Configure and monitor SAP applications with the OMEGAMON DB2 Performance Expert Extended Insight feature

Deploy in SAP shops running DB2 z/OS

Matthias Tschaffler (tschaffl@de.ibm.com)
Senior Software Engineer, OMEGAMON DB2 Performance Expert Technical Development Lead
IBM

Harald Duvenbeck (duven@de.ibm.com)
Software Developer
IBM

Heike Schmidt (heike_schmidt@de.ibm.com)
Software Developer
IBM

Learn the details about the installation and configuration of the IBM Tivoli® OMEGAMON® for DB2® Performance Expert on z/OS® Extended Insight feature in an SAP environment running on DB2 z/OS. This tutorial includes troubleshooting advice.

Introduction

Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS (OMPE) is the premier DB2 z/OS monitoring solution that contains a web-based UI that helps resolve database performance issues with a guided approach.

OMPE’s Extended Insight feature provides end-to-end database performance monitoring for Java™ and call-level interface (CLI) database applications (including .NET applications) using Distributed Relational Database Architecture (DRDA) to connect to DB2 z/OS, providing the ability to quickly understand where your database applications are spending their time. With Extended Insight, DBAs can see and understand where database applications are spending time across the IT stack from the application through the SAP application server, the database client,
the database server, and the network. When a critical business application is not performing as expected, Extended Insight can give end-to-end visibility from the line of code that issues a database statement right through to the database server. This enables you to swiftly isolate and address problems by understanding and identifying where the problem is. When the problem is not in the database itself, Extended Insight also makes it easier for you to work with the people who are managing other parts of the infrastructure or with developers to solve the problem.

Extended Insight recognizes transactions coming from an SAP application server automatically and groups them into *Workload Groups* (SAP report/IDoc message/BW query, SAP transaction/batch job, SAP application servers, SAP users).

**Figure 1. Workload groups**

For all instances of each connection attribute, you can see the breakdown of the average response time into the times which were spent in the SAP application server, in the database client, the network, and in the database server. This helps identify problematic SAP application server machines, specific problematic applications, or SAP users.

This tutorial describes the prerequisites and steps to perform to set up the Extended Insight feature for SAP applications. It focuses on SAP CLI applications (SAP application server ABAP stack) set up with the CLI thin-client connectivity. Additionally, it shows you how you can validate the setup and troubleshoot if necessary.

### Install and activate the Repository Server

Before you set up Extended Insight on your SAP application server instance, you need to install and activate the Repository Server using the license activation kit. Both the Repository Server base and the Activation Kit can be downloaded from Shop Z. It is strongly recommended that you install the Repository Server on a dedicated machine. The license can be activated during installation or by running a separate license activation after installation. (See the [System requirements for IBM InfoSphere Optim Performance Manager for DB2 for Linux, UNIX, and Windows](#).)
During license activation, the Extended Insight communication ports are configured. Make sure you specify unused and open ports. The following figure shows the installer panel used to specify the communication ports for Extended Insight.

**Figure 2. Extended Insight communication port specification**

![Extended Insight communication port specification](image)

You can review the specified communication ports or change them later using the Monitor Server option in the Product Setup section of the Web Console.

**Configuring your DB2 subsystem for Monitoring in the Web Console**

After you have installed and activated the Repository Server, make sure the Repository Server and Web Console are started. Then use the Web Console to configure the databases accessed by your SAP application server for monitoring.

To configure the databases for monitoring, open the Database Connection panel in the Web Console and define a connection to the DB2 subsystem you want to monitor. After the connection definition, you will be asked to acknowledge that you want to enable Historical Monitoring, and the Configure Monitoring panel will open.
In this panel, you will have to fill in the OMPE port number used to communicate with the instance in the OMPE Collector running on z/OS that is in charge to monitor the DB2 subsystem you are currently setting up. Click the **Test Connection** button to ensure that the communication with the z/OS LPAR hosting OMPE/DB2 z/OS is working properly.

Clicking **Next** will take you to Step 2 of the configuration.
Configure and monitor SAP applications with the OMEGAMON DB2 Performance Expert Extended Insight feature

Figure 4. Step 2: Configure monitoring profiles

Ensure that the **Collect Extended Insight data** option is checked.

The next two steps of the configuration wizard are for your information only and clicking **Finish** will start the creation of the necessary artifacts needed to store the DB2 z/OS server side and Extended Insight client-side metrics.

### Planning the Extended Insight client setup

Before you install and configure the Extended Insight Client software on your SAP application server machine, collect the following information.

#### Table 1. Installation and configuration information for Extended Insight Client

| <DIR_GLOBAL> | The SAP system global directory which must be accessible on all servers. Linux and UNIX: /sapmnt/<SAPSID>/global Windows: \%HOME%\sapmnt\<SAPSID>\global where HOME is the name of the host where the global directory is located |
| <SAPSID> | SAP system ID in upper case |
| <sapid> | SAP system ID in lower case |
| <PLATFORM> | Operating System Platform, for example LINUX_S390X_64 |

Prepare the SAP machine by completing the following steps:

- Ensure that you are using the SAP Kernel 7.xx.
- Ensure that the DB2 Runtime Client or DB2 CLI driver is at least of level DB2 V9.7 Fix Pack 6.
- Ensure that the `db2dsdriver.cfg` file is centrally available in directory `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/cfg`
- Check the prerequisites listed in SAP note 2011054 if you would like to use the `db2dsdriver.cfg` based failover mechanism instead of the `connect.ini` based failover mechanism.
**Note:** During Extended Insight configuration, additional parameters have to be added to the `db2dsdriver.cfg` configuration file.

**Client connectivity scenarios**
The SAP application server ABAP (AS ABAP) uses the DB2 CLI API, whereas the SAP application server Java (AS Java) uses the JDBC API to access the database.

SAP systems based on releases below SAP NetWeaver 7.0 SR3 used a DB2 client setup, which is referred to as *fat client* connectivity. In this scenario, every AS ABAP had its own DB2 Runtime Client installed. All of these clients had to be maintained separately.

Starting with SAP NetWeaver 7.0 SR3, a new DB2 client setup called *thin client* connectivity was introduced. The DB2 CLI driver is used for AS ABAP instead of the DB2 Runtime Client. The CLI Driver is installed once in a directory shared by all application servers of the SAP system. During the start of an application server, the CLI driver is copied to a local directory. For the DB2 JDBC driver of the AS Java, the same mechanism is used.

This paper will concentrate on the thin client connectivity and AS ABAP.

**Client Affinity mechanism and DB2 Connect client version**
If you are using client failover, there are two flavors to consider: the `connect.ini`-based failover mechanism and the `db2dsdriver.cfg` based failover mechanism.

To exactly determine which failover mechanism is currently used by an SAP application server instance, complete the following steps:

- Log on to the application server on which you want to check the DB2 client setup as user `<sapsid>adm`.
- Enter the following command:
  ```
  R3trans -d
  ```

  A log file `trans.log` is created in the current directory

  - Open the `trans.log` file in an editor

  If the string *we use the DB2 Connect fail over feature* shows up, the `db2dsdriver.cfg` based failover mechanism is used. If the string *we use the SAP fail over feature* shows up, the `connect.ini` based failover is used. The DB2 Connect client version is also shown in the `trans.log` file. Search for string `DB2 Connect driver version`.

  The failover mechanism used is important once we configure the Extended Insight Client.

**Extended Insight Client setup**

**Extended Insight Client installation and configuration**
For an SAP environment using the thin client, it is recommended to embed the Extended Insight Client software into the SAP `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver` directory, hence the libraries needed are copied to the local exe directory together with the DB2 CLI driver.
Starting with clients at level 9.7 FP6, the DB2 client looks up the Extended Insight client libraries in a default directory of the DB2 Client installation. This tutorial will describe how to use this mechanism. In the unlikely case you are using DB2 clients older than 9.7.FP6, the Extended Insight client library location needs to be explicitly specified in the db2dsdriver.cfg file.

**Installation of Data Tools Runtime Client (Extended Insight Client)**

The Data Tools Runtime Client (short DTRC) is a software bundle used for the OMPE Extended Insight feature and the Optim Configuration Manager.

Complete the following steps to install the DTRC on an SAP application server. In the following, we assume the SAP application server is running on a Linux® x86 64-bit OS:

1. Download and unpack the respective DTRC driver.
   You should see the following files:

   ```
   ./IBM.DTRC.v3.1.0.IFIX.install-on-linux-x86-64.bin -i console
   ```

2. Launch the console mode installation as an `<sapsid>adm` user by executing the following command from the directory of the Extended Insight installation image. The SAP user has the name `<sapsid>adm`.

   ```
   ./IBM.DTRC.v3.1.0.IFIX.install-on-linux-x86-64.bin -i console
   ```

3. Proceed through the Locale selection, the Welcome panel and the Software License Agreement panel.

4. Select option 1 to install the product and specify `/home/<sapsid>adm/pqcmx` as the installation directory.

5. Review the information shown on the Pre-Installation Summary panel and confirm to proceed with the installation.

   ```
   You choose to:
   1- Install a new product.

   IS THIS CORRECT? (Y/N) Y
   ```

   It will take a few minutes for the installation to complete. You will see an indicator that shows the progress. Press Enter to exit the installation.
Configuration of the Data Tools Runtime Client

To configure the DB2 client for Extended Insight monitoring, you need to copy the extended Insight client into your global db2_clidriver directory and create a new or modify an existing db2dsdriver.cfg file:

1. Copy directory `/home/<sapsid>/adm/pqcmx` to `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver`

   The directory name `pqcmx` is also important as this is the default directory the DB2 client is looking to load the Extended Insight client libraries.

2. Optionally, you can remove all subdirectories except the pureQuery® subdirectory in `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/pqcmx`

   This will avoid copying unnecessary files once the SAP application server starts up.

3. Locate the `db2dsdriver.cfg` file. The file can be found in one of the following locations:
   a. `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/cfg`
      In case there is no `db2dsdriver.cfg` file here, there will be a `db2dsdriver.cfg.sample` file which can be copied and used in the following steps as well.
   b. In case you already use the cfg-file for your DB2 Connect-based client failover, the environment variable DB2DSDRIVER_CFG_PATH in the environment of the `<sid>/adm` user points you to the directory. As a default this directory is `<DIR_GLOBAL>`.

4. Modify the `db2dsdriver.cfg` file
   a. Open the file in an editor and ensure you have write permission.
   b. Locate the section `<parameters>` toward the end of the file and add the following parameter name line in between the `<parameters>` and `</parameters>` tags:

   ```xml
   <parameters>
   <parameter name="connectionSupervisorProperties" value="controllerURL=RepServer.ibm.com:65000/"></parameters>
   ```

   c. In case you are using a DB2 client older than 9.7 FP6, you need to add the parameter `connectionSupervisorLibrary` in addition (note the platform dependency):

   ```xml
   <parameters>
   <parameter name="connectionSupervisorProperties" value="controllerURL=RepServer.ibm.com:65000/"></parameters>
   ```

   Unix/Linux 64 bit:
   ```xml
   <parameter name="connectionSupervisorLibrary" value="<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/pqcmx/pureQuery/lib64/pqcmx"></parameter>
   ```

   Windows:
   ```xml
   <parameter name="connectionSupervisorLibrary" value="<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/pqcmx/pureQuery/bin/pqcmx"></parameter>
   ```

   d. The following shows the modification in the total file context. The value for "controllerURL=" is pointing to the Repository Server name/IP address and the configured controller port as shown in Figure 1.

   ```xml
   <?xml version="1.0" encoding="UTF-8"?><configuration>
   <dsncollection>
   <dsn alias="alias1" host="server1.net1.com" name="name1" port="50001"/>
   <dsn alias="longaliasname2" host="server2.net1.com" name="name2" port="55551">
   <parameter name="Authentication" value="SERVER_ENCRYPT"></parameter>
   </dsn>
   </dsncollection>
   <databases>
   <database host="server1.net1.com" name="name1" port="50001">
   ...
e. Save the file and close the editor.

5. The configuration is now complete.

Special considerations when using private networks

It is quite common that SAP application servers do use a private network to communicate to DB2. For example on zLinux, a private network using HiperSockets is a common setup scenario, as it will ensure the best security and minimal network latency.

If your Repository Server is also installed on zLinux and is in the same private network as the SAP application server and DB2, no special measures need to be taken.

If your Repository Server is on a different network (for example, on an AIX® or Windows® machine outside of the closed HiperSockets-based network), the following special setup needs to be established in order for the Extended Insight clients to being able to find the correct monitored data source on the Repository Server.

Let’s illustrate the scenario with the following picture.

**Figure 5. SAP application servers using private network**

The trick here is to have the hosts file of the Repository Server map the HiperSockets-based addresses to an IP address that is known and can be reached by the Repository Server for the subsystem being monitored (the IP address of the z/OS LPAR used when configuring the connection to DB2 in the Web Console —db2lpar.ibm.com in our example).
In our example, the following line needs to be added to the `/etc/hosts` file of the Repository Server machine.

```
9.152.100.5 db2lparv.ibm.com
```

When the clients establish a connection with the Repository Server controller (using the `controllerURL`), in our example, they use the triplet `db2lparv.ibm.com:DB2DDF:446` to find a corresponding so called "monitoring instance" on the Repository Server to transfer the client data to. Without the hosts file mapping mentioned above, no such communication or handshake can be established, and no monitoring data will be transferred.

Furthermore, if you are running your SAP in a DB2 Data Sharing environment, you need to map all private IP addresses of your z/OS LPARs to the IP address you used to configure to configure the DB2 member for monitoring. This is necessary, as the Data Sharing group is seen as a single logical database within the Web Console and Repository Server.

**Configuration steps at the SAP application server**

After you finished the installation and configuration of the Extended Insight Client, you must reload the changes into CLI. If you use the connect.ini based failover mechanism you have to restart the SAP application server. If you use the `db2dsdriver.cfg` based failover mechanism the changes can be loaded dynamically for all SAP application servers out of the SAP application transaction "DBAcockpit", using the function "Relaod into CLI" (details please see "Database Administration Guide: SAP on IBM DB2 for z/OS").

OMPE Extended Insight gets access to the information that SAP sends to the database. Within this information, there is SAP-specific information like the SAP job name, the transaction number etc. To be able to display this information on the Extended Insight Dashboard, you have to set the following profile parameter in SAP:

```
dbs/db2/use_accounting = 1
```

This can be done in each instance profile or in the default profile of the SAP system.

**Validating your Extended Insight installation and configuration**

To perform the validation, follow these steps:

1. Ensure that your database is configured for Extended Insight monitoring.
   For this tutorial, we assume the following database settings:
   a. DB2 DDF location name: `DB2DDF`
   b. DB2 port number: `446`
   c. LPAR name hosting DB2: `db2lparv.ibm.com`
2. As `<sapsid>adm` change to directory `<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver/bin`
3. Before running the `db2cli` command on Linux or UNIX (next step), you might have to adapt environment variables. Try the command first and if it fails, set the `DB2_CLI_DRIVER_INSTALL_PATH` variable to the path of the DB2 client installation (`<DIR_GLOBAL>/db2/<PLATFORM>/db2_clidriver`).
If you are using the New Client Connectivity for which the `db2cli.ini` file and the `db2dsdriver.cfg` file are located in the directory `<DIR_GLOBAL>`, you have to set the `DB2_CLI_DRIVER_INSTALL_PATH` and `DB2CLIINIPATH` variables to the `<DIR_GLOBAL>` directory as well.

4. Using the example values from step 1. above, enter:
   ```bash
db2cli validate -database DB2DDF:db2lpar.ibm.com:446
```

5. Review the output of the command for errors (see the troubleshooting section).

6. Change to the home directory of `<sid>adm` and run `R3trans -d`, which should return with a code of 0. Check the Extended Insight Dashboard for data displayed (this does not require an SAP application server restart).

### Browsing the Extended Insight monitoring data for SAP applications

#### Activating SAP workload groups

Ensure that your configured SAP applications are running and executing database transactions, and that the Repository Server and Web Console are started. If you successfully installed and configured the Extended Insight client, you can see the collected monitoring data displayed on the Extended Insight dashboard in the Web Console.

On the Extended Insight dashboard, you can use workload groups to view transaction response time data by application, by user, or by server. Predefined workload groups are available to specify different groupings. To activate the SAP-specific predefined workload groups, select the **Activate Workload Groups** entry in the **More Actions** pull-down menu of the Extended Insight Dashboard and activate the respective SAP Workload Groups as shown in the following figure.

![Figure 6. Activating SAP workload groups](image)

By clicking **New**, you can add your own workload group that clusters your SAP application the way you want it to be clustered. In step 1, you specify the name of the workload group. In step 2, you should select **SAP on System Z** as type of workload group as shown in the following figure.
Select the clustering attributes of your choice, preview the created clusters based on the data collected so far (the Refresh button) and finally save the new workload group.

Viewing data in the Extended Insight dashboard

After you activate the SAP workload groups, you see your SAP applications represented in different Workload Groups as shown in the following figure. The response time columns show you the time spent metrics per SAP user, for example, so that you can identify the SAP user with the highest response time easily and determine where the most time is spent. The color coded bar chart gives you a high level understanding on where the time is spent. The blue part represents the time spent in DB2, the grey part the time spent in the network and the green part the time spent in the application or DB2 driver processing.

For example, in the following figure, you can see that the transactions run by user heikes spent on average 49 percent of their time in DB2, 23 percent in the network, and 26 percent in the application. To see the absolute values for the response-time breakdown, you can refer to the respective columns on the Dashboard Overview.
Figure 9. Extended Insight Dashboard Overview

Further analysis of the response-time metrics includes drill-down to more detailed time-spent metrics, and to the executed SQL statements and their time distribution and execution details.

Figure 10. Response-time details drill-down

Troubleshooting the Extended Insight setup

If after a few minutes of monitoring you do not see any data in the Extended Insight Dashboard, this is an indication that no Extended Insight data from your SAP applications is arriving at the Repository Server.

Verify the following items to find the cause for this problem:

- Ensure that you restarted/reloaded the SAP application server with the configuration changes.
- Ensure that the Extended Insight controller server is started within the Repository Server:
  - Check the Repository Server log db2pesrv.log for the following message (port 65000 is used as an example):
    ```
The Extended Insight controller server is started on port 65000.
```
• Ensure that the Extended Insight client is using the same port number for the controller server:
  • Check the `db2dsdriver.cfg` file in `<DIR_GLOBAL>` for the parameter `<parameter_name="connectionSupervisorProperties" value="controllerURL=RepServer.ibm.com:65000"/>
  • Check the `db2cli validate -database` command output for possible errors (see separate section).
  • Ensure that the `pqcmx` library can be loaded by checking the `db2diag.log` file as described (see separate section).

**db2cli validate output**

This is a sample output for a `db2cli validate` command followed successfully:

```
sapAppSrVr01:R03adm 61> db2cli validate -database DB2DDF:db2lpar.ibm.com:446
IBM DATABASE 2 Interactive CLI Sample Program
(C) COPYRIGHT International Business Machines Corp. 1993,1996
All Rights Reserved
Licensed Materials - Property of IBM
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA AOP Schedule Contract with IBM Corp.
---------------------------------------------------------------------------
[ CLI Driver Version : 10.01.0000 ]
[ Informational Tokens: "DB2 v10.1.0.2","special_31876","IP23394_31876","Fixpack 2" ]
[ CLI Driver Type : IBM Data Server Driver For ODBC and CLI ]
[ db2diag.log Path : /usr/sap/R03/SYS/global/db2/LINUX_X86_64/db2_clidriver/db2dump/db2diag.log ]
---------------------------------------------------------------------------
db2dsdriver.cfg Schema Validation :
Success: The schema validation operation completed successfully.
The configuration file /usr/sap/R03/SYS/global/db2/LINUX_X86_64/db2_clidriver/cfg/db2dsdriver.cfg is valid

CSC Pre-Connection Information Section :
--------------------------------------------------------------------------------
PQCMX is attempting to connect to a controller server using the controllerURL property fixed address:
PQCMX successfully connected to a controller server using the controllerURL property fixed address:
PQCMX datasource db2lpar.ibm.com:446:DB2DDF attempting to obtain properties using 3-part client specified
address.
PQCMX datasource db2lpar.ibm.com:446:DB2DDF obtained properties based on 3-part client specified address.
End CSC Pre-Connection Information Section
--------------------------------------------------------------------------------

db2cli.ini Validation :
--------------------------------------------------------------------------------
[ DB2CLIINIPATH env var : /sapmnt/R03/profile ]
[ db2cli.ini Path : /sapmnt/R03/profile/db2cli.ini ]
--------------------------------------------------------------------------------
[ Keywords used by CLI for Database: DB2DDF ]
Keyword                                      Value
---------------------------------------------
DIAGLEVEL                                    4
--------------------------------------------------------------------------------

db2dsdriver.cfg Validation :
--------------------------------------------------------------------------------
[ DB2DSDRIVER_CFG_PATH env var : unset ]
[ db2dsdriver.cfg Path : /usr/sap/R03/SYS/global/db2/LINUX_X86_64/db2_clidriver/cfg/db2dsdriver.cfg ]
--------------------------------------------------------------------------------
Consider the following when troubleshooting your setup:

- If the section CSC Pre-Connection Information Section is completely missing, the Extended Insight Client libraries are not loaded at all. The reason can be that there is no pqcmx directory copied or the connectionSupervisorLibrary property points to an invalid location.

- If the section CSC Pre-Connection Information Section contains the following highlighted line, there was a problem with the 3-part client specified address. In this case, the wrong port number had been specified:

The reason for this could be a wrong controller port or a firewall blocking the traffic.
CSC Pre-Connection Information Section:

PQCMX is attempting to connect to a controller server using the controllerURL property fixed address: RepServer.ibm.com:65000 with protocol 8.
PQCMX successfully connected to a controller server using the controllerURL property fixed address: RepServer.ibm.com:65000 with a negotiated protocol: 8.
PQCMX datasource db2lpar.ibm.com:446:DB2DDF attempting to obtain properties using 3-part client specified address.
PQCMX datasource db2lpar.ibm.com:446:DB2DDF was unable to obtain properties using 3-part client specified address.
End CSC Pre-Connection Information Section

This results also in the Monitoring status: off entry of the CSC Information Section : section of the validate output.

**db2diag.log**

In addition to the db2cli validate command, the db2diag.log can be used to verify that the correct db2dsdriver.cfg file used and that the Extended Insight library is loaded:

1. Set the DB2 diagnostic trace to level 4 by adding the following line to the common section of the db2cli.in file:

   
   ```
   [COMMON]
   DiagLevel=4
   ```

2. If you run the command `R3trans –d`, you will see the following messages in the db2diag.log file if the library could not be loaded or initialized correctly:

   ```
   2014-04-09-09.30.51.244740+120 I1025259A367         LEVEL: Error
   PID     : 62403                TID : 4397962241808  PROC : db2cli
   INSTANCE: DB2                  NODE : 000
   HOSTNAME: ihlscoh1
   FUNCTION: DB2 UDB, base sys utilities, sqle_cscInvokeInitialize, probe:0
   DATA #1 : <preformatted>
   CSC Initialization failed. Requested CSC API major level: 6, minor level: 1
   ```

3. Furthermore, the following message indicates the location from which the db2dsdriver.cfg is loaded (for R3trans it is always loaded from the global directory, not the instance directory).

   ```
   2014-07-29-14.52.54.072125-420 E7003A453            LEVEL: Info
   PID     : 63300                TID : 4397996140304  PROC : R3trans
   INSTANCE: DB2                  NODE : 000           DB   : DB2DDF
   APPID   : 192.168.32.227.31700.140729215254
   HOSTNAME: ihlscohv
   FUNCTION: DB2 UDB, call level interface, CLI_utlGetCfgPath, probe:10
   DATA #1 : <preformatted>
   db2dsdriver.cfg path: /usr/sap/R03/SYS/global/db2/LINUX_S390X_64/db2_clidriver/cfg/db2dsdriver.cfg
   ```

4. The load of the Extended Insight library is successful, and the communication to the Repository Server EI controller is successful if the following messages show up:

   ```
   2014-07-31-12.47.44.267343+120 I1295555A515         LEVEL: Info
   PID     : 45183                TID : 4397794912528  PROC : R3trans
   INSTANCE: DB2                  NODE : 000
   HOSTNAME: ihlscoh1
   FUNCTION: DB2 UDB, call level interface, CSCLogMessage, probe:10
   DATA #1 : String, 229 bytes
   [PQCMX '10.5.0.0' 's130526' 'LINUX390X64105' '0'][File cmxcs.C, Line 601] PQCMX is attempting to connect to a controller server using the controllerURL property fixed address: RepServer.ibm.com:65000 with protocol 8.
   ```

   ```
   2014-07-31-12.47.44.296125+120 I1296071A527         LEVEL: Info
   PID     : 45183                TID : 4397794912528  PROC : R3trans
   ```
INSTANCE: DB2                  NODE : 000
HOSTNAME: ihlscohi
FUNCTION: DB2 UDB, call level interface, CSCLogMessage, probe:10
DATA #1 : String, 241 bytes
[PQCMX '10.5.0.0' 's130526' 'LINUX390X64105' '0'][File cmxcs.C, Line 621] PQCMX successfully connected to a controller server using the controllerURL property fixed address: RepServer.ibm.com:65000 with a negotiated protocol: 8.

2014-07-31-12.47.44.313809+120 I1296599A496 LEVEL: Info
PID     : 45183                TID : 4397794912528  PROC : R3trans
INSTANCE: DB2                  NODE : 000
HOSTNAME: ihlscohi
FUNCTION: DB2 UDB, call level interface, CSCLogMessage, probe:10
DATA #1 : String, 210 bytes
[PQCMX '10.5.0.0' 's130526' 'LINUX390X64105' '0'][File cmxdisser.C, Line 538] PQCMX datasource db2lparv.ibm.com:446:DB2DDF attempting to obtain properties using 3-part client specified address.

5. If you see the following message in the log, it is a strong indication that there is a problem with the setup on the Repository Server (see special considerations for private networks).

2014-07-31-12.47.44.373764+120 I1297824A490 LEVEL: Info
PID     : 45183                TID : 4397794912528  PROC : R3trans
INSTANCE: DB2                  NODE : 000
HOSTNAME: ihlscohi
FUNCTION: DB2 UDB, call level interface, CSCLogMessage, probe:10
DATA #1 : String, 204 bytes
[PQCMX '10.5.0.0' 's130526' 'LINUX390X64105' '0'][File cmxdisparser.C, Line 579] PQCMX datasource db2lparv.ibm.com:446:DB2DDF obtained properties based on 3-part client specified address.

Conclusion

By following the steps outlined here, you have learned how to install the Extended Insight client and configure it to monitor the transaction and SQL statement response times of your SAP applications, including the time-spent breakdown across the IT stack. The Extended Insight feature can help DBAs collaborate more efficiently with other IT staff by providing a web-based UI that provides overview, alert, and diagnostic dashboards to simplify root-cause analysis for SAP applications that access DB2 data.
Resources

Learn

- Read the developerWorks article "Developing enterprise OSGi applications for WebSphere Application Server."
- Learn more about Context Class Loader Enhancements.
- Visit the developerWorks Information Management zone to find more resources for DB2 developers and administrators.
- Stay current with developerWorks technical events and webcasts focused on a variety of IBM products and IT industry topics.
- Follow developerWorks on Twitter.
- Watch developerWorks on-demand demos ranging from product installation and setup demos for beginners, to advanced functionality for experienced developers.

Get products and technologies

- Evaluate IBM products in the way that suits you best: Download a product trial, try a product online, or use a product in a cloud environment.

Discuss

- Get involved in the developerWorks community. Connect with other developerWorks users while exploring the developer-driven blogs, forums, groups, and wikis.
About the authors

**Matthias Tschaffler**

Matthias Tschaffler is a senior software engineer at IBM Deutschland Research & Development GmbH, Böblingen. He holds a master's degree in computer science from the Technical University of Munich, Germany, and has worked for IBM since 1999. In 2004 he joined the IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS team where he held different positions as a developer and is currently the technical development lead, also engaging with customer all over the world. Matthias co-architected the OMPE Extended Insight feature and is closely working with several SAP customers to demonstrate the value of this feature.

---

**Harald Duvenbeck**

Harald Duvenbeck is a software developer at IBM Deutschland Research & Development GmbH, Böblingen. He holds a master's degree in mathematics from the of Bonn, Germany and has worked for IBM since 1990. In 1994, he the IBM/SAP platform team that developed the SAP on System z solution. Since 2008, he has been working in the IBM Team in the Böblingen lab that focuses on Availability for SAP on IBM server platforms.

---

**Heike Schmidt**

Heike Schmidt has worked for IBM since 1986, starting as an application programmer. Since 1998, she has worked as an SAP Basis Specialist, running tests on IBM platforms and supporting for SAP basis problems inside IBM (SAP Customer Competence Center). She was involved in several SAP projects and benchmarks for customers. Since 2007, she is in the IBM System z Technology Center for SAP applications in the Böblingen Lab, where she focuses on z/OS and Linux on System z certification for SAP applications and on High Availability for SAP on System z.

© Copyright IBM Corporation 2014
**Trademarks**
(www.ibm.com/developerworks/ibm/trademarks/)