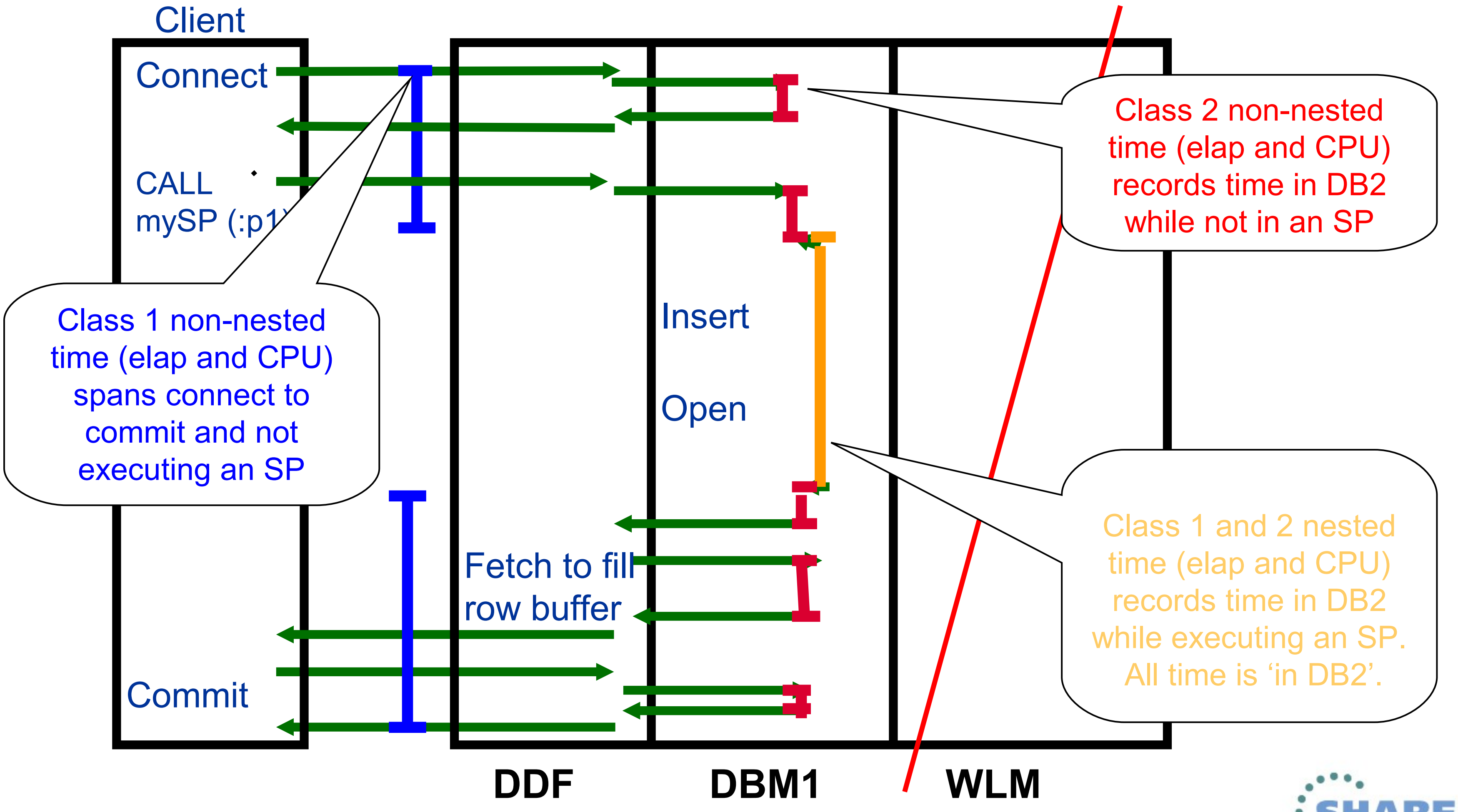
Class 1 non-nested time (ET & CPU)

Class 2 non-nested time (ET & CPU)

Class 1 nested time (ET & CPU)

Class 2 nested time (ET & CPU)

Performance reporting – Native SQL stored procedure



Class 1 non-nested time (elap and CPU) spans connect to commit and not executing an SP

Class 2 non-nested time (elap and CPU) records time in DB2 while not in an SP

Class 1 and 2 nested time (elap and CPU) records time in DB2 while executing an SP. All time is 'in DB2'.

Native SQL stored procedure performance summary - plan-level

- DB2 Accounting class 1 and 2 needed (3 is recommended)

```
AVERAGE          APPL(CL.1)  DB2 (CL.2)
------------     ----------  ----------
ELAPSED TIME       0.004834    0.002789
 NONNESTED         0.002819    0.000774
 STORED PROC       0.002015    0.002015
 UDF               0.000000    0.000000
 TRIGGER           0.000000    0.000000

CP CPU TIME        0.000963    0.000909
 AGENT             0.000963    0.000909
  NONNESTED        0.000198    0.000143
  STORED PRC       0.000765    0.000765
  UDF              0.000000    0.000000
  TRIGGER          0.000000    0.000000
 PAR.TASKS         0.000000    0.000000
```

CL1 and CL2 will always be equal!

CL1 and CL2 will always be equal!

Stored procedure detail reporting - package level reporting

- Accounting class 7 and/or 8 needed
- SYSSTAT package contains time for CALL statement, result set processing, SET special registers, and VALUES statements for LOB handling

```
SYSSTAT              VALUE                 SYSSTAT                    TIMES
-------------------  ------------------    ------------------  ------------
TYPE                 PACKAGE               ELAP-CL7 TIME-AVG       0.000387
                                           CP CPU TIME             0.000072
LOCATION             DSND91B                 AGENT                 0.000072
COLLECTION ID        NULLID                  PAR.TASKS             0.000000
PROGRAM NAME         SYSSTAT               SE CPU TIME             0.000000

NSQLNEW              VALUE                 NSQLNEW                    TIMES
-------------------  ------------------    ------------------  ------------
TYPE                 PACKAGE               ELAP-CL7 TIME-AVG       0.004751
                                           CP CPU TIME             0.001667
LOCATION             DSND91B                 AGENT                 0.001667
COLLECTION ID        USRT001                 PAR.TASKS             0.000000
PROGRAM NAME         NSQLNEW               SE CPU TIME             0.000000
```

Issues with plan and package ;evel stored procedure analysis

- Multiple stored procedures called in a transaction are summed at the plan level. By definition this affects the analysis of nested SPs.

- Package level analysis can be difficult if a stored procedure executes different paths and SQL based on parameters. How do you differentiate between the invocations?

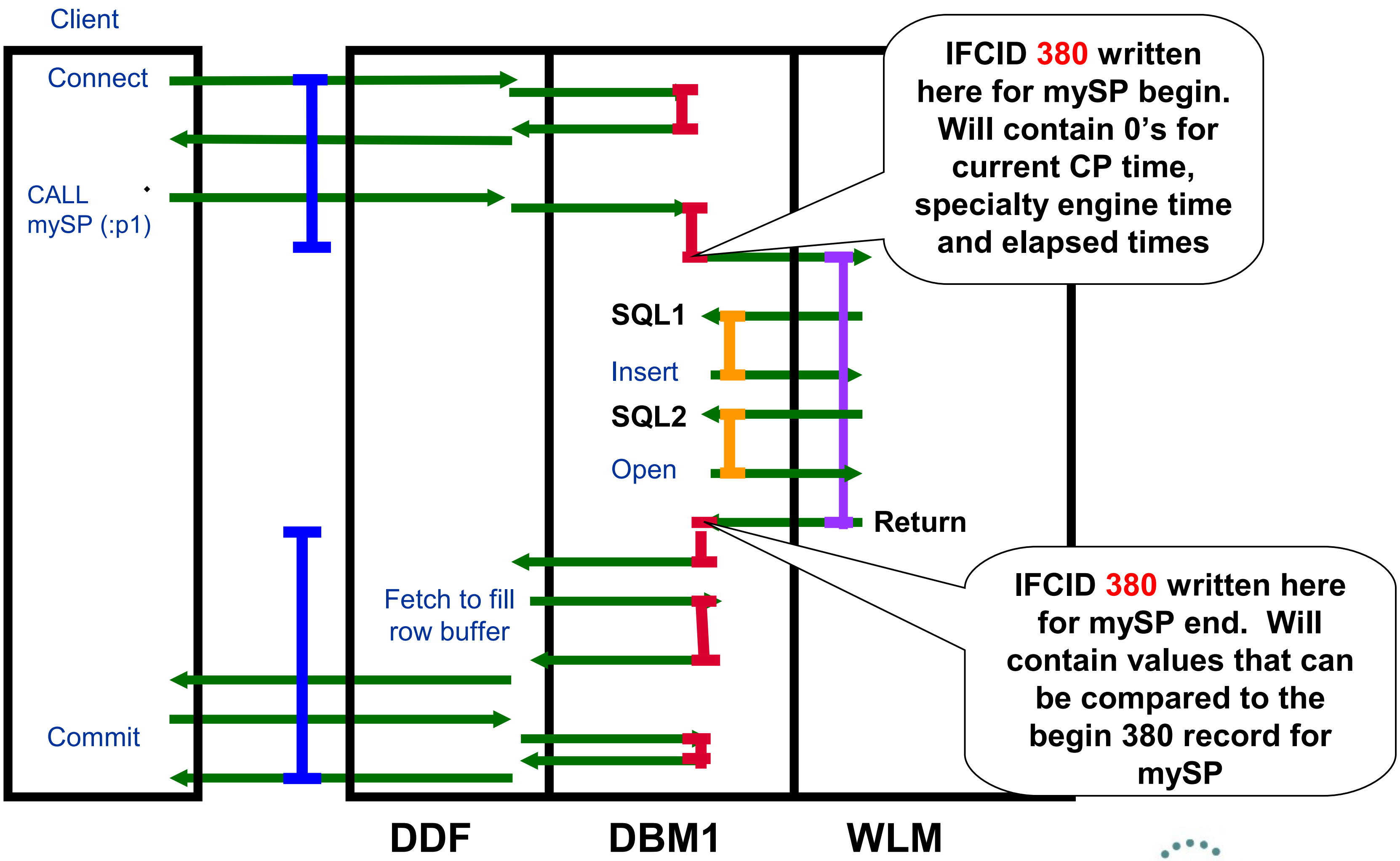
- Package level analysis does not apply to stored procedures that do not execute SQL

Enhanced instrumentation for stored procedure performance analysis

- PM53243 (DB2 10) New IFCIDs 380 and 381 are created for stored procedure and User-Defined Function detail respectively. These records:
  - Identify the stored procedure or UDF beginning or end
  - Include the current CP, specialty engine, and elapsed time details for nested activity

- These record can be used to determine the CP, specialty engine, and elapsed time for a given stored procedure or UDF invocation

Enhanced instrumentation for stored procedure performance analysis



IFCID 380 written here for mySP begin. Will contain 0's for current CP time, specialty engine time and elapsed times

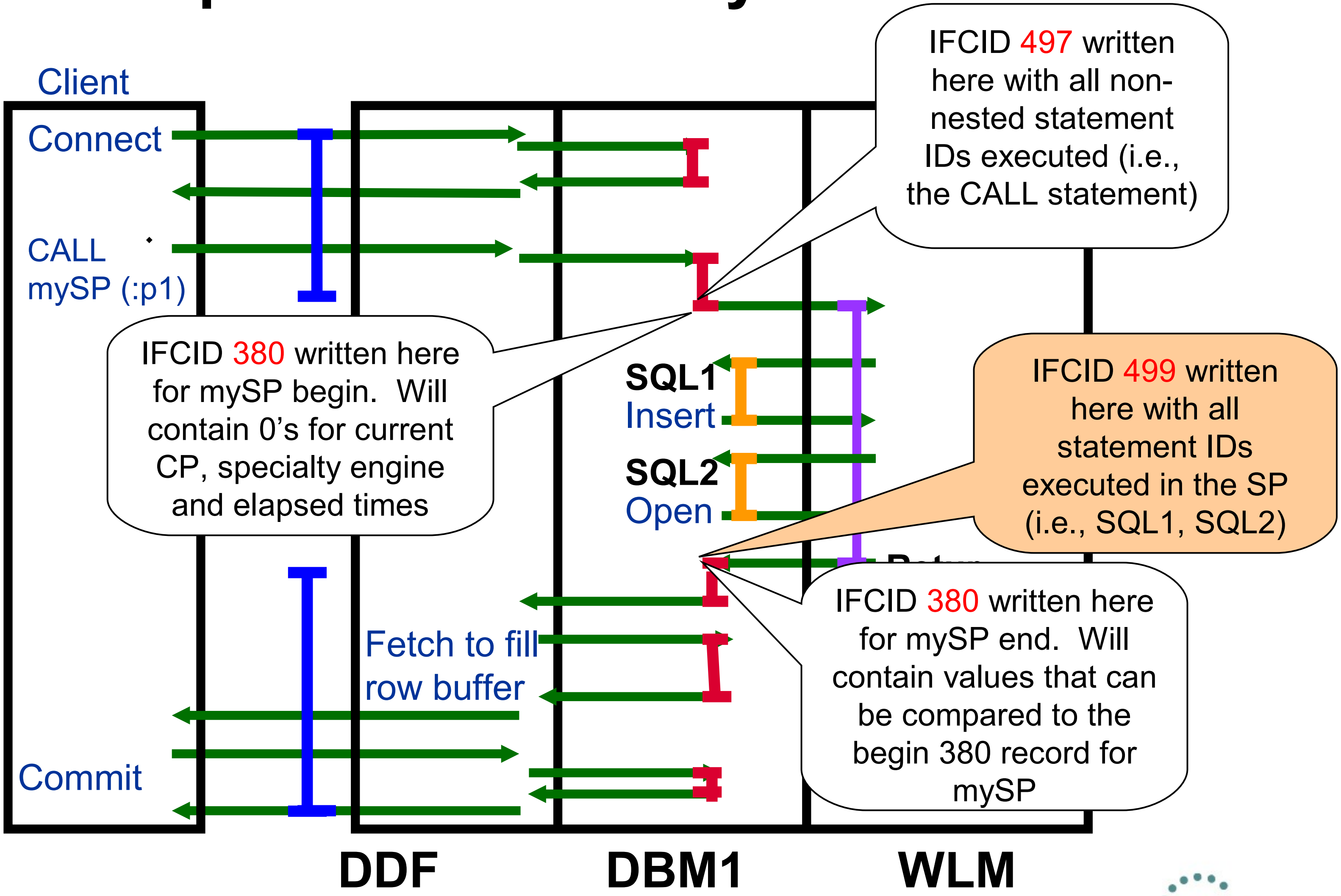
IFCID 380 written here for mySP end. Will contain values that can be compared to the begin 380 record for mySP

Enhanced instrumentation for stored procedure performance analysis

- Additionally PM53243 (DB2 10) added IFCID 497, 498, 499 for SQL drill down analysis.
  - These records contain the dynamic or static statement IDs for non-nested, UDF, and SP work respectively.

- The statement IDs can be correlated to IFCID 316 dynamic statement or IFCID 401 static statement cache data.

Enhanced instrumentation for stored procedure performance analysis



IFCID 497 written here with all non-nested statement IDs executed (i.e., the CALL statement)

IFCID 380 written here for mySP begin. Will contain 0's for current CP, specialty engine and elapsed times

IFCID 499 written here with all statement IDs executed in the SP (i.e., SQL1, SQL2)

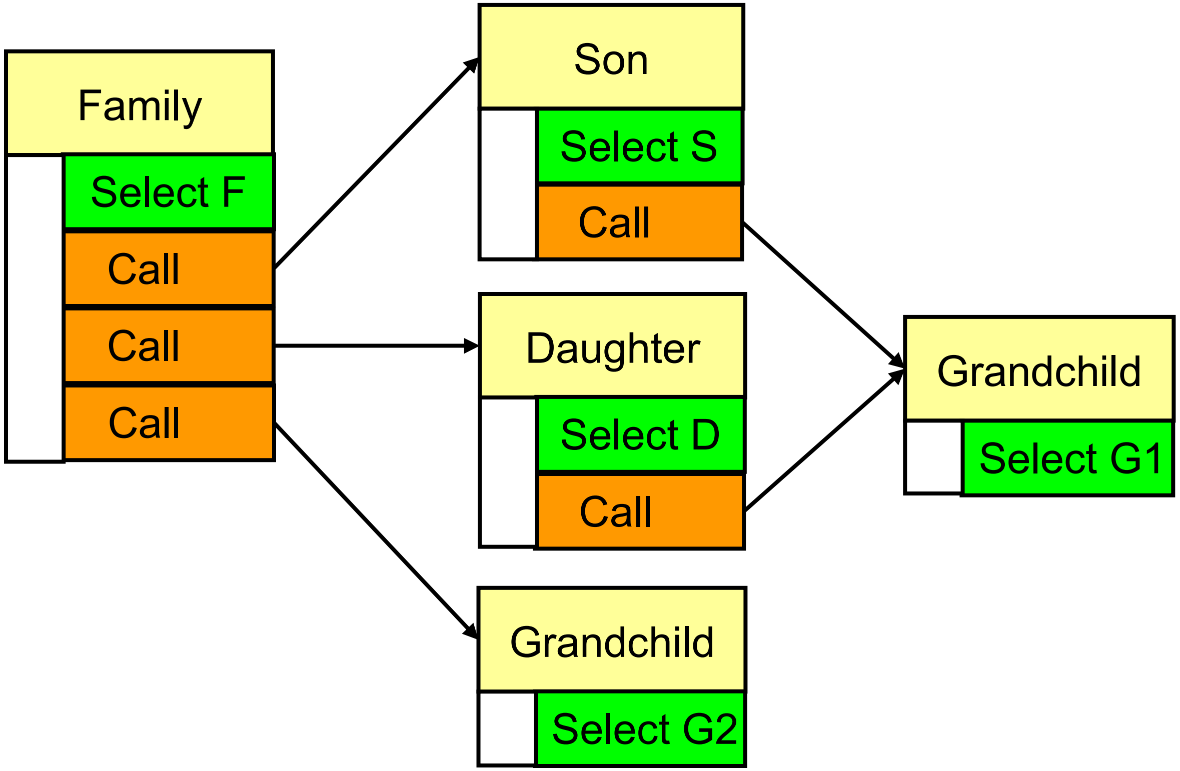
IFCID 380 written here for mySP end. Will contain values that can be compared to the begin 380 record for mySP

Monitoring stored procedures with OMEGAMON XE for DB2 Performance Expert

- These DB2 instrumentation records for stored procedures are evaluated by OMEGAMON, aggregated on a system level and returned to the repository server engine.
- OMEGAMON processing includes the sequencing logic and the calculation of elapsed times for the different accounting class times written in the IFI records as timestamps, considering nesting as well.
- In parallel the IFCID 316/401 data for the Statement Caches is collected and a correlation to the executed stored procedure statements via IFCID 499 is made.

Using the OMEGAMON DB2 Performance Expert web console to analyze stored procedures

- Workload:

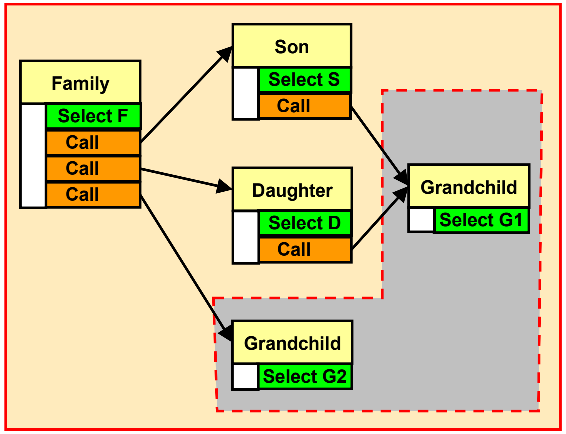


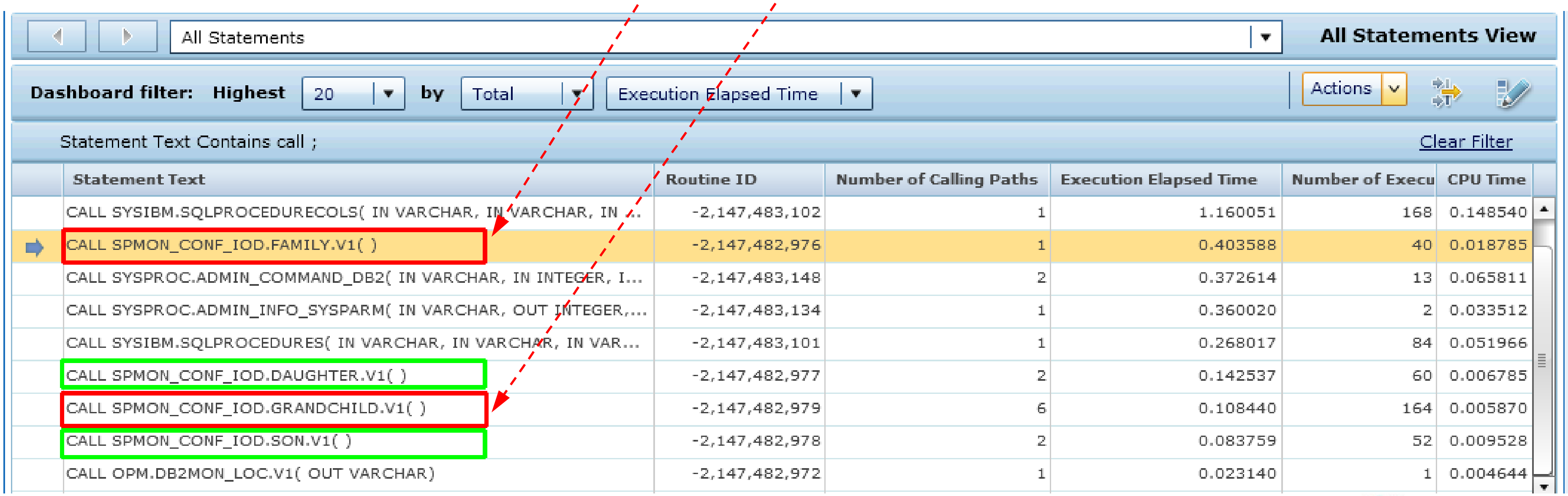
SQL Dashboard – aggregation by ROUTINEID

- Workload at SQL dashboard ("All statements" view) executed in the selected time period (time slider), valid for all subsequent views

∑ of Family

∑ of Grandchild



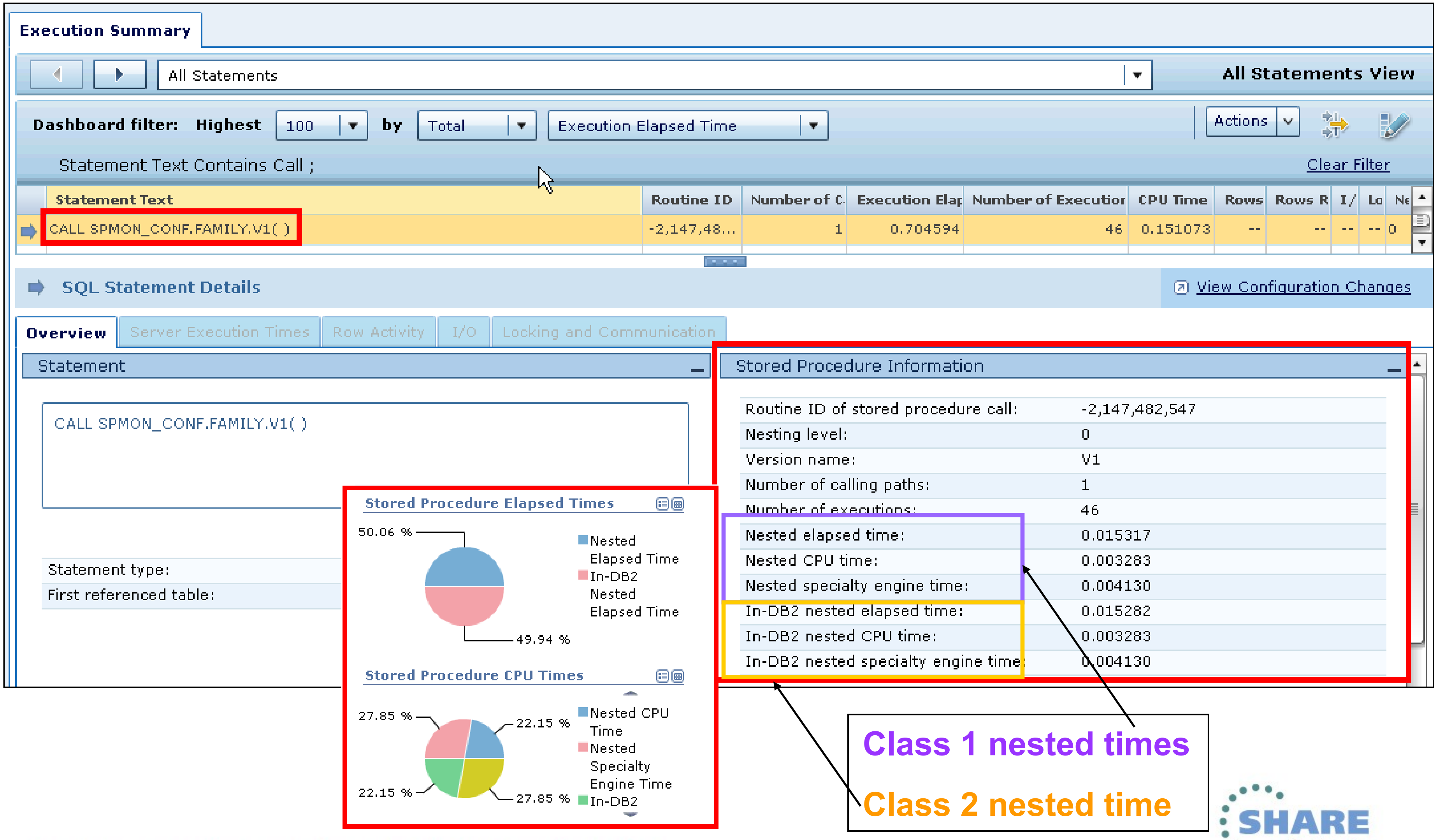


Dashboard filter: Highest 20 by Total Execution Elapsed Time

Statement Text Contains call ;

	Statement Text	Routine ID	Number of Calling Paths	Execution Elapsed Time	Number of Execu	CPU Time
	CALL SYSIBM.SQLPROCEDURECOLS(IN VARCHAR, IN VARCHAR, IN ...	-2,147,483,102	1	1.160051	168	0.148540
➡	CALL SPMON_CONF_IOD.FAMILY.V1( )	-2,147,482,976	1	0.403588	40	0.018785
	CALL SYSPROC.ADMIN_COMMAND_DB2(IN VARCHAR, IN INTEGER, I...	-2,147,483,148	2	0.372614	13	0.065811
	CALL SYSPROC.ADMIN_INFO_SYSPARM(IN VARCHAR, OUT INTEGER,...	-2,147,483,134	1	0.360020	2	0.033512
	CALL SYSIBM.SQLPROCEDURES(IN VARCHAR, IN VARCHAR, IN VAR...	-2,147,483,101	1	0.268017	84	0.051966
	CALL SPMON_CONF_IOD.DAUGHTER.V1( )	-2,147,482,977	2	0.142537	60	0.006785
	CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	-2,147,482,979	6	0.108440	164	0.005870
	CALL SPMON_CONF_IOD.SON.V1( )	-2,147,482,978	2	0.083759	52	0.009528
	CALL OPM.DB2MON_LOC.V1(OUT VARCHAR)	-2,147,482,972	1	0.023140	1	0.004644

Showing stored procedure details



Execution Summary

All Statements

All Statements View

Dashboard filter: Highest 100 by Total Execution Elapsed Time

Actions

Statement Text Contains Call ;

Clear Filter

Statement Text	Routine ID	Number of C	Execution Elap	Number of Executior	CPU Time	Rows	Rows R	I/	Lo	Ne
CALL SPMON_CONF.FAMILY.V1( )	-2,147,48...	1	0.704594	46	0.151073	--	--	--	--	0

SQL Statement Details

View Configuration Changes

Overview | Server Execution Times | Row Activity | I/O | Locking and Communication

Statement

CALL SPMON_CONF.FAMILY.V1( )

Statement type:

First referenced table:

Stored Procedure Information

Routine ID of stored procedure call:	-2,147,482,547
Nesting level:	0
Version name:	V1
Number of calling paths:	1
Number of executions:	46
Nested elapsed time:	0.015317
Nested CPU time:	0.003283
Nested specialty engine time:	0.004130
In-DB2 nested elapsed time:	0.015282
In-DB2 nested CPU time:	0.003283
In-DB2 nested specialty engine time:	0.004130

Stored Procedure Elapsed Times

Stored Procedure CPU Times

Class 1 nested times

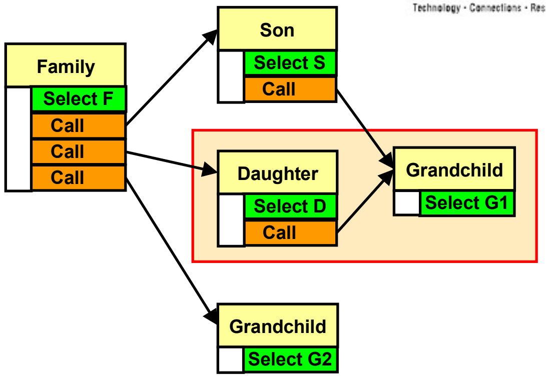
Class 2 nested time

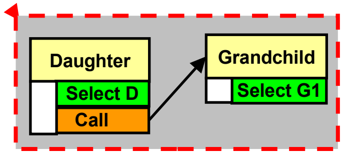
Showing the calling paths of SPs (1/2)

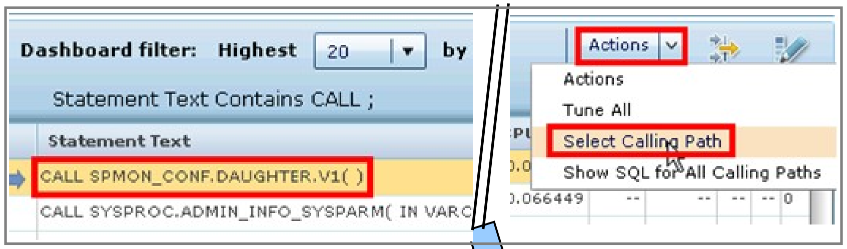
- Select Calling Path for Daughter

∑ of Daughter(1) called by Family(0)

∑ of Daughter(0)

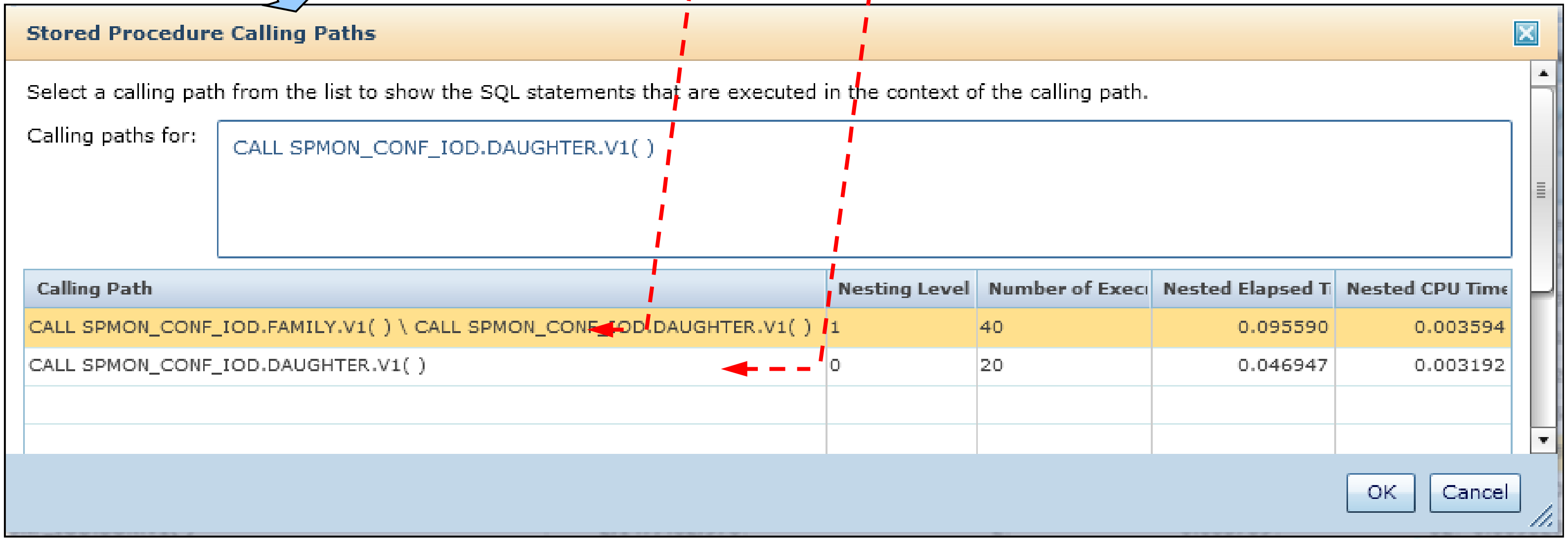





Dashboard filter: Highest 20 by

Statement Text Contains CALL ;

Statement Text					
CALL SPMON_CONF.DAUGHTER.V1( )	0.0				
CALL SYSPROC.ADMIN_INFO_SYSPARM(IN VARC	0.066449	--	--	--	-- 0


Stored Procedure Calling Paths

Select a calling path from the list to show the SQL statements that are executed in the context of the calling path.

Calling paths for: CALL SPMON_CONF_IOD.DAUGHTER.V1( )

Calling Path	Nesting Level	Number of Execu	Nested Elapsed Ti	Nested CPU Time
CALL SPMON_CONF_IOD.FAMILY.V1( ) \ CALL SPMON_CONF_IOD.DAUGHTER.V1( )	1	40	0.095590	0.003594
CALL SPMON_CONF_IOD.DAUGHTER.V1( )	0	20	0.046947	0.003192

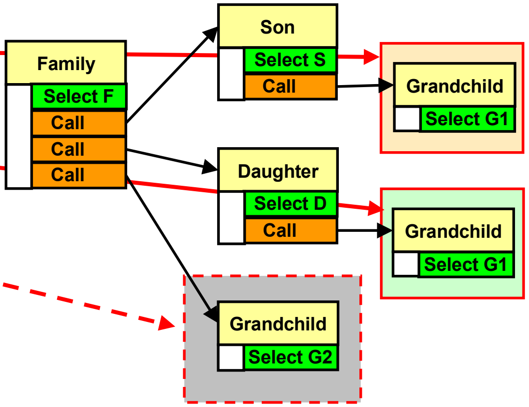
Showing the calling paths of SPs (2/2)

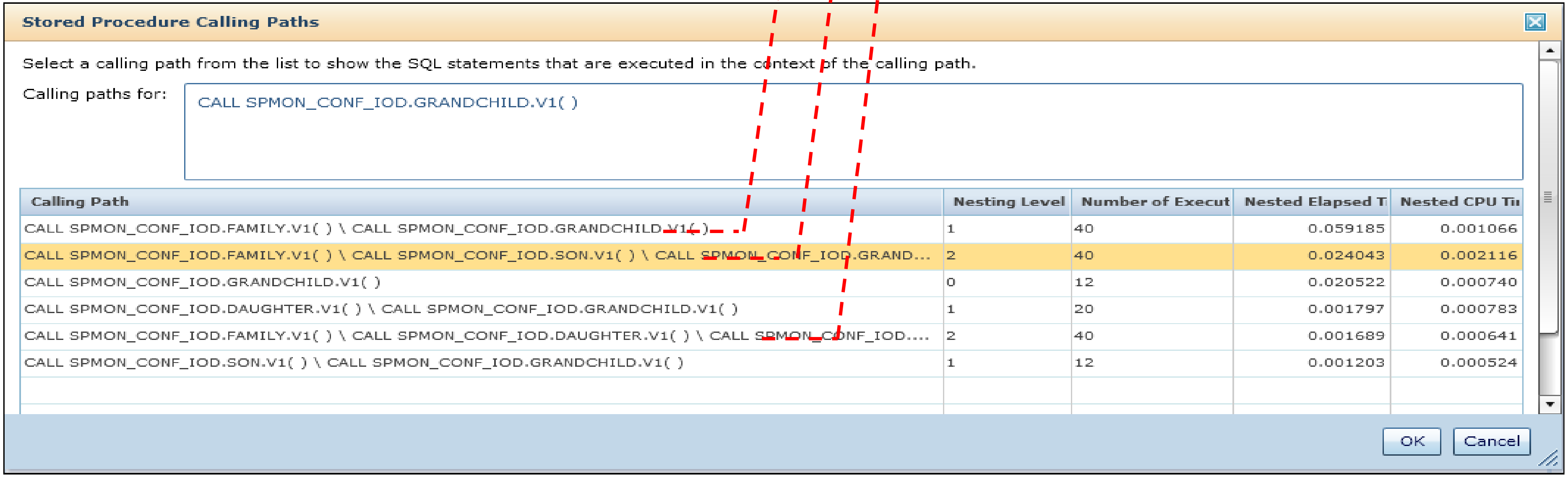
- Select Calling Path for Grandchild

∑ of Grandchild(2) called by Family(0)+Son(1)

∑ of Granchild(2) called by Family (0)+Daughter(1)

∑ of Grandchild(1) called by Family (0)





Stored Procedure Calling Paths

Select a calling path from the list to show the SQL statements that are executed in the context of the calling path.

Calling paths for: CALL SPMON_CONF_IOD.GRANDCHILD.V1( )

Calling Path	Nesting Level	Number of Execut	Nested Elapsed T	Nested CPU Ti
CALL SPMON_CONF_IOD.FAMILY.V1( ) \ CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	1	40	0.059185	0.001066
CALL SPMON_CONF_IOD.FAMILY.V1( ) \ CALL SPMON_CONF_IOD.SON.V1( ) \ CALL SPMON_CONF_IOD.GRAND...	2	40	0.024043	0.002116
CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	0	12	0.020522	0.000740
CALL SPMON_CONF_IOD.DAUGHTER.V1( ) \ CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	1	20	0.001797	0.000783
CALL SPMON_CONF_IOD.FAMILY.V1( ) \ CALL SPMON_CONF_IOD.DAUGHTER.V1( ) \ CALL SPMON_CONF_IOD....	2	40	0.001689	0.000641
CALL SPMON_CONF_IOD.SON.V1( ) \ CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	1	12	0.001203	0.000524

Show SQL executed by a SP (1/2)

- Action: Show SQL for This Calling Path

Select of Family (0) shows

SELECT F *

∑ of Call Son(1) called by Family(0)

∑ of Call Daughter(1) called by Family(0)

∑ of Call Grandchild(1) called by Family(0)

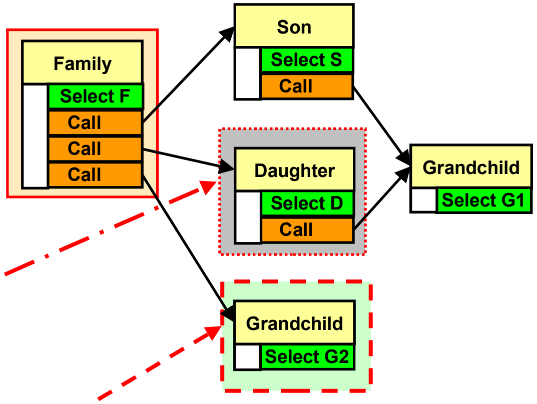
Select of Daughter(1) shows

SELECT D

∑ of Call Grandchild(2) called by Daughter(1)

Select of Grandchild(1) shows

SELECT G

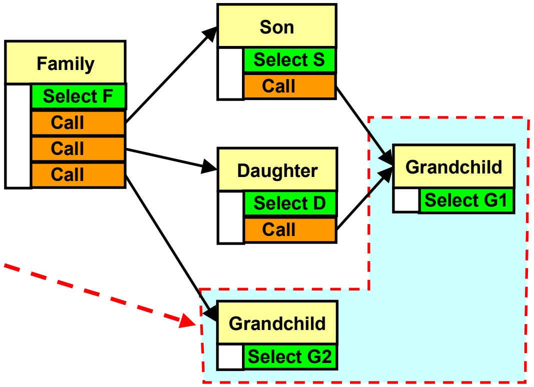


* see next slide

- Action: Show SQL for All Calling Paths

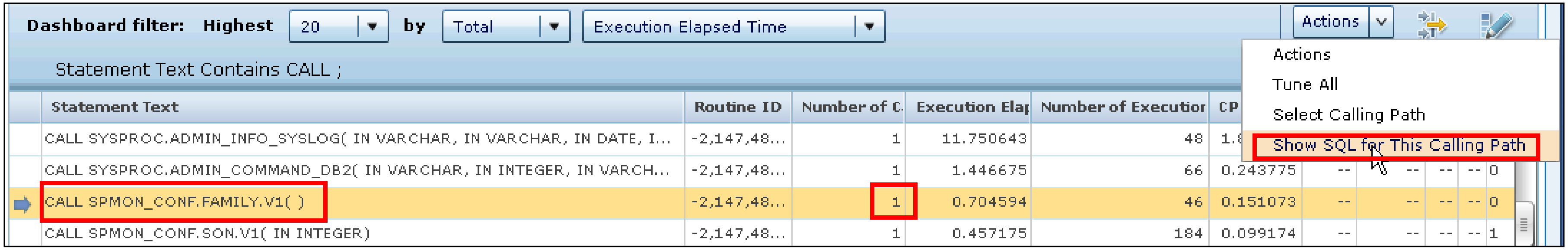
Select Grandchild() shows ∑

SELECT G1
SELECT G2



Show SQL executed by a SP (2/2)

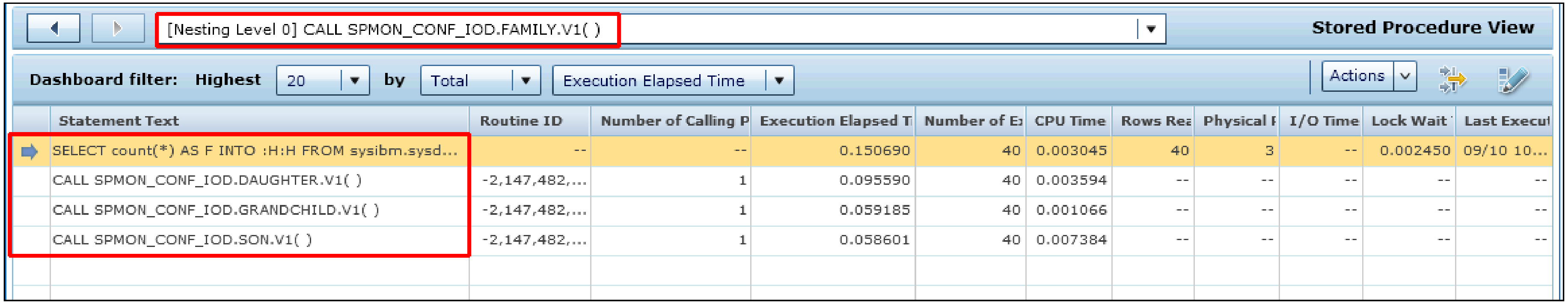
- Show SQL for This Calling Path for Family(0)



Dashboard filter: Highest 20 by Total Execution Elapsed Time

Statement Text Contains CALL ;

	Statement Text	Routine ID	Number of C.	Execution Elap	Number of Executior	CP					
	CALL SYSPROC.ADMIN_INFO_SYSLOG(IN VARCHAR, IN VARCHAR, IN DATE, I...	-2,147,48...	1	11.750643	48	1.8					
	CALL SYSPROC.ADMIN_COMMAND_DB2(IN VARCHAR, IN INTEGER, IN VARCH...	-2,147,48...	1	1.446675	66	0.243775	--	--	--	--	0
➡	CALL SPMON_CONF.FAMILY.V1( )	-2,147,48...	1	0.704594	46	0.151073	--	--	--	--	0
	CALL SPMON_CONF.SON.V1(IN INTEGER)	-2,147,48...	1	0.457175	184	0.099174	--	--	--	--	1



[Nesting Level 0] CALL SPMON_CONF_IOD.FAMILY.V1( )

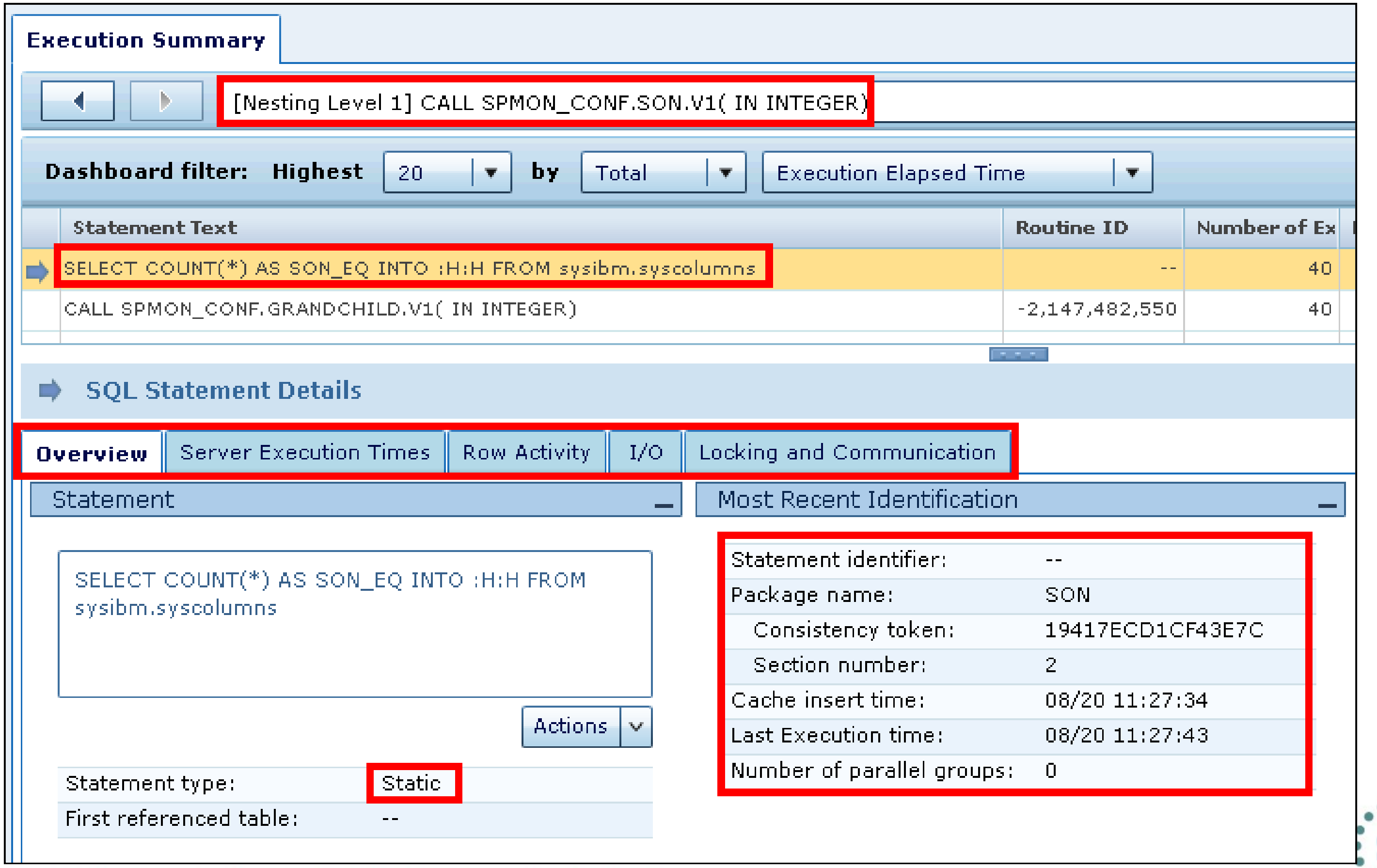
Stored Procedure View

Dashboard filter: Highest 20 by Total Execution Elapsed Time

	Statement Text	Routine ID	Number of Calling P	Execution Elapsed T	Number of Ex	CPU Time	Rows Rea	Physical I	I/O Time	Lock Wait	Last Execut
➡	SELECT count(*) AS F INTO :H:H FROM sysibm.sysd...	--	--	0.150690	40	0.003045	40	3	--	0.002450	09/10 10...
	CALL SPMON_CONF_IOD.DAUGHTER.V1( )	-2,147,482,...	1	0.095590	40	0.003594	--	--	--	--	--
	CALL SPMON_CONF_IOD.GRANDCHILD.V1( )	-2,147,482,...	1	0.059185	40	0.001066	--	--	--	--	--
	CALL SPMON_CONF_IOD.SON.V1( )	-2,147,482,...	1	0.058601	40	0.007384	--	--	--	--	--

SQL cache correlation

- For a nested statement correlation to the cache is shown in “SQL Statements Details” area:


Execution Summary
Dashboard filter: Highest 20 by Total Execution Elapsed Time
SQL Statement Details
Overview
Server Execution Times
Row Activity
I/O
Locking and Communication
Statement
Most Recent Identification
Actions

[Nesting Level 1] CALL SPMON_CONF.SON.V1(IN INTEGER)

Statement Text	Routine ID	Number of Ex
SELECT COUNT(*) AS SON_EQ INTO :H:H FROM sysibm.syscolumns	--	40
CALL SPMON_CONF.GRANDCHILD.V1(IN INTEGER)	-2,147,482,550	40

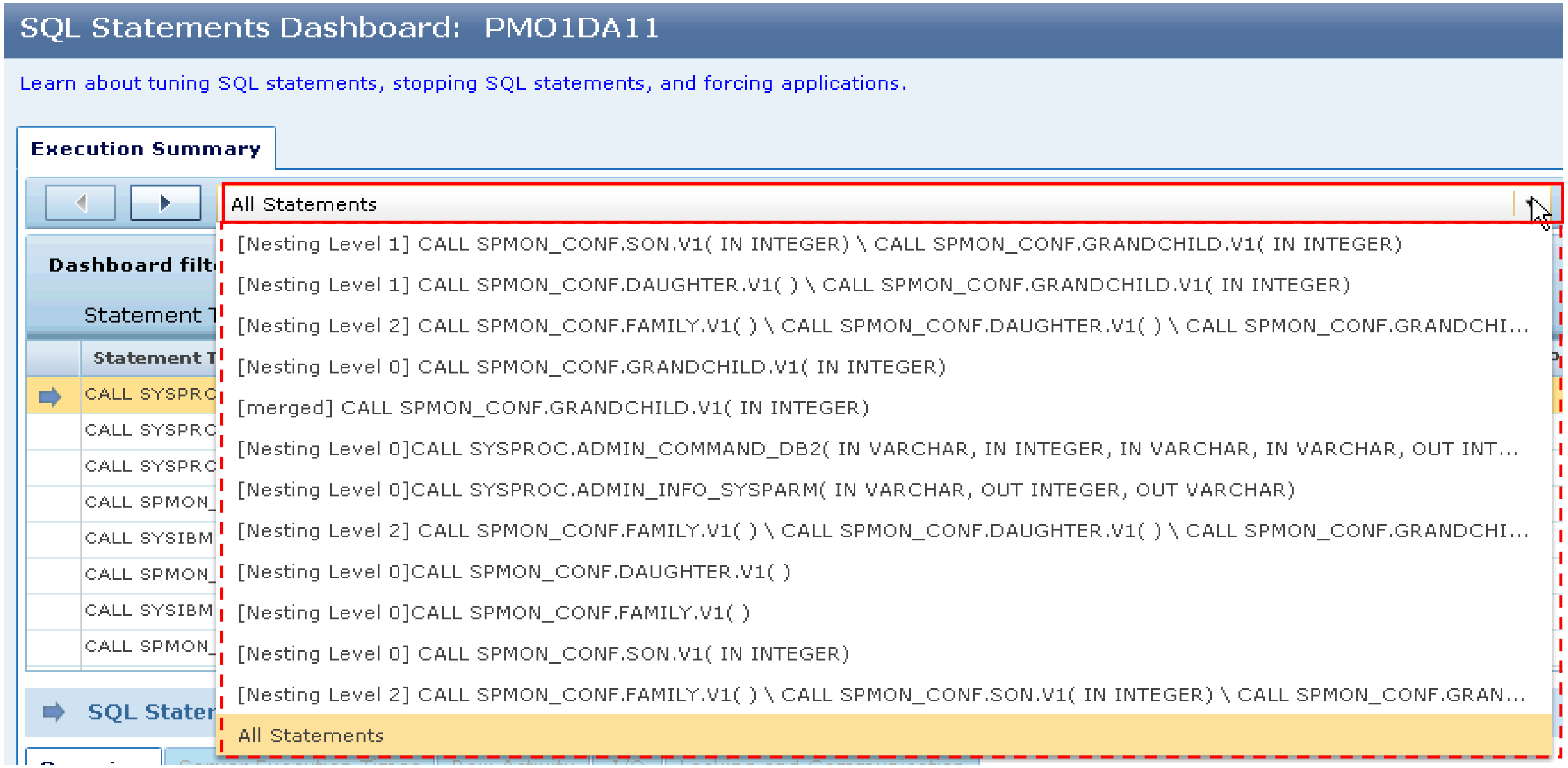
```
SELECT COUNT(*) AS SON_EQ INTO :H:H FROM
sysibm.syscolumns
```

Statement type:	Static
First referenced table:	--

Statement identifier:	--
Package name:	SON
Consistency token:	19417ECD1CF43E7C
Section number:	2
Cache insert time:	08/20 11:27:34
Last Execution time:	08/20 11:27:43
Number of parallel groups:	0

History navigator

- The History Navigator shows the drill down history for stored procedures and can be used similar to a Browser History



SQL Statements Dashboard: PMO1DA11

Learn about tuning SQL statements, stopping SQL statements, and forcing applications.

Execution Summary

All Statements

[Nesting Level 1] CALL SPMON_CONF.SON.V1(IN INTEGER) \ CALL SPMON_CONF.GRANDCHILD.V1(IN INTEGER)
[Nesting Level 1] CALL SPMON_CONF.DAUGHTER.V1( ) \ CALL SPMON_CONF.GRANDCHILD.V1(IN INTEGER)
[Nesting Level 2] CALL SPMON_CONF.FAMILY.V1( ) \ CALL SPMON_CONF.DAUGHTER.V1( ) \ CALL SPMON_CONF.GRANDCHI...
[Nesting Level 0] CALL SPMON_CONF.GRANDCHILD.V1(IN INTEGER)
[merged] CALL SPMON_CONF.GRANDCHILD.V1(IN INTEGER)
[Nesting Level 0]CALL SYSPROC.ADMIN_COMMAND_DB2(IN VARCHAR, IN INTEGER, IN VARCHAR, IN VARCHAR, OUT INT...
[Nesting Level 0]CALL SYSPROC.ADMIN_INFO_SYSPARM(IN VARCHAR, OUT INTEGER, OUT VARCHAR)
[Nesting Level 2] CALL SPMON_CONF.FAMILY.V1( ) \ CALL SPMON_CONF.DAUGHTER.V1( ) \ CALL SPMON_CONF.GRANDCHI...
[Nesting Level 0]CALL SPMON_CONF.DAUGHTER.V1( )
[Nesting Level 0]CALL SPMON_CONF.FAMILY.V1( )
[Nesting Level 0] CALL SPMON_CONF.SON.V1(IN INTEGER)
[Nesting Level 2] CALL SPMON_CONF.FAMILY.V1( ) \ CALL SPMON_CONF.SON.V1(IN INTEGER) \ CALL SPMON_CONF.GRAN...
All Statements

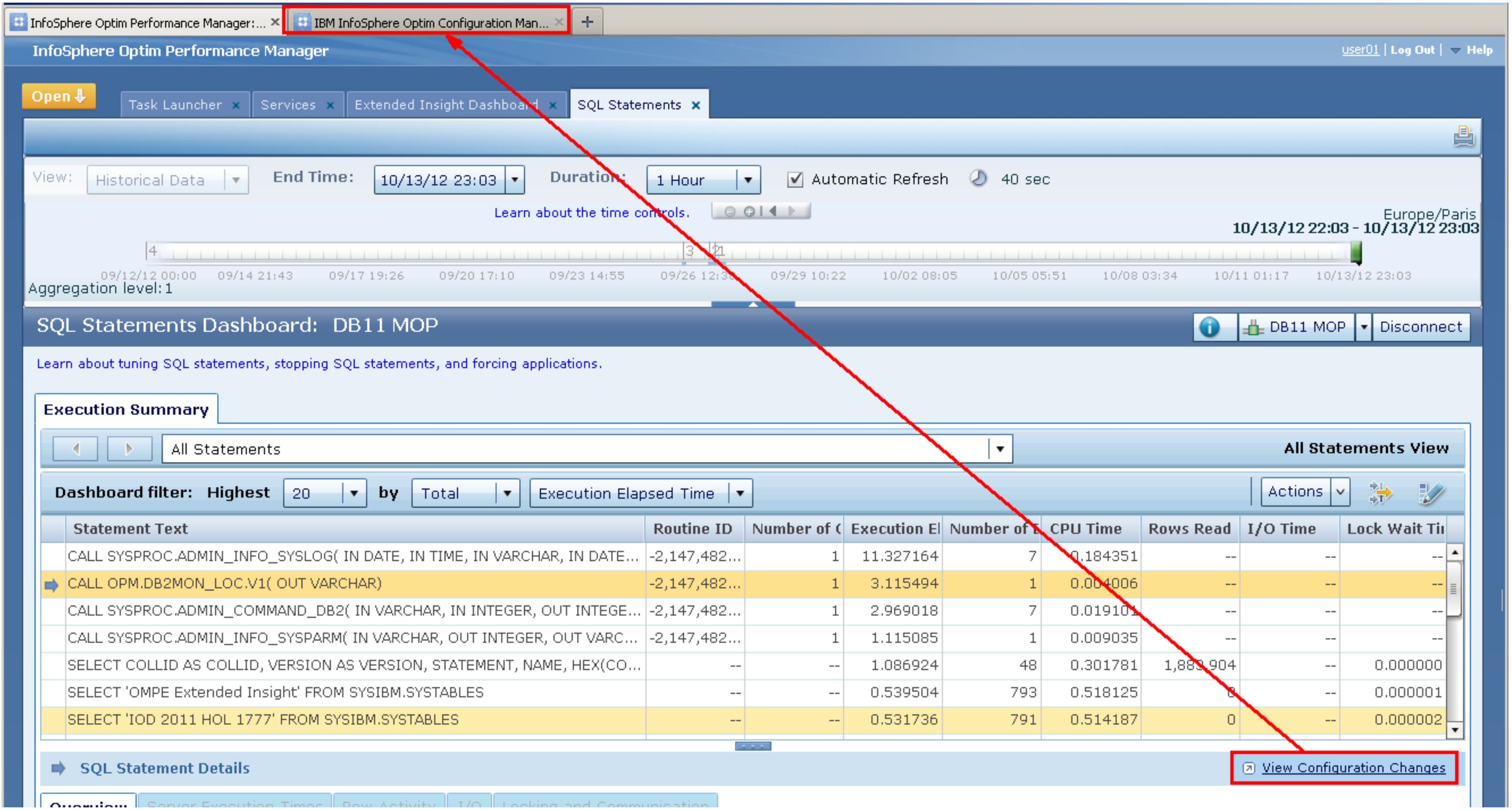
Isolating and tuning stored procedures

- The poorly performing stored procedure has been **identified** and its performance **analyzed** using OMEGAMON XE for DB2 Performance Expert

- The next step is to **isolate** the stored procedure so that other workloads will not be adversely affected

- With the stored procedure isolated, then it can be **tuned**

Launch Optim Configuration Manager for z/OS



InfoSphere Optim Performance Manager:...
IBM InfoSphere Optim Configuration Man...
InfoSphere Optim Performance Manager
user01 | Log Out | Help
Open
Task Launcher
Services
Extended Insight Dashboard
SQL Statements

View: Historical Data

End Time: 10/13/12 23:03

Duration: 1 Hour

☑ Automatic Refresh 40 sec

Learn about the time controls.

Europe/Paris
10/13/12 22:03 - 10/13/12 23:03

Aggregation level: 1

SQL Statements Dashboard: DB11 MOP

Learn about tuning SQL statements, stopping SQL statements, and forcing applications.

All Statements

All Statements View

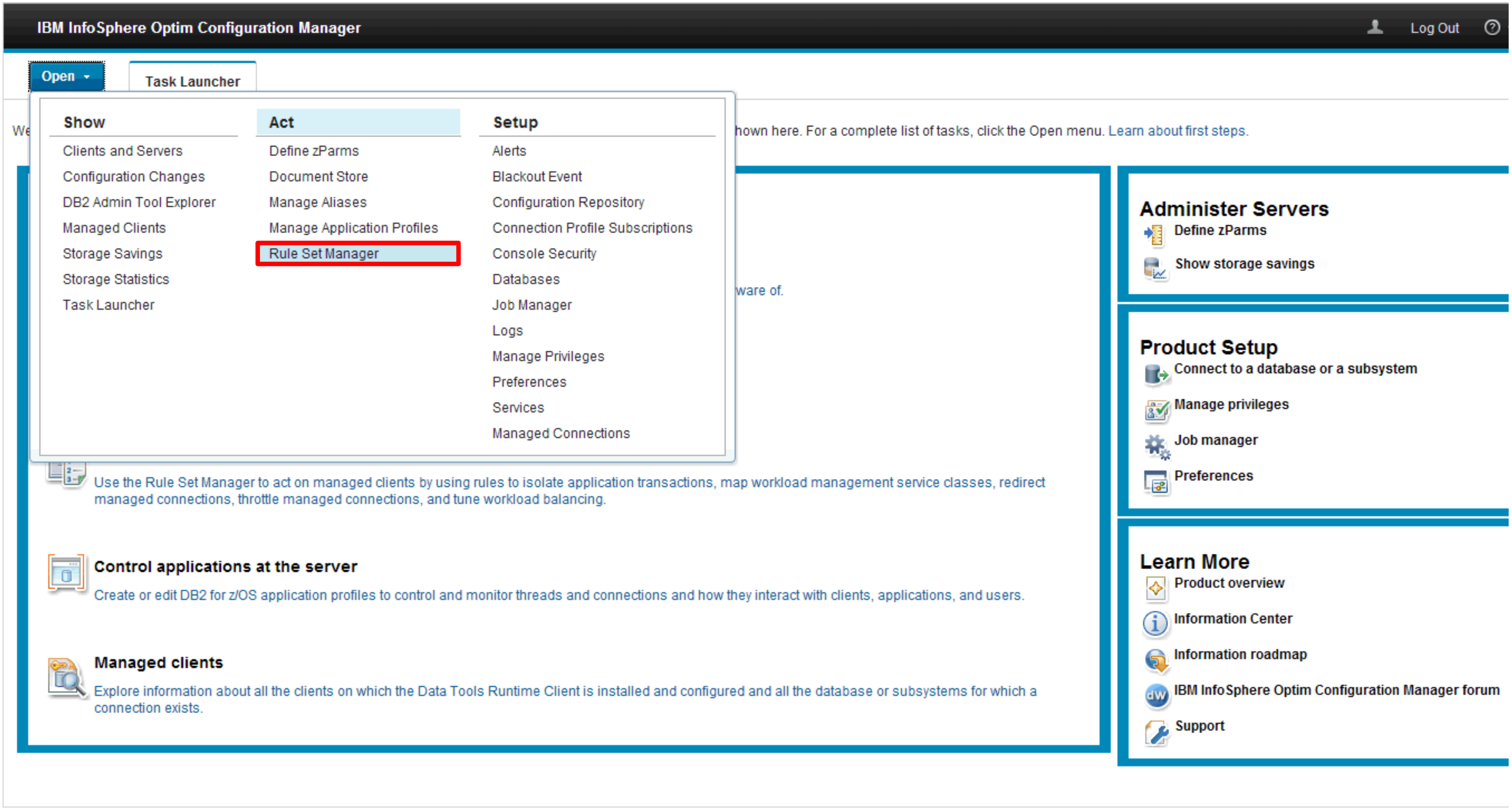
Dashboard filter: Highest 20 by Total Execution Elapsed Time

Statement Text	Routine ID	Number of (	Execution El	Number of E	CPU Time	Rows Read	I/O Time	Lock Wait Ti
CALL SYSPROC.ADMIN_INFO_SYSLOG(IN DATE, IN TIME, IN VARCHAR, IN DATE...	-2,147,482...	1	11.327164	7	0.184351	--	--	--
CALL OPM.DB2MON_LOC.V1(OUT VARCHAR)	-2,147,482...	1	3.115494	1	0.004006	--	--	--
CALL SYSPROC.ADMIN_COMMAND_DB2(IN VARCHAR, IN INTEGER, OUT INTEGE...	-2,147,482...	1	2.969018	7	0.019101	--	--	--
CALL SYSPROC.ADMIN_INFO_SYSPARM(IN VARCHAR, OUT INTEGER, OUT VARC...	-2,147,482...	1	1.115085	1	0.009035	--	--	--
SELECT COLLID AS COLLID, VERSION AS VERSION, STATEMENT, NAME, HEX(CO...	--	--	1.086924	48	0.301781	1,889,904	--	0.000000
SELECT 'OMPE Extended Insight' FROM SYSIBM.SYSTABLES	--	--	0.539504	793	0.518125	0	--	0.000001
SELECT 'IOD 2011 HOL 1777' FROM SYSIBM.SYSTABLES	--	--	0.531736	791	0.514187	0	--	0.000002

SQL Statement Details

View Configuration Changes

Navigate to the “Rule Set Manager” tab



IBM InfoSphere Optim Configuration Manager

Log Out

Open

Task Launcher

We... ...hown here. For a complete list of tasks, click the Open menu. Learn about first steps.

Show	Act	Setup
Clients and Servers	Define zParms	Alerts
Configuration Changes	Document Store	Blackout Event
DB2 Admin Tool Explorer	Manage Aliases	Configuration Repository
Managed Clients	Manage Application Profiles	Connection Profile Subscriptions
Storage Savings	Rule Set Manager	Console Security
Storage Statistics		Databases
Task Launcher		Job Manager
		Logs
		Manage Privileges
		Preferences
		Services
		Managed Connections

...ware of.

Use the Rule Set Manager to act on managed clients by using rules to isolate application transactions, map workload management service classes, redirect managed connections, throttle managed connections, and tune workload balancing.

Control applications at the server

Create or edit DB2 for z/OS application profiles to control and monitor threads and connections and how they interact with clients, applications, and users.

Managed clients

Explore information about all the clients on which the Data Tools Runtime Client is installed and configured and all the database or subsystems for which a connection exists.

Administer Servers

Define zParms

Show storage savings

Product Setup

Connect to a database or a subsystem

Manage privileges

Job manager

Preferences

Learn More

Product overview

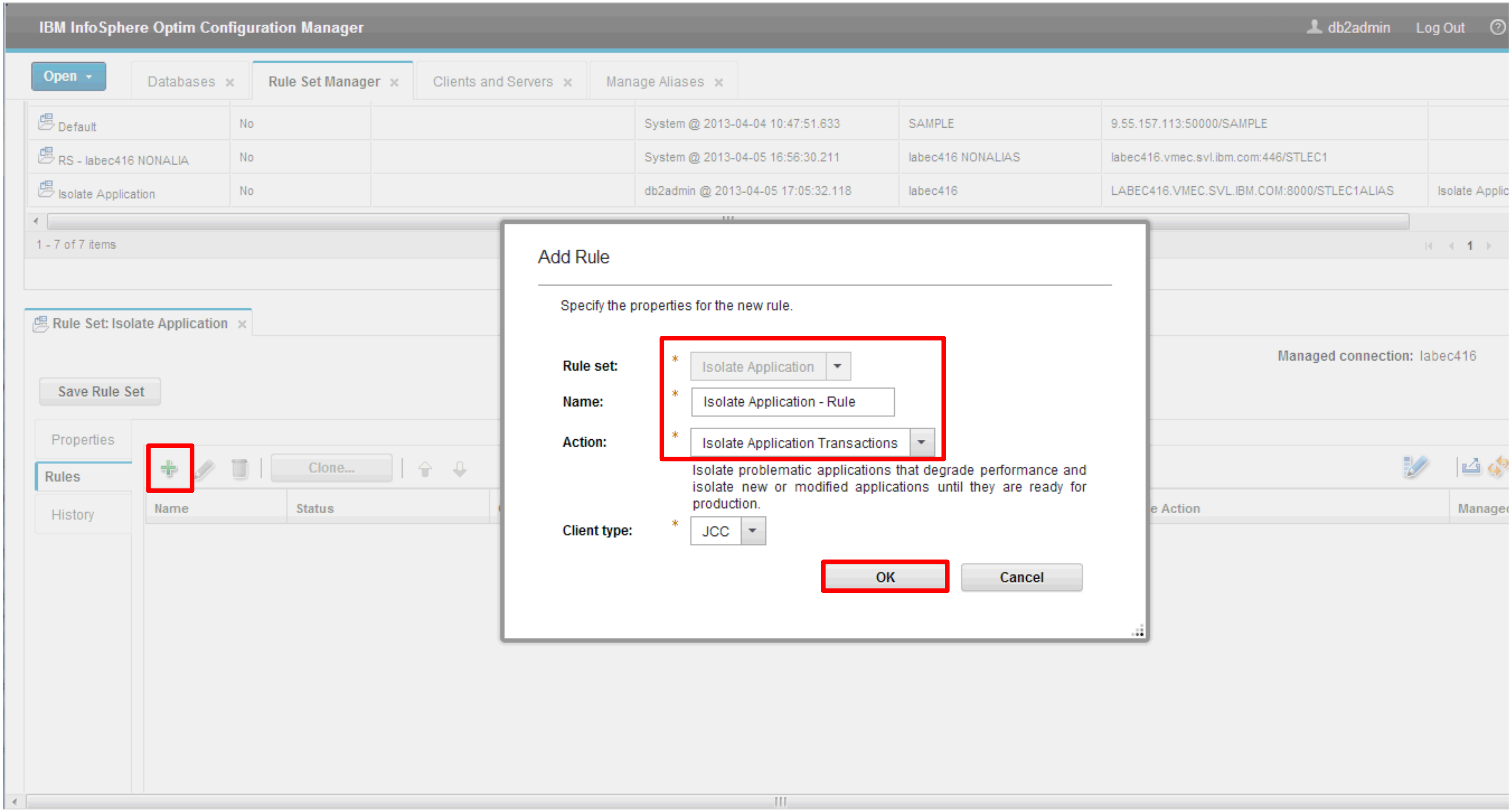
Information Center

Information roadmap

IBM InfoSphere Optim Configuration Manager forum

Support

Add Rule Set To Isolate Application



Default	No		System @ 2013-04-04 10:47:51.633	SAMPLE	9.55.157.113:50000/SAMPLE	
RS - labec416 NONALIA	No		System @ 2013-04-05 16:56:30.211	labec416 NONALIAS	labec416.vmec.svl.ibm.com:446/STLEC1	
Isolate Application	No		db2admin @ 2013-04-05 17:05:32.118	labec416	LABEC416.VMEC.SVL.IBM.COM:8000/STLEC1ALIAS	Isolate Applic

Add Rule

Specify the properties for the new rule.

Rule set: * Isolate Application

Name: * Isolate Application - Rule

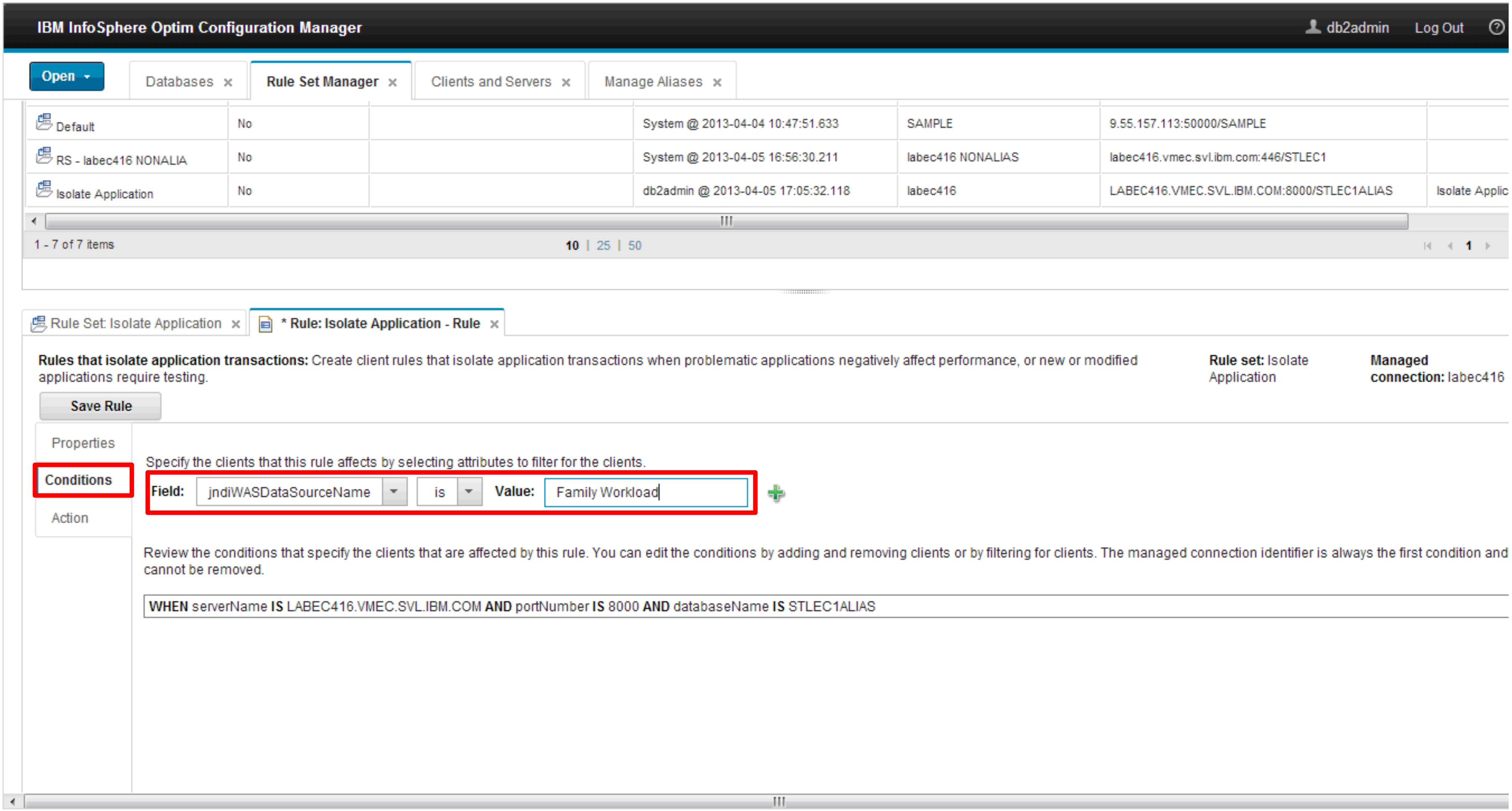
Action: * Isolate Application Transactions

Isolate problematic applications that degrade performance and isolate new or modified applications until they are ready for production.

Client type: * JCC

OK Cancel

Define A Condition



IBM InfoSphere Optim Configuration Manager
db2admin
Log Out
Open
Databases
Rule Set Manager
Clients and Servers
Manage Aliases

Default	No		System @ 2013-04-04 10:47:51.633	SAMPLE	9.55.157.113:50000/SAMPLE	
RS - labec416 NONALIA	No		System @ 2013-04-05 16:56:30.211	labec416 NONALIAS	labec416.vmec.svl.ibm.com:446/STLEC1	
Isolate Application	No		db2admin @ 2013-04-05 17:05:32.118	labec416	LABEC416.VMEC.SVL.IBM.COM:8000/STLEC1ALIAS	Isolate Applic

1 - 7 of 7 items   10 | 25 | 50   1

Rule Set: Isolate Application
* Rule: Isolate Application - Rule

Rules that isolate application transactions: Create client rules that isolate application transactions when problematic applications negatively affect performance, or new or modified applications require testing.

Rule set: Isolate Application

Managed connection: labec416

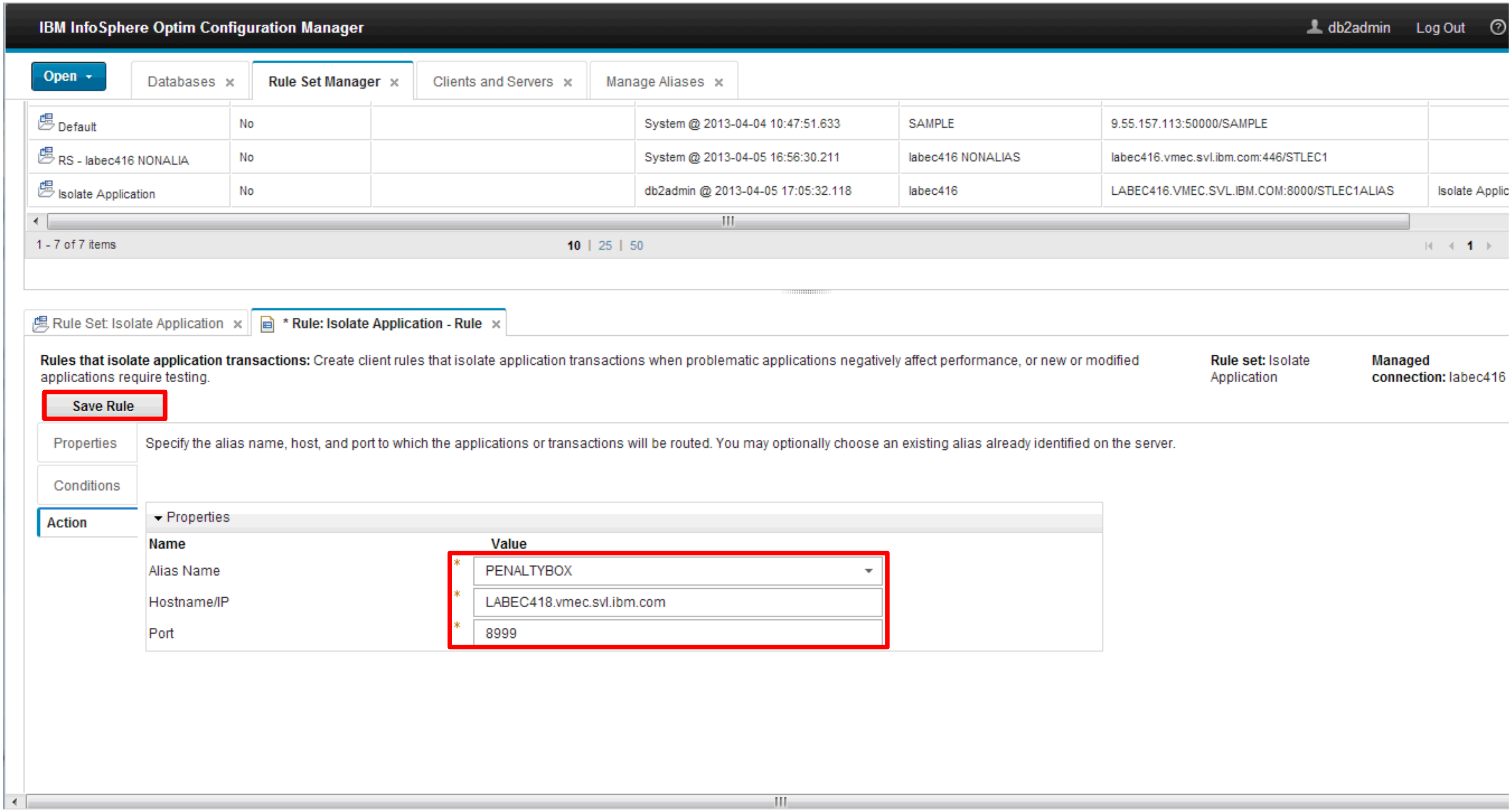
Specify the clients that this rule affects by selecting attributes to filter for the clients.

Field: jndiWASDataSourceName   is   **Value:** Family Workload

Review the conditions that specify the clients that are affected by this rule. You can edit the conditions by adding and removing clients or by filtering for clients. The managed connection identifier is always the first condition and cannot be removed.

WHEN serverName **IS** LABEC416.VMEC.SVL.IBM.COM **AND** portNumber **IS** 8000 **AND** databaseName **IS** STLEC1ALIAS

Define An Action



IBM InfoSphere Optim Configuration Manager
db2admin
Log Out
Open
Databases
Rule Set Manager
Clients and Servers
Manage Aliases

Default	No		System @ 2013-04-04 10:47:51.633	SAMPLE	9.55.157.113:50000/SAMPLE	
RS - labec416 NONALIA	No		System @ 2013-04-05 16:56:30.211	labec416 NONALIAS	labec416.vmec.svl.ibm.com:446/STLEC1	
Isolate Application	No		db2admin @ 2013-04-05 17:05:32.118	labec416	LABEC416.VMEC.SVL.IBM.COM:8000/STLEC1ALIAS	Isolate Applic

1 - 7 of 7 items 10 | 25 | 50 1

Rule Set: Isolate Application
* Rule: Isolate Application - Rule

Rules that isolate application transactions: Create client rules that isolate application transactions when problematic applications negatively affect performance, or new or modified applications require testing.

Rule set: Isolate Application

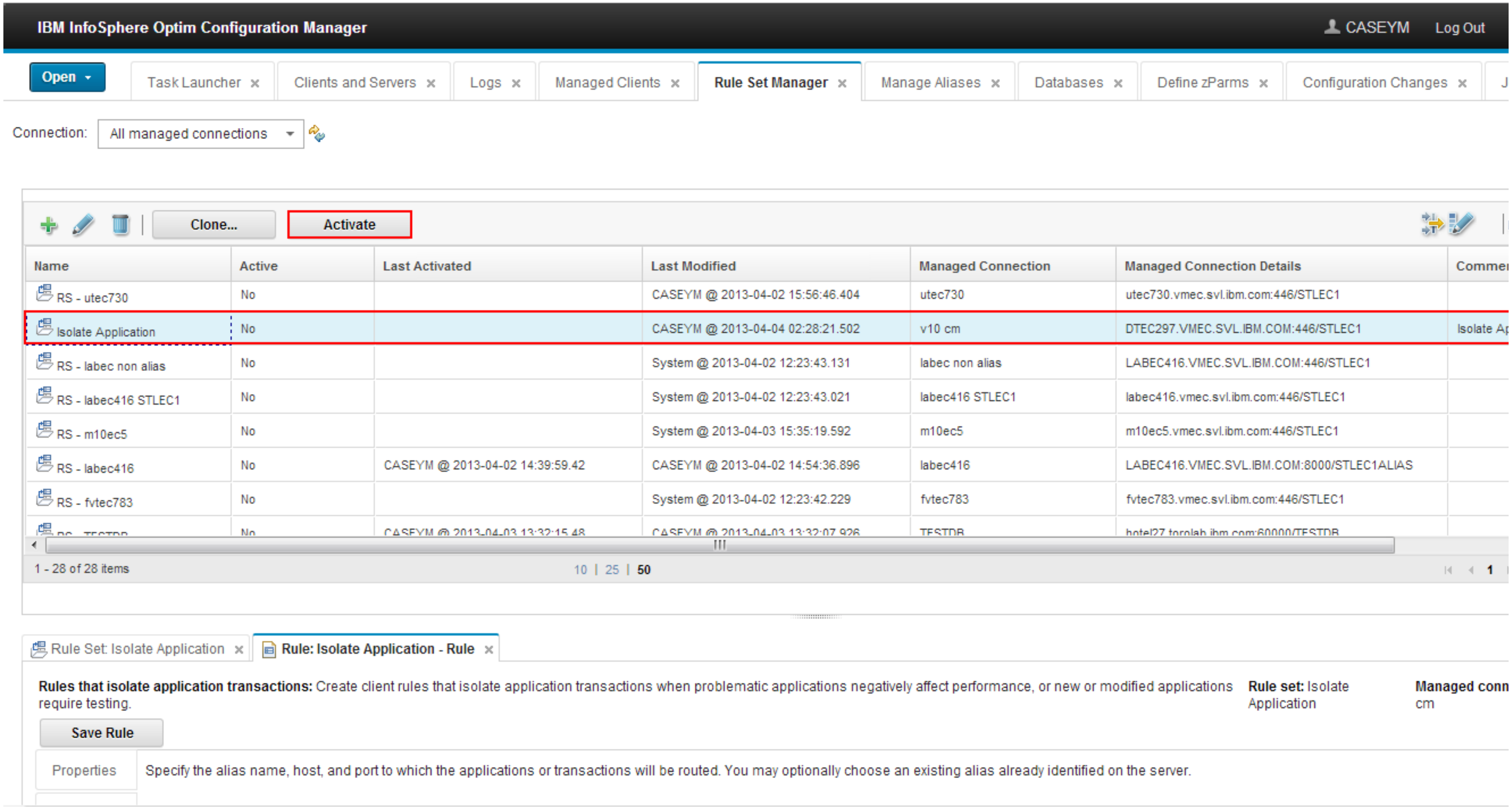
Managed connection: labec416

Specify the alias name, host, and port to which the applications or transactions will be routed. You may optionally choose an existing alias already identified on the server.

Properties

Name	Value
Alias Name	* PENALTYBOX
Hostname/IP	* LABEC418.vmec.svl.ibm.com
Port	* 8999

Activate The Rule Set



Connection: All managed connections

Name	Active	Last Activated	Last Modified	Managed Connection	Managed Connection Details	Commer
RS - utec730	No		CASEYM @ 2013-04-02 15:56:46.404	utec730	utec730.vmec.svl.ibm.com:446/STLEC1	
Isolate Application	No		CASEYM @ 2013-04-04 02:28:21.502	v10 cm	DTEC297.VMEC.SVL.IBM.COM:446/STLEC1	Isolate Ap
RS - labec non alias	No		System @ 2013-04-02 12:23:43.131	labec non alias	LABEC416.VMEC.SVL.IBM.COM:446/STLEC1	
RS - labec416 STLEC1	No		System @ 2013-04-02 12:23:43.021	labec416 STLEC1	labec416.vmec.svl.ibm.com:446/STLEC1	
RS - m10ec5	No		System @ 2013-04-03 15:35:19.592	m10ec5	m10ec5.vmec.svl.ibm.com:446/STLEC1	
RS - labec416	No	CASEYM @ 2013-04-02 14:39:59.42	CASEYM @ 2013-04-02 14:54:36.896	labec416	LABEC416.VMEC.SVL.IBM.COM:8000/STLEC1ALIAS	
RS - fvtec783	No		System @ 2013-04-02 12:23:42.229	fvtec783	fvtec783.vmec.svl.ibm.com:446/STLEC1	
RS - TESTDB	No	CASEYM @ 2013-04-03 13:32:15.48	CASEYM @ 2013-04-03 13:32:07.926	TESTDB	hotel27.torolab.ibm.com:60000/TESTDB	

1 - 28 of 28 items

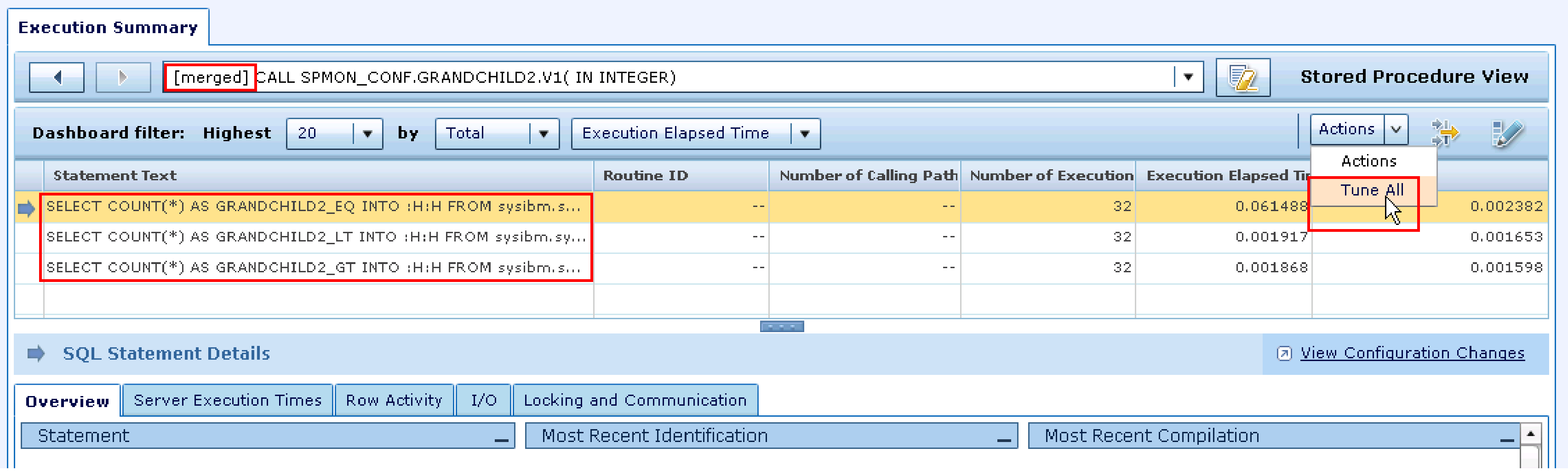
Rules that isolate application transactions: Create client rules that isolate application transactions when problematic applications negatively affect performance, or new or modified applications require testing.

Rule set: Isolate Application

Managed conn cm

Specify the alias name, host, and port to which the applications or transactions will be routed. You may optionally choose an existing alias already identified on the server.

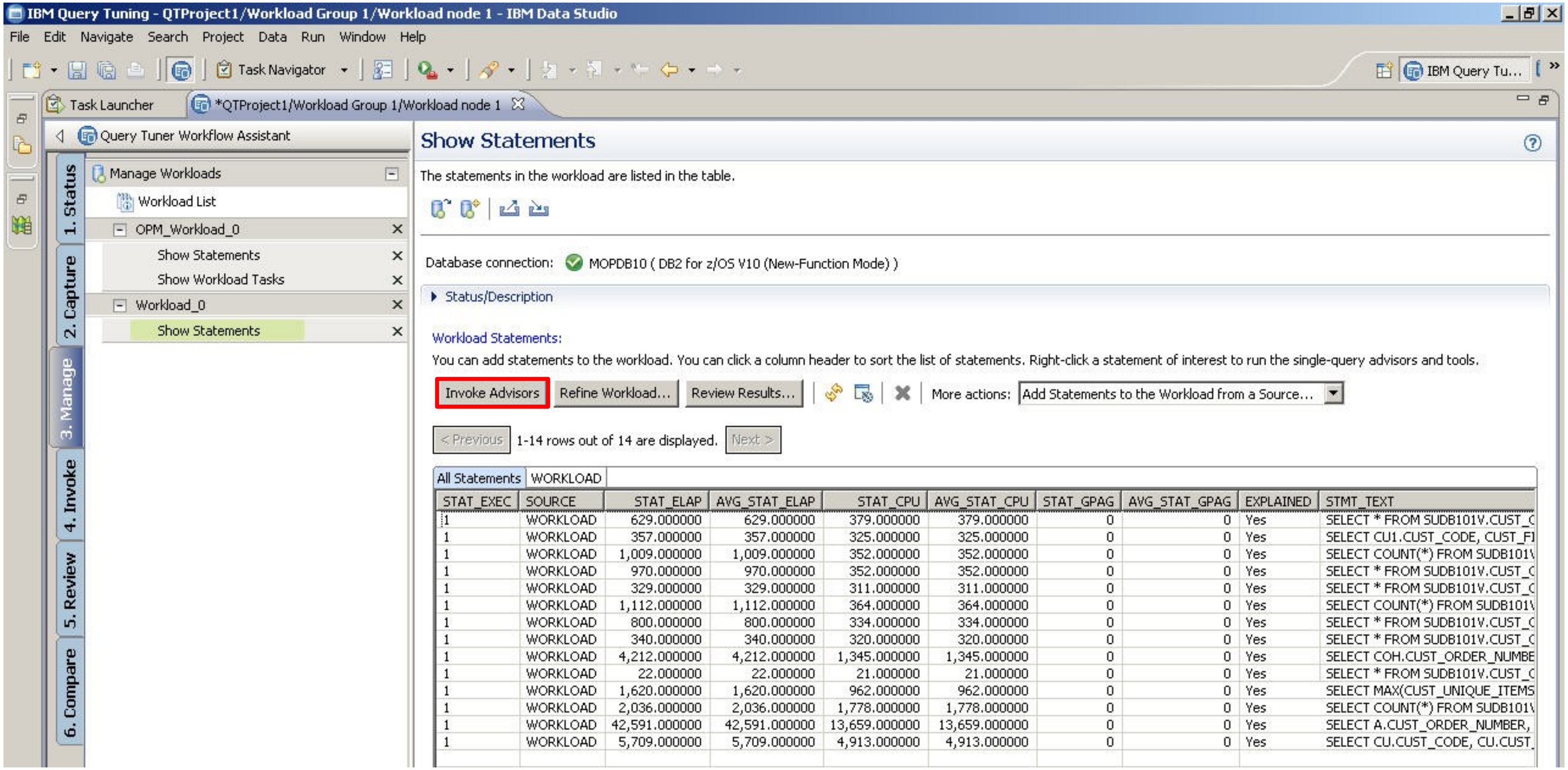
Tuning stored procedures at the SQL workload level



Execution Summary
[merged] CALL SPMON_CONF.GRANDCHILD2.V1(IN INTEGER)
Stored Procedure View
Dashboard filter: Highest 20 by Total Execution Elapsed Time
Actions
Actions
Tune All
SQL Statement Details
View Configuration Changes
Overview
Server Execution Times
Row Activity
I/O
Locking and Communication
Statement
Most Recent Identification
Most Recent Compilation

Statement Text	Routine ID	Number of Calling Path	Number of Execution	Execution Elapsed Ti	
SELECT COUNT(*) AS GRANDCHILD2_EQ INTO :H:H FROM sysibm.s...	--	--	32	0.061488	0.002382
SELECT COUNT(*) AS GRANDCHILD2_LT INTO :H:H FROM sysibm.sy...	--	--	32	0.001917	0.001653
SELECT COUNT(*) AS GRANDCHILD2_GT INTO :H:H FROM sysibm.s...	--	--	32	0.001868	0.001598

SQL Workload is loaded into Optim Query Workload Tuner


IBM Query Tuning - QTProject1/Workload Group 1/Workload node 1 - IBM Data Studio
File Edit Navigate Search Project Data Run Window Help
Task Navigator
IBM Query Tu...
Task Launcher
*QTProject1/Workload Group 1/Workload node 1
Query Tuner Workflow Assistant
1. Status
2. Capture
3. Manage
4. Invoke
5. Review
6. Compare
Manage Workloads
Workload List
OPM_Workload_0
Show Statements
Show Workload Tasks
Workload_0
Show Statements

Show Statements

The statements in the workload are listed in the table.

Database connection: MOPDB10 (DB2 for z/OS V10 (New-Function Mode))

Status/Description

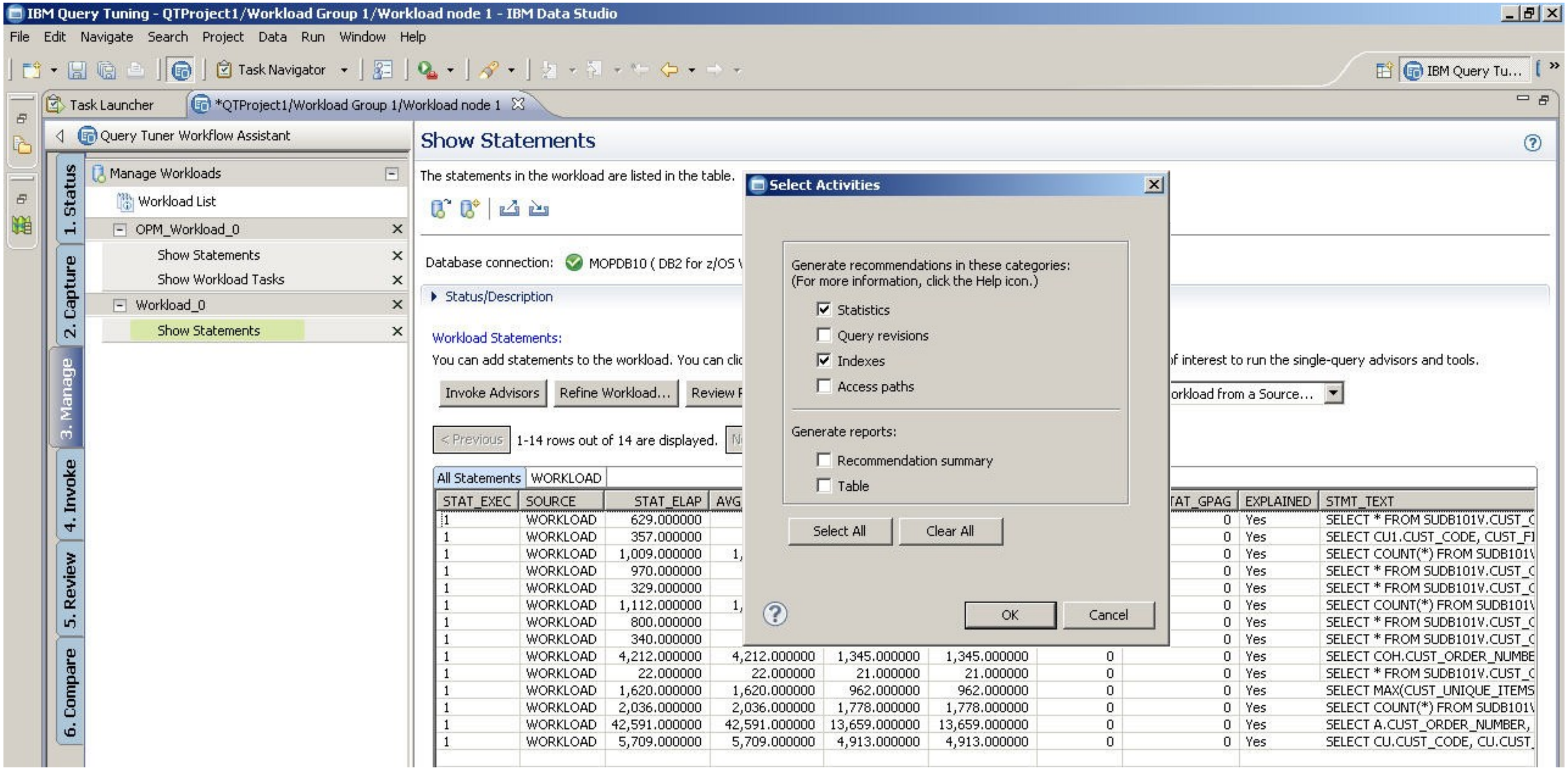
Workload Statements:

You can add statements to the workload. You can click a column header to sort the list of statements. Right-click a statement of interest to run the single-query advisors and tools.

1-14 rows out of 14 are displayed.

STAT_EXEC	SOURCE	STAT_ELAP	AVG_STAT_ELAP	STAT_CPU	AVG_STAT_CPU	STAT_GPAG	AVG_STAT_GPAG	EXPLAINED	STMT_TEXT
1	WORKLOAD	629.000000	629.000000	379.000000	379.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	357.000000	357.000000	325.000000	325.000000	0	0	Yes	SELECT CU1.CUST_CODE, CUST_FI
1	WORKLOAD	1,009.000000	1,009.000000	352.000000	352.000000	0	0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	970.000000	970.000000	352.000000	352.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	329.000000	329.000000	311.000000	311.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	1,112.000000	1,112.000000	364.000000	364.000000	0	0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	800.000000	800.000000	334.000000	334.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	340.000000	340.000000	320.000000	320.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	4,212.000000	4,212.000000	1,345.000000	1,345.000000	0	0	Yes	SELECT COH.CUST_ORDER_NUMBE
1	WORKLOAD	22.000000	22.000000	21.000000	21.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	1,620.000000	1,620.000000	962.000000	962.000000	0	0	Yes	SELECT MAX(CUST_UNIQUE_ITEMS
1	WORKLOAD	2,036.000000	2,036.000000	1,778.000000	1,778.000000	0	0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	42,591.000000	42,591.000000	13,659.000000	13,659.000000	0	0	Yes	SELECT A.CUST_ORDER_NUMBER,
1	WORKLOAD	5,709.000000	5,709.000000	4,913.000000	4,913.000000	0	0	Yes	SELECT CU.CUST_CODE, CU.CUST_

Invoke advisors to generate expert tuning recommendations



IBM Query Tuning - QTProject1/Workload Group 1/Workload node 1 - IBM Data Studio

Show Statements

The statements in the workload are listed in the table.

Database connection: MOPDB10 (DB2 for z/OS \

Status/Description

Workload Statements:

You can add statements to the workload. You can clic … f interest to run the single-query advisors and tools.

1-14 rows out of 14 are displayed.

Select Activities

Generate recommendations in these categories:
(For more information, click the Help icon.)

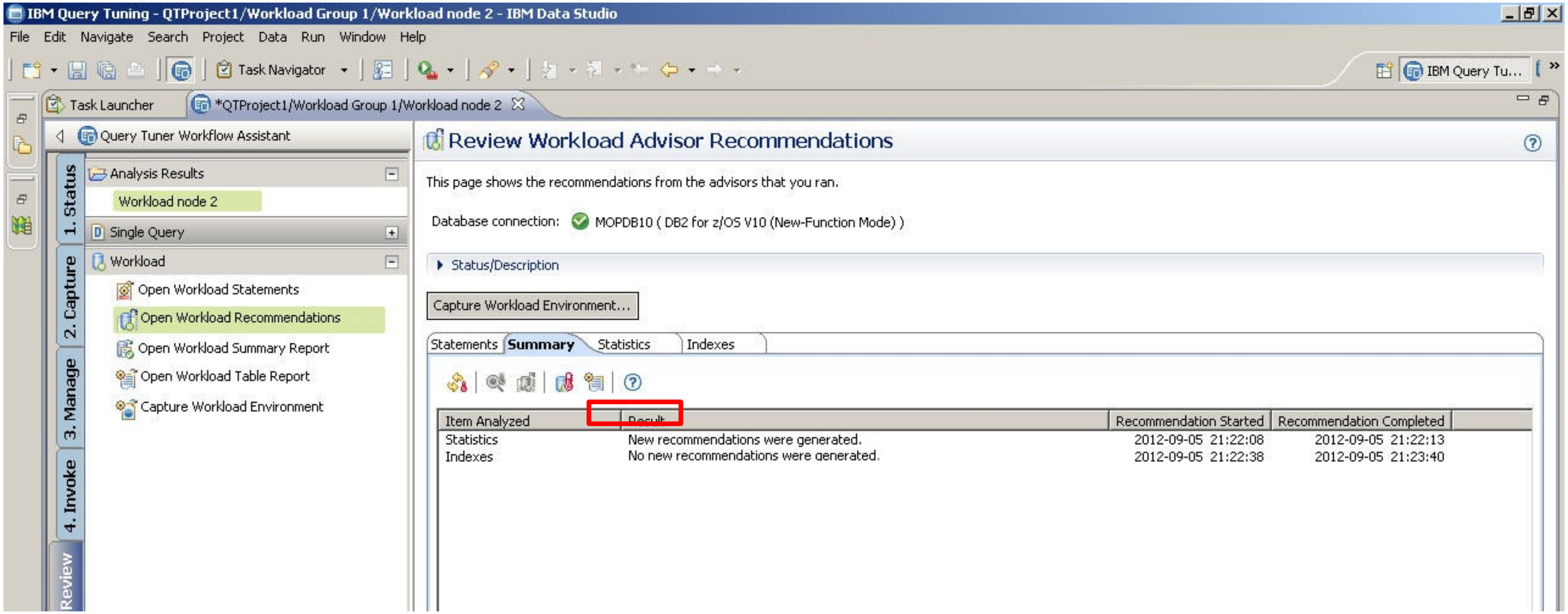
- ☑ Statistics
- □ Query revisions
- ☑ Indexes
- □ Access paths

Generate reports:

- □ Recommendation summary
- □ Table

STAT_EXEC	SOURCE	STAT_ELAP	AVG				TAT_GPAG	EXPLAINED	STMT_TEXT
1	WORKLOAD	629.000000					0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	357.000000					0	Yes	SELECT CU1.CUST_CODE, CUST_FI
1	WORKLOAD	1,009.000000	1,				0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	970.000000					0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	329.000000					0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	1,112.000000	1,				0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	800.000000					0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	340.000000					0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	4,212.000000	4,212.000000	1,345.000000	1,345.000000	0	0	Yes	SELECT COH.CUST_ORDER_NUMBE
1	WORKLOAD	22.000000	22.000000	21.000000	21.000000	0	0	Yes	SELECT * FROM SUDB101V.CUST_C
1	WORKLOAD	1,620.000000	1,620.000000	962.000000	962.000000	0	0	Yes	SELECT MAX(CUST_UNIQUE_ITEMS
1	WORKLOAD	2,036.000000	2,036.000000	1,778.000000	1,778.000000	0	0	Yes	SELECT COUNT(*) FROM SUDB101\
1	WORKLOAD	42,591.000000	42,591.000000	13,659.000000	13,659.000000	0	0	Yes	SELECT A.CUST_ORDER_NUMBER,
1	WORKLOAD	5,709.000000	5,709.000000	4,913.000000	4,913.000000	0	0	Yes	SELECT CU.CUST_CODE, CU.CUST_

Review advisor recommendations summary


IBM Query Tuning - QTProject1/Workload Group 1/Workload node 2 - IBM Data Studio
File Edit Navigate Search Project Data Run Window Help
Task Navigator
IBM Query Tu...
Task Launcher
*QTProject1/Workload Group 1/Workload node 2
Query Tuner Workflow Assistant
1. Status
2. Capture
3. Manage
4. Invoke
Review
Analysis Results
Workload node 2
Single Query
Workload
Open Workload Statements
Open Workload Recommendations
Open Workload Summary Report
Open Workload Table Report
Capture Workload Environment
Status/Description
Capture Workload Environment...
Statements
Summary
Statistics
Indexes

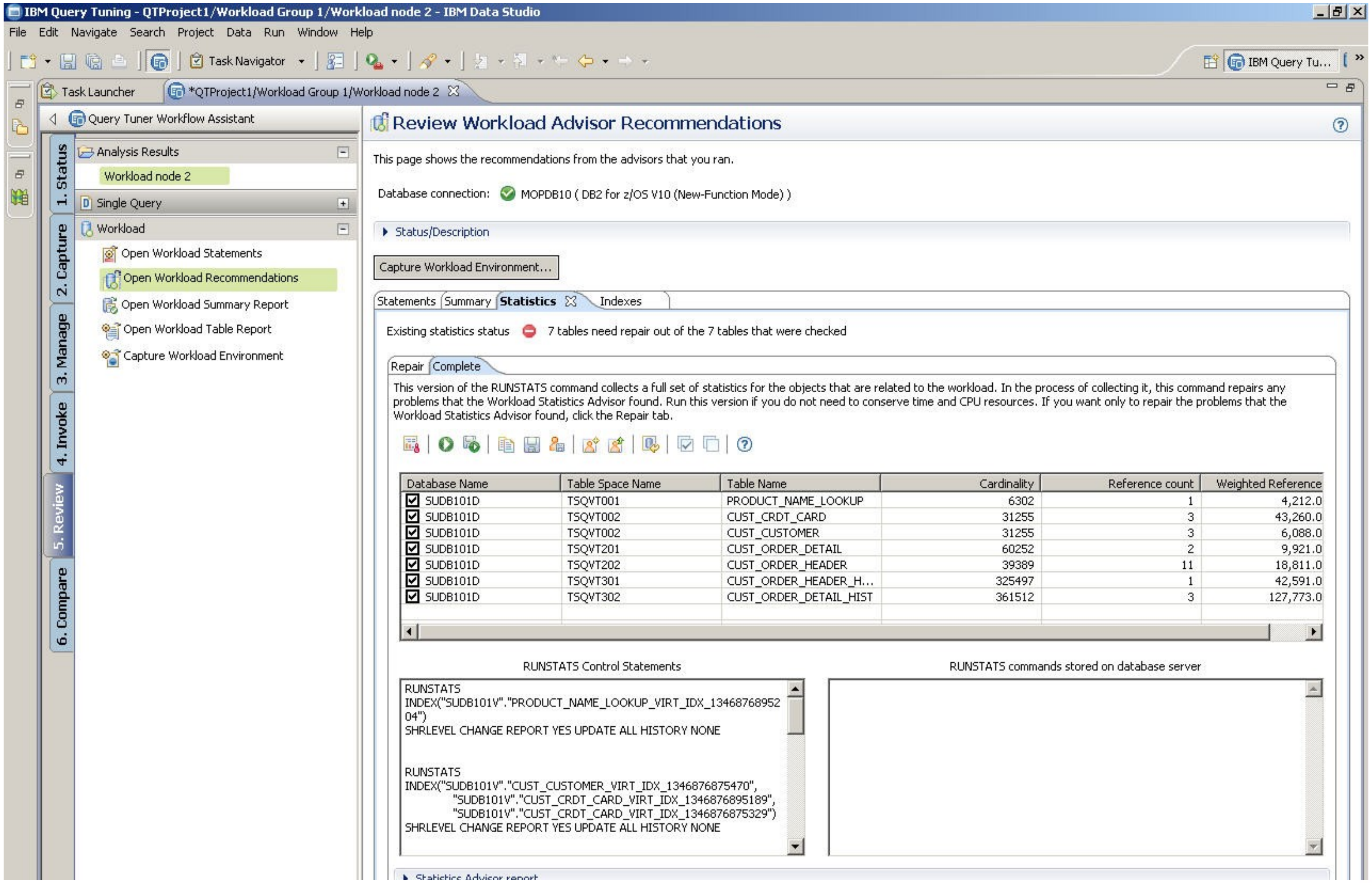
Review Workload Advisor Recommendations

This page shows the recommendations from the advisors that you ran.

Database connection: ✓ MOPDB10 (DB2 for z/OS V10 (New-Function Mode))

Item Analyzed	Result	Recommendation Started	Recommendation Completed
Statistics	New recommendations were generated.	2012-09-05 21:22:08	2012-09-05 21:22:13
Indexes	No new recommendations were generated.	2012-09-05 21:22:38	2012-09-05 21:23:40

Review specific advisor recommendations (Stats)


IBM Query Tuning - QTProject1/Workload Group 1/Workload node 2 - IBM Data Studio
File Edit Navigate Search Project Data Run Window Help
Task Navigator
IBM Query Tu...
Task Launcher
*QTProject1/Workload Group 1/Workload node 2
Query Tuner Workflow Assistant
1. Status
2. Capture
3. Manage
4. Invoke
5. Review
6. Compare
Analysis Results
Workload node 2
Single Query
Workload
Open Workload Statements
Open Workload Recommendations
Open Workload Summary Report
Open Workload Table Report
Capture Workload Environment
Status/Description
Capture Workload Environment...
Statements
Summary
Statistics
Indexes
Repair
Complete
Statistics Advisor report

Review Workload Advisor Recommendations

This page shows the recommendations from the advisors that you ran.

Database connection: MOPDB10 (DB2 for z/OS V10 (New-Function Mode))

Existing statistics status 7 tables need repair out of the 7 tables that were checked

This version of the RUNSTATS command collects a full set of statistics for the objects that are related to the workload. In the process of collecting it, this command repairs any problems that the Workload Statistics Advisor found. Run this version if you do not need to conserve time and CPU resources. If you want only to repair the problems that the Workload Statistics Advisor found, click the Repair tab.

Database Name	Table Space Name	Table Name	Cardinality	Reference count	Weighted Reference
☑ SUDB101D	TSQVT001	PRODUCT_NAME_LOOKUP	6302	1	4,212.0
☑ SUDB101D	TSQVT002	CUST_CRDT_CARD	31255	3	43,260.0
☑ SUDB101D	TSQVT002	CUST_CUSTOMER	31255	3	6,088.0
☑ SUDB101D	TSQVT201	CUST_ORDER_DETAIL	60252	2	9,921.0
☑ SUDB101D	TSQVT202	CUST_ORDER_HEADER	39389	11	18,811.0
☑ SUDB101D	TSQVT301	CUST_ORDER_HEADER_H...	325497	1	42,591.0
☑ SUDB101D	TSQVT302	CUST_ORDER_DETAIL_HIST	361512	3	127,773.0

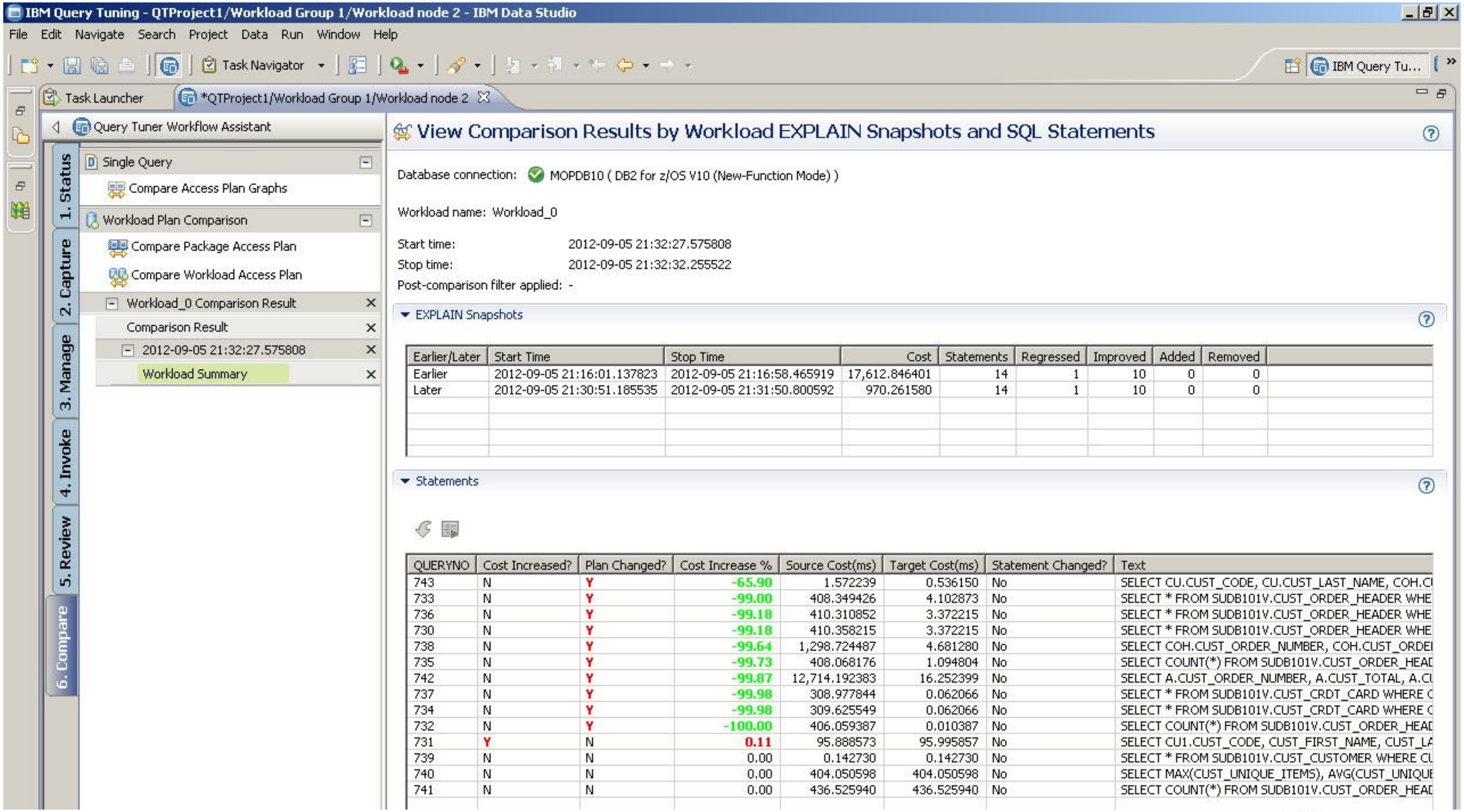
RUNSTATS Control Statements

```
RUNSTATS
INDEX("SUDB101V"."PRODUCT_NAME_LOOKUP_VIRT_IDX_13468768952
04")
SHRLEVEL CHANGE REPORT YES UPDATE ALL HISTORY NONE


RUNSTATS
INDEX("SUDB101V"."CUST_CUSTOMER_VIRT_IDX_1346876875470",
         "SUDB101V"."CUST_CRDT_CARD_VIRT_IDX_1346876895189",
         "SUDB101V"."CUST_CRDT_CARD_VIRT_IDX_1346876875329")
SHRLEVEL CHANGE REPORT YES UPDATE ALL HISTORY NONE
```

RUNSTATS commands stored on database server

Further analysis such as plan comparison



IBM Query Tuning - QTProject1/Workload Group 1/Workload node 2 - IBM Data Studio
File Edit Navigate Search Project Data Run Window Help
Task Navigator
IBM Query Tu...
Task Launcher
*QTProject1/Workload Group 1/Workload node 2
Query Tuner Workflow Assistant
1. Status
2. Capture
3. Manage
4. Invoke
5. Review
6. Compare
Single Query
Compare Access Plan Graphs
Workload Plan Comparison
Compare Package Access Plan
Compare Workload Access Plan
Workload_0 Comparison Result
Comparison Result
2012-09-05 21:32:27.575808
Workload Summary

View Comparison Results by Workload EXPLAIN Snapshots and SQL Statements

Database connection: MOPDB10 (DB2 for z/OS V10 (New-Function Mode))

Workload name: Workload_0

Start time: 2012-09-05 21:32:27.575808

Stop time: 2012-09-05 21:32:32.255522

Post-comparison filter applied: -

EXPLAIN Snapshots

Earlier/Later	Start Time	Stop Time	Cost	Statements	Regressed	Improved	Added	Removed
Earlier	2012-09-05 21:16:01.137823	2012-09-05 21:16:58.465919	17,612.846401	14	1	10	0	0
Later	2012-09-05 21:30:51.185535	2012-09-05 21:31:50.800592	970.261580	14	1	10	0	0

Statements

QUERYNO	Cost Increased?	Plan Changed?	Cost Increase %	Source Cost(ms)	Target Cost(ms)	Statement Changed?	Text
743	N	Y	-65.90	1.572239	0.536150	No	SELECT CU.CUST_CODE, CU.CUST_LAST_NAME, COH.C
733	N	Y	-99.00	408.349426	4.102873	No	SELECT * FROM SUDB101V.CUST_ORDER_HEADER WHE
736	N	Y	-99.18	410.310852	3.372215	No	SELECT * FROM SUDB101V.CUST_ORDER_HEADER WHE
730	N	Y	-99.18	410.358215	3.372215	No	SELECT * FROM SUDB101V.CUST_ORDER_HEADER WHE
738	N	Y	-99.64	1,298.724487	4.681280	No	SELECT COH.CUST_ORDER_NUMBER, COH.CUST_ORDE
735	N	Y	-99.73	408.068176	1.094804	No	SELECT COUNT(*) FROM SUDB101V.CUST_ORDER_HEAD
742	N	Y	-99.87	12,714.192383	16.252399	No	SELECT A.CUST_ORDER_NUMBER, A.CUST_TOTAL, A.C
737	N	Y	-99.98	308.977844	0.062066	No	SELECT * FROM SUDB101V.CUST_CRDT_CARD WHERE C
734	N	Y	-99.98	309.625549	0.062066	No	SELECT * FROM SUDB101V.CUST_CRDT_CARD WHERE C
732	N	Y	-100.00	406.059387	0.010387	No	SELECT COUNT(*) FROM SUDB101V.CUST_ORDER_HEAD
731	Y	N	0.11	95.888573	95.995857	No	SELECT CU1.CUST_CODE, CUST_FIRST_NAME, CUST_L
739	N	N	0.00	0.142730	0.142730	No	SELECT * FROM SUDB101V.CUST_CUSTOMER WHERE C
740	N	N	0.00	404.050598	404.050598	No	SELECT MAX(CUST_UNIQUE_ITEMS), AVG(CUST_UNIQU
741	N	N	0.00	436.525940	436.525940	No	SELECT COUNT(*) FROM SUDB101V.CUST_ORDER_HEAD

Conclusion

- Stored procedures are a good application architecture and programming technique
- Certain types of stored procedures can save MIPs
- Can have many-to-many relationships between applications
- Have the potential to be reused millions of times daily

➢ A key monitoring and tuning opportunity

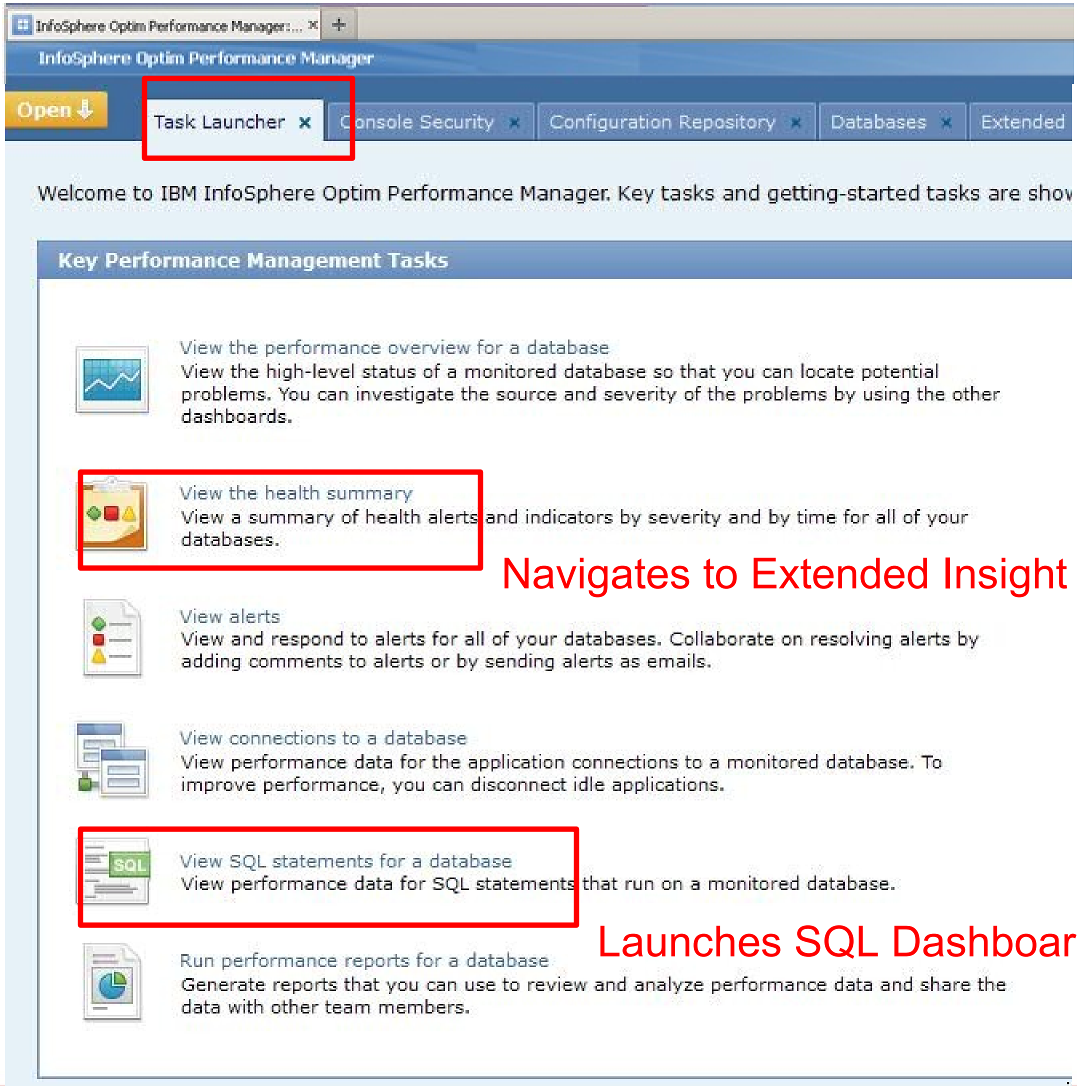
Thank you !

More information

- Websites
  - DB2 for z/OS home page
  - DB2 Tools for z/OS home page
  - Tivoli OMEGAMON XE for DB2 PE on z/OS home page
  - Optim Query Workload Tuner for z/OS home page
  - Optim Configuration Manager for z/OS home page
  - DB2 for z/OS: Information Roadmap
- Other resources
  - Online demo: stored procedure monitoring and analysis
  - eBook: Optimizing database performance through an integrated solution for DB2

Backup slides

Navigating to OMPE Extended Insight and the SQL Dashboard



InfoSphere Optim Performance Manager:...
InfoSphere Optim Performance Manager
Open
Task Launcher
Console Security
Configuration Repository
Databases
Extended

Welcome to IBM InfoSphere Optim Performance Manager. Key tasks and getting-started tasks are show

Key Performance Management Tasks

View the performance overview for a database
View the high-level status of a monitored database so that you can locate potential problems. You can investigate the source and severity of the problems by using the other dashboards.

View the health summary
View a summary of health alerts and indicators by severity and by time for all of your databases.

View alerts
View and respond to alerts for all of your databases. Collaborate on resolving alerts by adding comments to alerts or by sending alerts as emails.

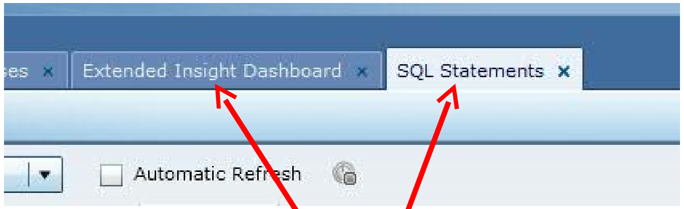
View connections to a database
View performance data for the application connections to a monitored database. To improve performance, you can disconnect idle applications.

View SQL statements for a database
View performance data for SQL statements that run on a monitored database.

Run performance reports for a database
Generate reports that you can use to review and analyze performance data and share the data with other team members.

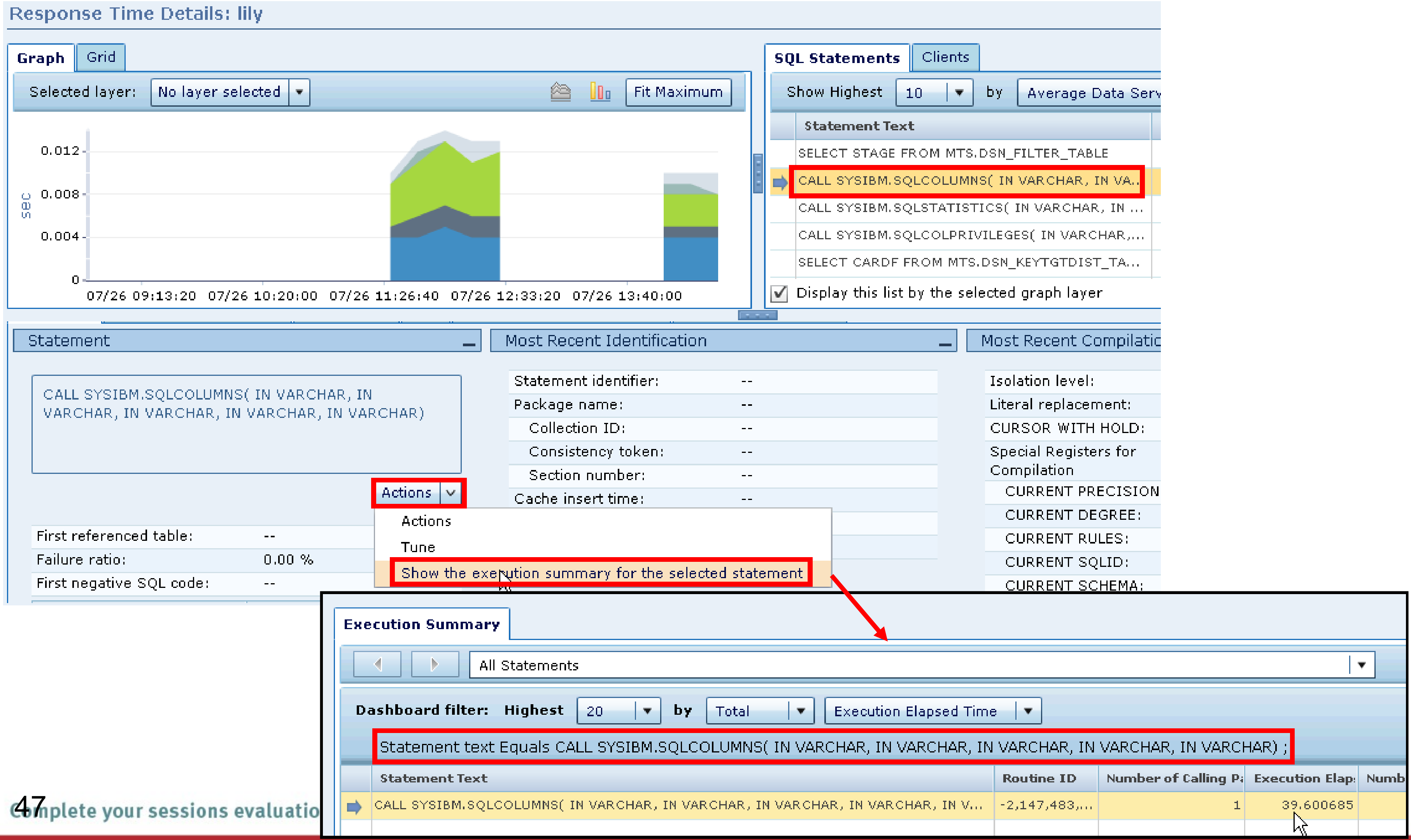
Navigates to Extended Insight

Launches SQL Dashboard



Once launched, tabs are available to navigate between them.

Finally: Link to ‘Extended Insight’ functionality



Response Time Details: lily

Statement Text
SELECT STAGE FROM MTS.DSN_FILTER_TABLE
CALL SYSIBM.SQLCOLUMNS(IN VARCHAR, IN VA...
CALL SYSIBM.SQLSTATISTICS(IN VARCHAR, IN ...
CALL SYSIBM.SQLCOLPRIVILEGES(IN VARCHAR,...
SELECT CARDF FROM MTS.DSN_KEYTGTDIST_TA...

☑ Display this list by the selected graph layer

Statement

CALL SYSIBM.SQLCOLUMNS(IN VARCHAR, IN VARCHAR, IN VARCHAR, IN VARCHAR, IN VARCHAR)

First referenced table:	--
Failure ratio:	0.00 %
First negative SQL code:	--

Most Recent Identification

Statement identifier:	--
Package name:	--
Collection ID:	--
Consistency token:	--
Section number:	--
Cache insert time:	--

Most Recent Compilatic

Isolation level:
Literal replacement:
CURSOR WITH HOLD:
Special Registers for Compilation
CURRENT PRECISION
CURRENT DEGREE:
CURRENT RULES:
CURRENT SQLID:
CURRENT SCHEMA:

Execution Summary

Dashboard filter: Highest 20 by Total Execution Elapsed Time

Statement text Equals CALL SYSIBM.SQLCOLUMNS(IN VARCHAR, IN VARCHAR, IN VARCHAR, IN VARCHAR, IN VARCHAR) ;

Statement Text	Routine ID	Number of Calling Pa	Execution Elap:	Numb
CALL SYSIBM.SQLCOLUMNS(IN VARCHAR, IN VARCHAR, IN VARCHAR, IN VARCHAR, IN V...	-2,147,483,...	1	39.600685	