OMEGAMON Extended Insight
White paper

OMEGAMON Extended Insight Analysis: Where is your application spending its time?

Authors:

Steven E. Fafard
Matthias Tschaffler
Norbert Jenninger

April 2011

Contents

OMEGAMON XE for DB2 Performance Expert V510 and Extended Insight

In October 2010, IBM was the first to market "Extended Insight" into identifying the source of response time problems for web-based and distributed applications running with a DB2 for z/OS backend. Extended Insight is packaged with IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS Version 510, which is IBM's premier DB2 z/OS monitoring offering (the other one is Performance Monitor).

Applications today go through a series of layers and technologies mashed together to deliver business value in a manner suitable to users of all levels of sophistication. The internet enables customers to access personal information and carry out time-sensitive transactions virtually anywhere and anytime. It's critical to every business today to deliver these services to the consumer as reliably and quickly as technologically possible. Business assets, including the most difficult asset to maintain and repair – customer satisfaction – are at risk with every nanosecond of a slow-down or hiccough. Customers must have the highest degree of confidence that their transaction is in safe and secure hands, especially transactions of a financial nature.

System and application programmers, DBAs, and network and system administrators need every tool available to maintain insight into the performance of their applications.

In the old days, a transaction was limited to a single, self-contained system and one monitor was enough. Today, in the interest of extensibility, there are dozens of layers and literally thousands of "hops" to get from point A to point B and there are nearly as many people with responsibility to maintain those touchpoints. How do you communicate between these people? How do you keep your arms around the entire infrastructure supporting the livelihood of your business?

What is the OMEGAMON Extended Insight Analysis Tool?

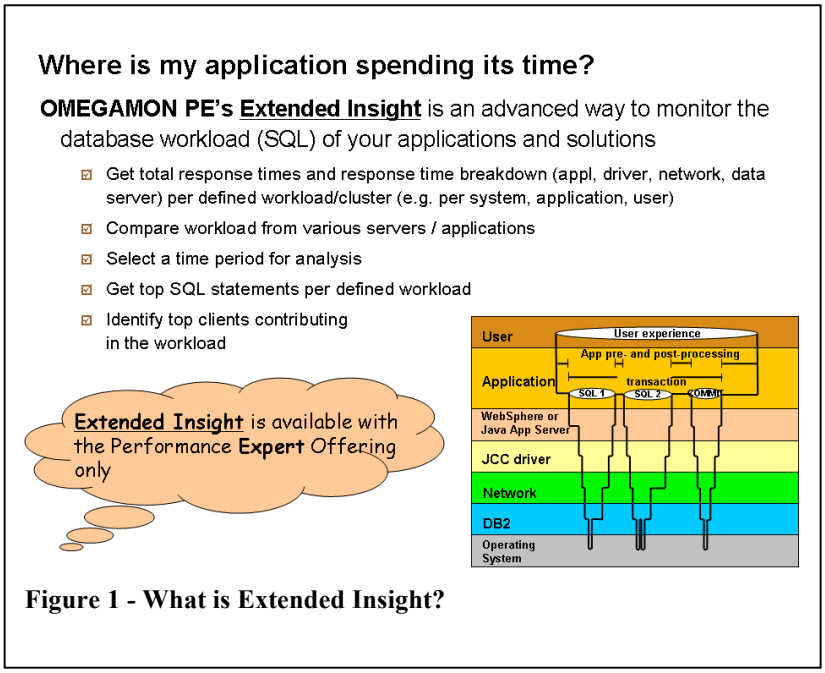
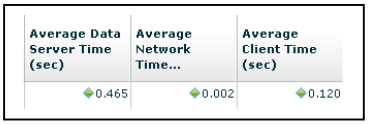


Figure 1 - What is Extended Insight?

In the graphic at the right, we see the typical "layers" or components that make up today's web-based Java™, CLI (SAP), and .NET™ applications. Your end-user's response time experience is measured on the x-axis's "time continuum." Each layer contributes some x-axis time to service the user's request by drilling through the y-axis layers. Traditional DB2 for z/OS monitoring captures metrics mostly about the blue DB2 layer, but that tells us only a minimal amount about what the user experiences. The OMEGAMON Extended Insight feature begins measuring response time at the first SQL invocation and culminates with the final SQL "commit" or "rollback" in the transaction or Unit of Work (UOW.) This technique, coupled with metrics from the JCC driver, enables IBM to calculate the time components that make up DB2 time, network time, driver time, application server time, and application time (between the first and last SQL). **Note**: Application time before and after the first SQL can be captured with additional offerings from IBM.

What does this mean to analysts responsible for service-level agreements and accountable for the optimum user experience (which further translates into customer satisfaction)? It means countless hours are saved that were typically spent trying to first isolate the problem area to one of the layers identified above. Extended Insight makes it readily apparent which layer(s) is primarily responsible.



A simple comparison of average response time captured for each of the major layers quickly directs the investigation to the most likely solution to the problem.

The key dashboard for Extended Insight Analysis is shown below.

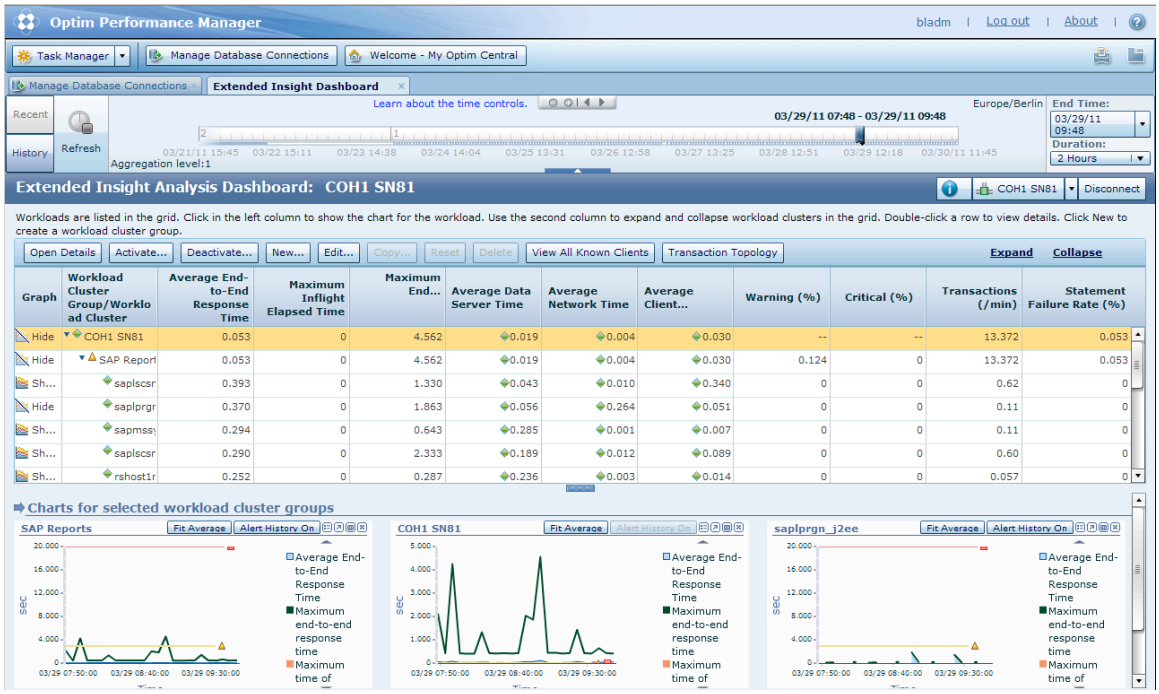


Figure 2 - Extended Insight Analysis Dashboard

Shown here are averages for a specified DB2 subsystem (COH1 SN81) based on the time period selected for analysis for the key components of the application stack, as shown in the first graphic (see "Data Server" (DB2 z/OS), "Network", and "Client"). Two additional metrics highlight the "Average End-To-End Response Time" (the UOW described earlier) and the "Maximum Inflight Elapsed Time" which is the current maximum response time of a transaction that has not finished.

With the architecture depicted in Figure 4, the technique for determining component response requires the capture of a sort of correlation key by the client software at SQL invocation that is then fed to the repository server. OMEGAMON captures the same SQL statement via the DB2 for z/OS statement cache that is correlated with the statement from the client that is in the repository server.

On a one-minute interval, we capture the actual response times and the breakdown (driver time, server time, network time) for each SQL statement execution from the dataserver driver to the client software and report this to the repository database. In the same one-minute interval, the repository server then retrieves the performance metrics for each SQL statement and statement text from the DB2 for z/OS server, and correlates this with the data captured from the distributed client

It's important to understand that the values shown for each layer are average values and represent the time period selected for analysis. The average value may represent a wide range of response times, so where average network time may appear acceptable overall, there may be a problem for a specific application server connection. You can drill-down to a more granular level of detail to identify a SQL issues for a single application.

The Extended Insight histogram in Figure 3 demonstrates why you might want to drill further to understand the detail making up the average values. The Response Time Histogram illustrates the number of transactions that make up each response time bucket composing the average. Using this graphic, it's clear there is still a problem even if the average seems acceptable.

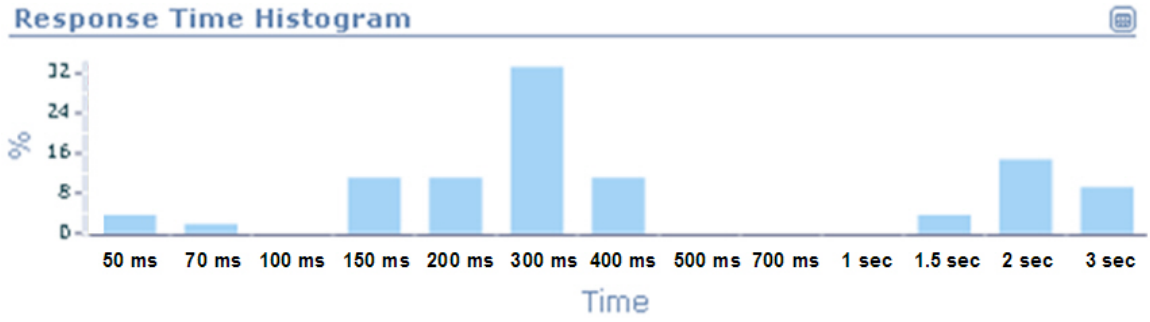


Figure 3 - Response Time Histogram

You might also identify problems not evident at the rolled-up level by expanding the workload cluster groups, for example:

Graph	Workload Cluster Group/Workload Cluster	Average End-to-End Response Time	Maximum Inflight Elapsed Time	Maximum End-to-End Response Time	Average Data Server Time	Average Network Time	Average Client Time
Sh...	SAP application serv	0.071	0	7.460	0.019	0.007	0.046
Sh...	9.152.20.157	0.092	0	7.460	0.020	0.003	0.068
Sh...	9.152.20.159	0.069	0	5.095	0.022	0.012	0.036
Sh...	9.152.20.158	0.040	0	1.829	0.010	0.001	0.028

Using this, the analyst is able to find the server suffering from the poorest network connection or poor client response time, which may in turn be an application server or operating system workload issue.

How are the response time components calculated?

Figure 4 depicts the product components that enable *extended insight* into application response time.

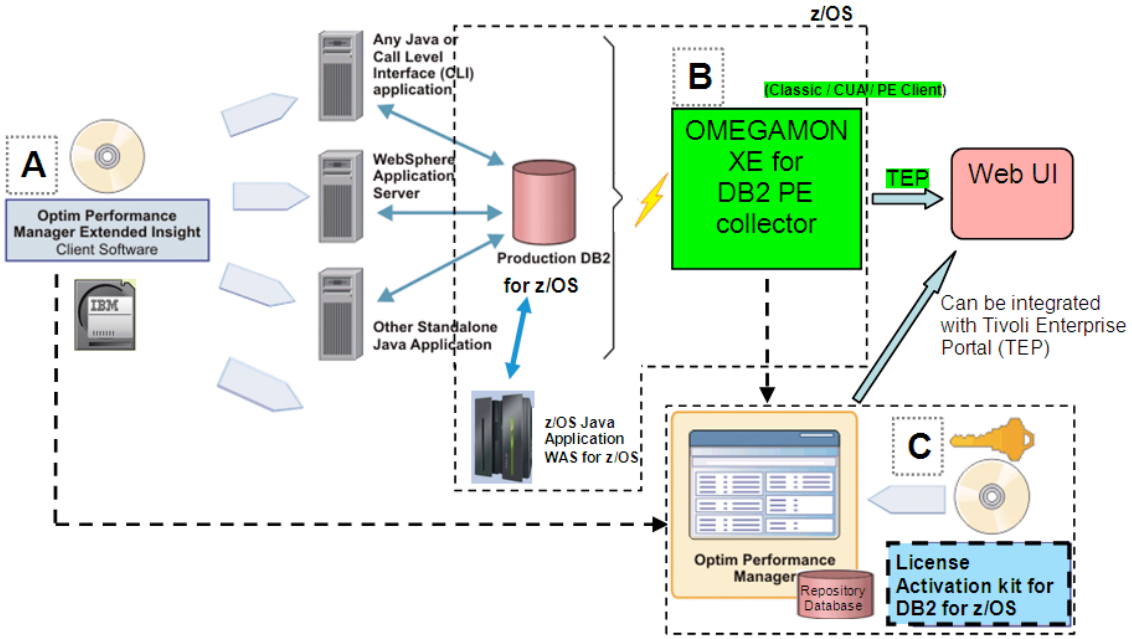


Figure 4 - General architecture/components for Extended Insight

The key components are:

A. A Client Software component (on the left side) installed on the appropriate application server that is responsible for collecting the application and driver metrics at the location that is issuing the SQL to the DB2 for z/OS database
B. The OMEGAMON XE for DB2 PE collector that is responsible for capturing SQL statements and statement metrics from the SQL cache to be correlated with the client SQL request metrics.
C. The Repository Database controlled by the embedded repository server named Optim Performance Manager, that is responsible for correlating and merging the metrics captured from A and B. (Note, in this figure, the WebSphere® Application Server or console server serving the "Web UI" that accesses the repository database for display is not shown.)

Together, these components correlate timestamps, SQL statements, application time, and DB2 statistics in order to stitch together a response time story that quickly pinpoints where the bottleneck is.

Identifying and solving response time problems

Having response time broken into components is more meaningful if the workload being measured is a workload that makes sense to the business. While measuring at a physical level might provide some level of insight, translating the problem into business impact carries a much greater sense of importance. As shown in Figure 5, the OMEGAMON Extended Insight ships a set of initial, customizable workload clusters of Client Application Name, Application Type, Client User ID, and Host names/IP address.

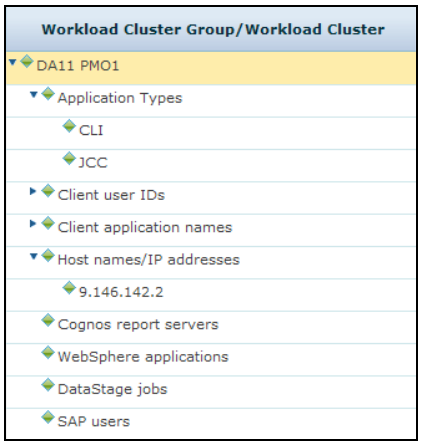


Figure 5 - Default workload clusters can be customized for more meaning

Expanding any one of these clusters will reveal the default application, user IDs, or custom applications, depending on how the connection was made and the application is coded. To turn these default names into meaningful business names, application developers and DBAs should leverage what DB2 for z/OS provides in the form of special registers that remote applications can use to identify applications, users and user location to DB2 and Extended Insight. Here is an example: CURRENT CLIENT_APPLNAME: application name, CURRENT CLIENT_USERID: client user ID, CURRENT CLIENT_WRKSTNNAME: workstation name.

IMPORTANT: Extended Insight relies on the content in these special register to create meaningful clusters for problem analysis. The following depicts an example of how an application might set these values to enable effective Extended Insight clustering:

```
DB2ConnectionPoolDataSource ds = new DB2ConnectionPoolDataSource();
            ds.setUser("myuser");
            ds.setPassword("mypass");
            ds.setDatabaseName("mydb");
            ds.setConnectionAttribute
              ("ClientApplName=WebSphere-Samples;
                 ClientWrkstnName=WebSphere-Wkstn;
                 ClientUserid=WebSphere-User
                 ClientAcctStr=WebSphere-Acctstr");
            PooledConnection poolconn = ds.getPooledConnection();
            con = poolconn.getConnection();
```

In addition to specifying the values at connection (as shown here), they can also be changed for an existing connection by programmatic assignment during application execution. For more details, we highly recommend this whitepaper "**Monitoring WebSphere Applications on DB2 Servers**."

You would then define these custom workload clusters to Extended Insight using the New or Edit buttons (see Figure 2 - Extended Insight Analysis Dashboard) using the Client fields above in combination with qualification values to narrow down the workload cluster to the level of granularity that makes sense to the analysts.

Alternatively you can choose one of the built-in workloads IBM provides using the "Activate" Activate... button. IBM worked with SAP and Cognos®, among others, to predefine workload clusters for quicker time-to-value with Extended Insight.

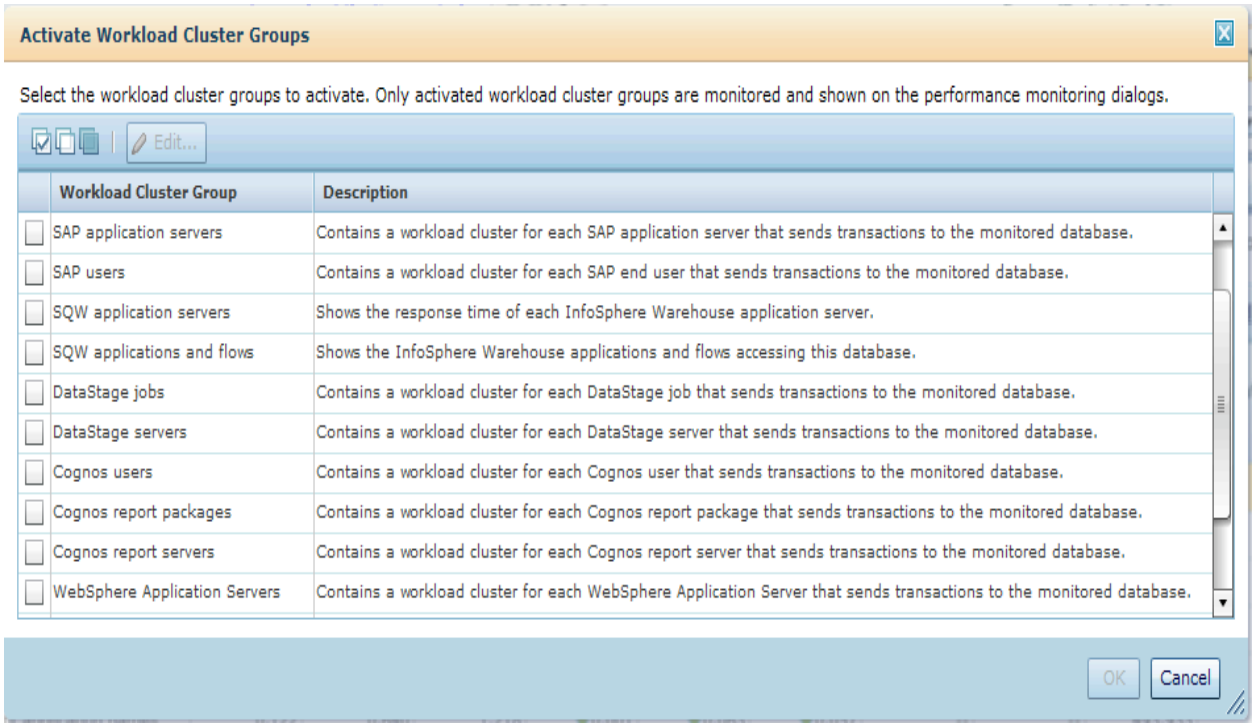


Figure 6 - Predefined workload clusters

Investigating the application itself

Aside from business logic, good SQL or bad SQL can make or break DB2 application performance. The OMEGAMON capability to correlate the DB2 for z/OS SQL statement and related metrics from the DB2 for z/OS statement cache with the application issuing the SQL is where the real diagnostic power of Extended Insight comes to bear. By providing the top 'n' SQL statements executing in the data server (i.e., DB2) for a given workload, DBAs can pinpoint potentially troublesome SQL.

Highlight a workload and click Open Details 

to get to the top SQL

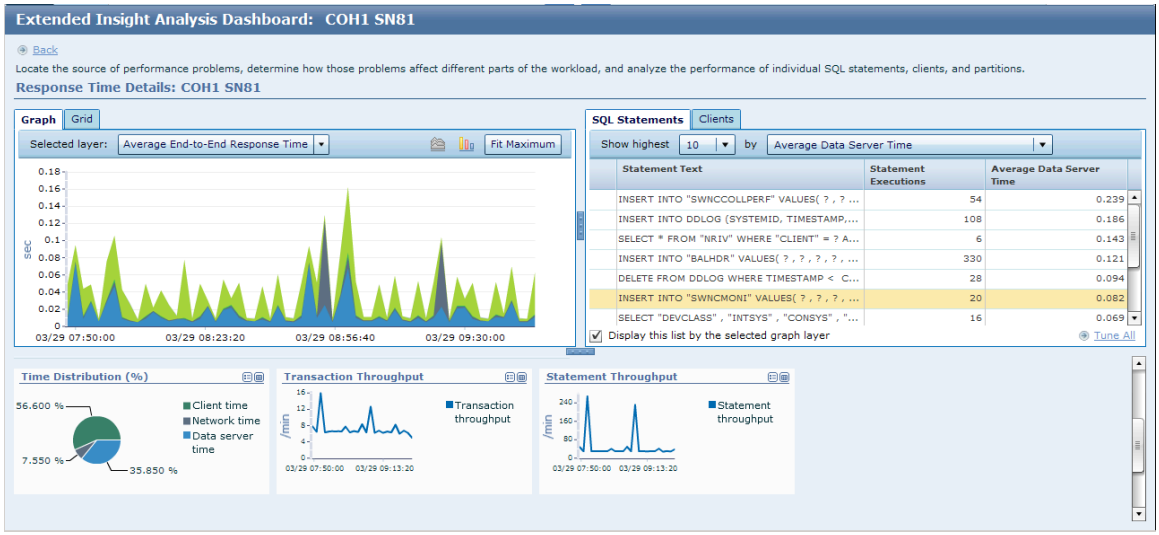


Figure 7 - Response Time Details Overview

Highlight an SQL statement in the box at the upper right from Figure 7 - to reveal the following SQL detail.

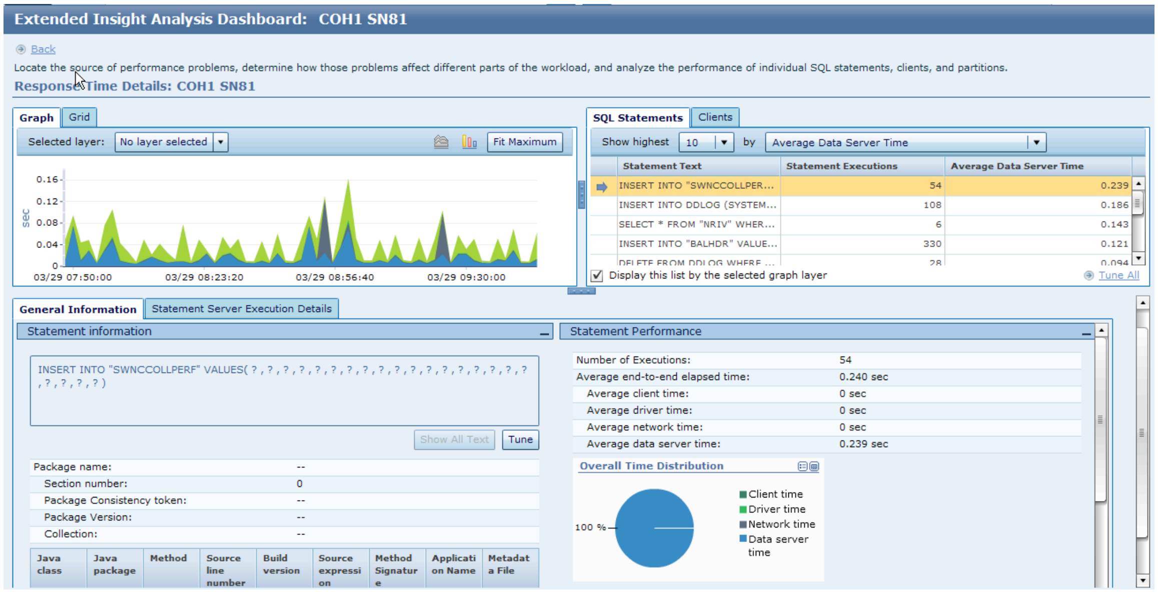
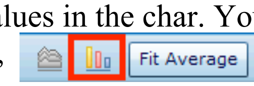


Figure 8 - Response Time Details for selected SQL statement

Besides the general information about the metrics collected by the Extended Insight Client, an additional tab called Statement Server Execution Details is available that, when selected, reveals SQL statement metrics as provided by DB2 for z/OS via the Dynamic and Static Statement Cache. (Figure 9) A DBA can use this view to see an average of how many table or index scans have been performed by this statement, along with the average elapsed and in DB2 CPU times.

Note: The dashboard needs at least two data points to show values in the char. You can use the Bar Chart view option to display "punctual workload." 

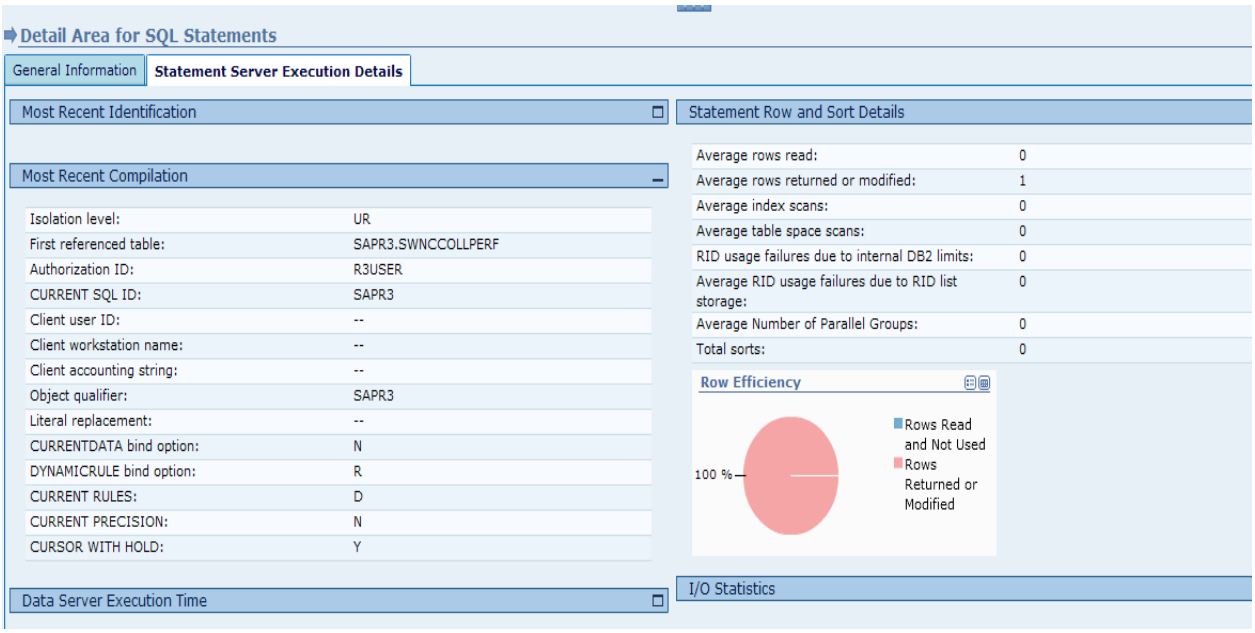


Figure 9 - Statement Execution Server Details

Integrating with additional products for more detail

Besides what IBM provides "out of the box" with OMEGAMON Extended Insight, there are complementary products that support a full application development scenario. This section details some of the optional products that further facilitate communication between the production experience and application development.

Use IBM pureQuery to collect metadata about your monitored Java application

When you look at the Response Time Details for a specific SQL statement in a Workload Cluster Group, you can get very detailed information on where the application calls the SQL statement, i.e. where in the application source code.

Java class	Java package	Method	Source line number	Build version	Source expres ▲ sion	Method Signatu re	Applicatio n Name	Metadata File
EISample	com.ibm.opm.ei	initStatements	287	1.1	--	--	EISample	EISampleCaptu...

Figure 10 - PureQuery metadata in Statement Information section

Figure 10 shows the part of the Response time Details dashboard that contains the pureQuery metadata. The information shown is as granular as the source line number that issues the SQL call statement.

Use IBM Optim Query (Workload) Tuner to optimize your SQL statements

Again, looking at the Response Time Details for a specific SQL statement, you can use IBM Optim Query Tuner (OQT) to do a single SQL tuning and Optim Query Workload Tuner (OQWT) for SQL Workload Tuning.

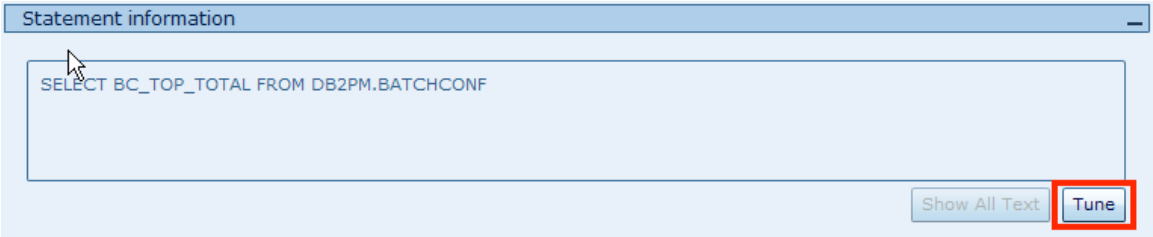


Figure 11 - OQT integration for single SQL tuning

Note: Before pressing the “Tune” button or “Tune All” link, you will have to start up OQT Client in order to start the communication infrastructure (the so-called ”data bridge server”).

When launching OQT, a new OQT project is created automatically and shows the SQL statement or Workloads for further processing in OQ(W)T (see Figure 12) .

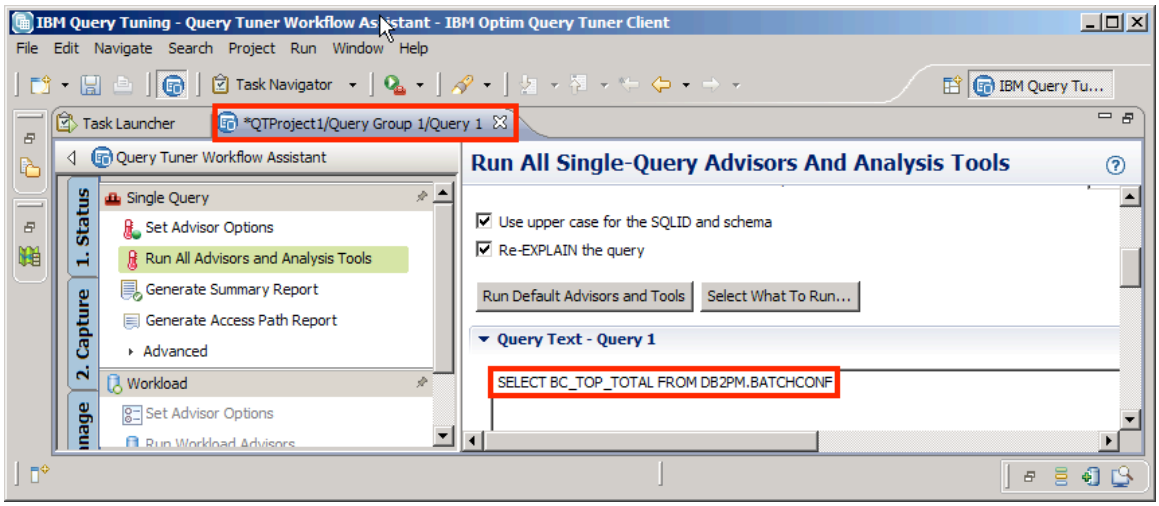


Figure 12 - OQT project containing SQL statement selected

Within the OQT user interface, for example, you can look at the Access Plan Graph or run the Index Advisor. Figure 13 shows the access plan graph.

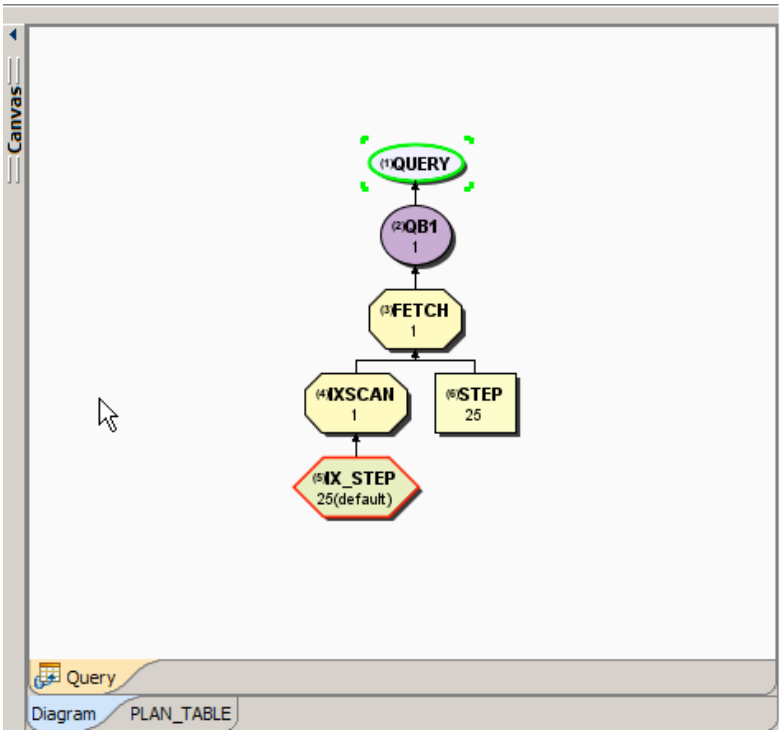


Figure 13 - OQT Access Plan Graph

Note: There is also integration with the free-of-charge tool, IBM Data Studio standalone, that can be used to do single query tuning for SQL statements.

Conclusion

There's no way to avoid (nor is it prudent to do so) the relevance, the importance and the influence that customer demand has on technology. The leaders in the technical community will always leverage a layering or building-block approach in order to preserve the reliability of a known technology upon which to build the newest advances. This initially means increasing complexity for those watching from the inside out, but it supports a more reliable and extensible platform for the future. Extended Insight takes strides to first get the support staff pointed in the right direction, at the right layer. It then provides additional tools to dissect the components we have the most control over -- the application and the SQL.

Additional information

The "**Quick Start Guide for the end-to-end SQL monitoring function**" provides a great overview of the installation process for the OMEGAMON XE for DB2 Performance Expert on z/OS Extended Insight components.

This matrix reflects the depth of Extended Insight support by DB2 version;

	Dynamic SQL (Host)		Static SQL (Host)	
DB2 Version	**SQL statement text**	**SQL cache metrics**	**SQL statement text**	**SQL cache metrics**
DB2 for z/OS v10	Yes	Yes	Yes	Yes (**NFM**)
DB2 for z/OS v9	Yes	Yes	Yes	**No**

Capturing response time details before and after the SQL Unit of Work as referred to in the "**Note"** in the "What is the OMEGAMON Extended Insight Analysis Tool?" section, can be accomplished with IBM Tivoli Composite Application Monitor (ITCAM) for Transactions. OMEGAMON XE for DB2 Performance Expert V5 has the ability to launch ITCAM to view application topology, and ITCAM provides a drill-down link to Extended Insight. More information about ITCAM for Transactions can be found at http://www-01.ibm.com/software/tivoli/products/composite-application-mgr-transactions/

Details about OMEGAMON XE for DB2 Performance Expert can be found at http://www-01.ibm.com/software/tivoli/products/omegamon-xe-db2-peex-zos/

For information on other DB2 for z/OS related tools, please visit http://www-01.ibm.com/software/data/db2imstools/products/db2-zos-tools.html

Note: This Program is licensed only for development and test of applications that run on IBM z/OS. The Program may not be used to run production workloads of any kind, nor more robust development workloads including without limitation production module builds,preproduction testing, stress testing, or performance testing.