# IBM Integration Bus

# Global Cache using a DataPower XC10 appliance

June, 2013

Hands-on lab built at product code level Version 9.0.0.0

# 1. Introduction to Integration Bus Global Cache

WebSphere Message Broker V8.0.0.1 (fix pack 1), provided a new feature called Global Cache. IBM WebSphere Message Broker V8.0.0.2 (fix pack 2) further extended this feature to provide an automatic data revocation feature. IBM Integration Bus V9.0.0.0 provides these features without the need to activate them using the `mqsichangebroker` command.

Global Cache allows a number of Integration Nodes to be integrated for workload balancing. Global Cache provides the ability to have a cache storing information about the requester which would later be used to correlate the replies correctly. Using Global cache gives the flexibility for different Integration Nodes to handle the request and reply parts of an application. Using a global cache allows you to scale up the number of clients, while maintaining predictable response time for each client.

Global Cache uses an embedded version of WebSphere eXtreme Scale, and provides the following functions:

- 'Elastic' In-Memory Data Grid – managing itself to scale out, scale in, failover, failure etc.
- Virtualizes free memory within a grid of Java Virtual machines (JVMs) into a single logical space which is accessible as a partitioned, key addressable space for use by applications
- Provides fault tolerance through replication with self-monitoring and healing
- The space can be scaled out by adding more JVMs while its running without restarting
- Provides a predictable scaling option
- Access to external WebSphere Xtreme Scale data grids

IBM Integration Bus Version 9.0.0.0 contains support for:

1) An embedded WebSphere eXtreme Scale (WXS) grid which can be used as a global cache from within message flows.

   WXS components are hosted within Integration Server processes and operate with no requirement for additional configuration. The default scope of one cache is across one Integration node (i.e. multiple Integration Servers) but it can be extended to be across multiple Integration nodes. Integration Bus support for embedded WebSphere Extreme Scale grids is covered under the Lab Guide "**IBM Integration Bus, Version 9.0.0.0, Embedded Global Cache**" from the same series of Lab Guides.

2) Access to WebSphere Xtreme Scale Grids that are running outside of the Integration Node for example a DataPower XC10 appliance.

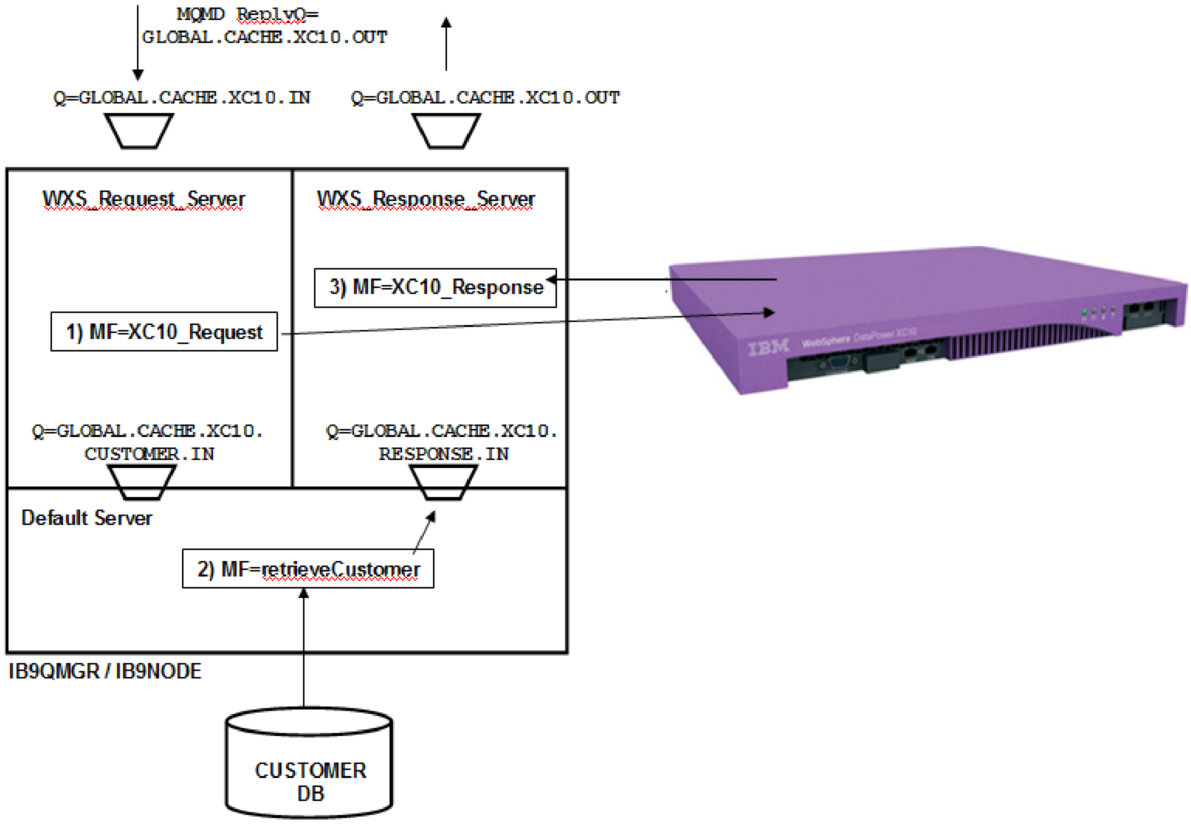
This lab guide demonstrates a simple setup / configuration of an application using a DataPower XC10 appliance.

This lab guide has been written with the XC10 firmware at V2.1. XC10 Firmware at V2.0 and V2.5 will also work when communicating with IBM Integration Bus V9.0. A small change in the XC10 configuration is required when using XC10 V2.5 firmware. This change is documented in Appendix A of this guide. Some screen captures may appear slightly different when using the V2.5 level of the firmware.

# 2. Scenario

The XC10 Global cache scenario is based on the Integration Bus embedded Global Cache scenario and demonstrates the configuration required to use the external DataPower appliance.

The following schematic shows the system context of the Global Cache Lab.



There are three message flows in this scenario.

1. **XC10_Request** – this flow receives messages on the `GLOBAL.CACHE.XC10.IN` queue. The application that puts messages on this queue also specifies the ultimate reply queue, which will be set to `GLOBAL.CACHE.XC10.OUT` in this example.

   The Request message flow takes the MessageID and ReplyToQ from the incoming message, and stores them in an XC10 appliance data grid. This is so that the Response message flow will be able to use this data to send the reply back to the specified reply queue, even though Response may be running in a different Integration Server, or Integration Node.

   The Request flow has a user-defined property, which specifies the name of the MQ queue that the retrieveCustomer flow will send the resulting message to. This value is used to populate the ReplyToQ that will be used by retrieveCustomer. This has been done for ease of illustration of the Global Caching component, when switching the demo between multiple Integration Servers and Integration Nodes.

2. **retrieveCustomer** – this flow takes the input message (passed by Request) and uses the data to retrieve a customer record from the CUSTOMER database. This is done with a simple Compute node. It then sends the output message to the specified Reply queue. This flow is the same flow used for the embedded global cache application and has nothing specific to the XC10 appliance.

3. **XC10_Response** – this flow runs in both brokers, and reads the `GLOBAL.CACHE.RESPONSE.XC10.IN` queue. It uses the MessageID to retrieve the required final reply queue name from the global cache, and sends the message to the final output queue.

# 3. Preparation for this Lab Guide

This lab guide uses IBM Integration Bus and a DataPower XC10 appliance. The following sections outline the preparation for IBM Integration Bus and options for using an external DataPower XC10 appliance.

## 3.1 IBM Integration Bus preparation

In the pre-built vmware, this section has been done for you. If you have previously done the embedded Global Cache Lab in this series, the tables and queues will also have been set up. **You can ignore this section.**

To run this lab, unzip the supplied file `global_cache.zip` into the directory `c:\student` directory. This will create a subdirectory called `\global_cache` with several further subdirectories.

**Create the MQ Queues**
In the `c:\student\global_cache\install` folder, run the command `createCacheQueues.bat.`

**Create the databases**
In a Integration Command Console, go the folder `c:\student\global_cache\install`.

Issue the commands:

```
CreateRegularDB
CreateRegularTables

SetBrokerSecurityForCaching

mqsistop IB9NODE
mqsistart IB9NODE
```

## 3.2 Connectivity to a DataPower XC10 appliance

This lab guide shows you how to configure IBM Integration Bus V9.0 to communicate and use a DataPower XC10 appliance. There are two possible choices of XC10 device when communicating with IBM Integration Bus V9.0:

1) Use a ***virtual*** XC10 DataPower appliance.

2) Use a ***physical*** XC10 DataPower appliance.

The following sections will outline the necessary pre-req set up when using both of these options.

## 3.2.1 Using a Virtual DataPower XC10 appliance

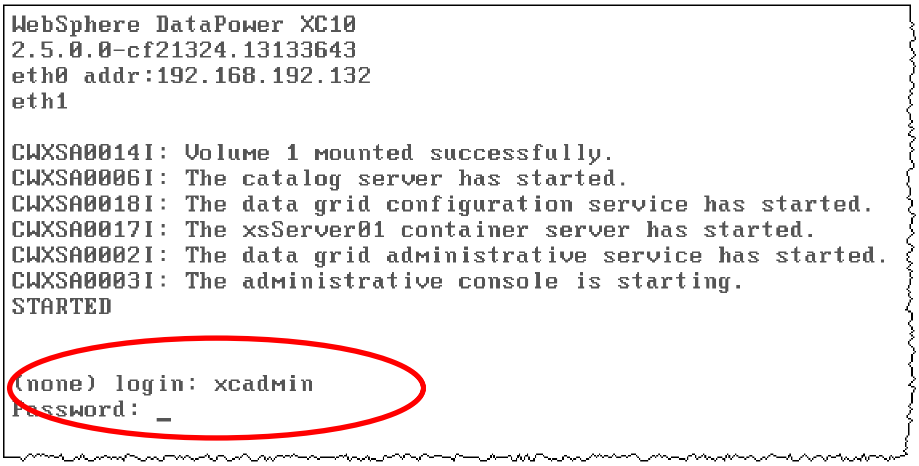
**If you are using a *physical* DataPower XC10 appliance you can ignore this section and go to the next section entitled “Using a physical DataPower XC10 appliance”.**

### 3.2.1.1 Verify the XC10 appliance ip address

1. Focus your keyboard input to the virtual appliance console by clicking the (black) virtual XC10 console screen. (*There may be initialization messages on the screen from the appliance start up process*)

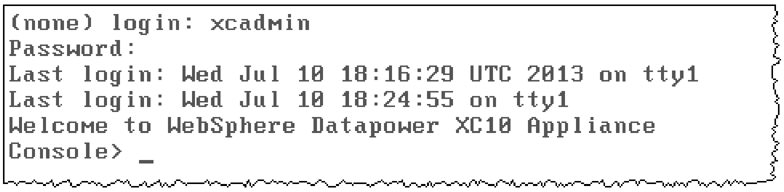
   Press the <enter key>, a login prompt will appear.

2. login using `xcadmin`, with a password of `xcadmin` :



```
WebSphere DataPower XC10
2.5.0.0-cf21324.13133643
eth0 addr:192.168.192.132
eth1

CWXSA0014I: Volume 1 mounted successfully.
CWXSA0006I: The catalog server has started.
CWXSA0018I: The data grid configuration service has started.
CWXSA0017I: The xsServer01 container server has started.
CWXSA0002I: The data grid administrative service has started.
CWXSA0003I: The administrative console is starting.
STARTED


(none) login: xcadmin
Password: _
```

3. When you are successfully signed on to the appliance you will see something similar to the following:

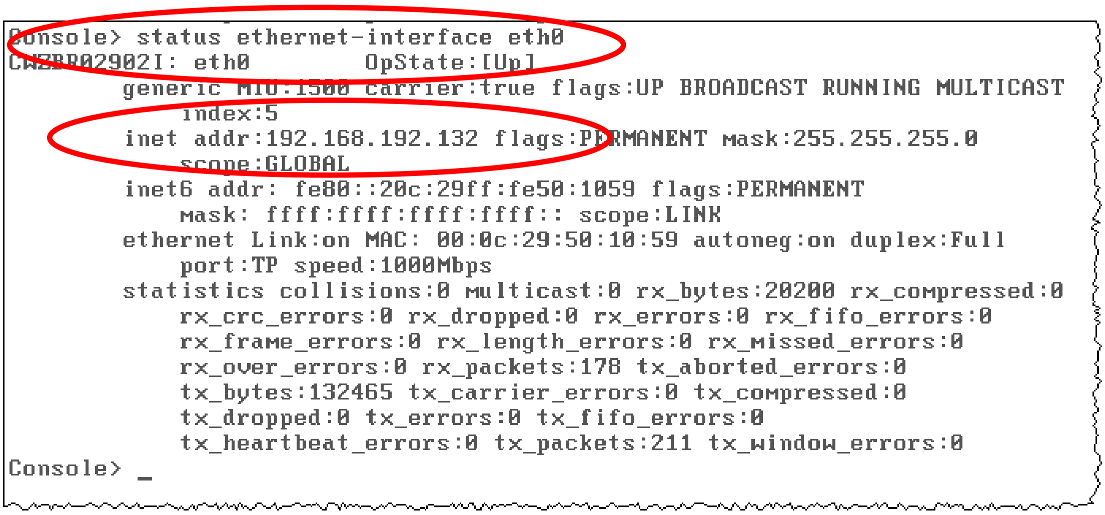


```
(none) login: xcadmin
Password:
Last login: Wed Jul 10 18:16:29 UTC 2013 on tty1
Last login: Wed Jul 10 18:24:55 on tty1
Welcome to WebSphere Datapower XC10 Appliance
Console> _
```

4. At the console prompt type the command:

“status ethernet-interface eth0”

The command will respond with the following:



```
Console> status ethernet-interface eth0
CWZBR02902I: eth0        OpState:[Up]
        generic MTU:1500 carrier:true flags:UP BROADCAST RUNNING MULTICAST
            index:5
        inet addr:192.168.192.132 flags:PERMANENT mask:255.255.255.0
            scope:GLOBAL
        inet6 addr: fe80::20c:29ff:fe50:1059 flags:PERMANENT
            mask: ffff:ffff:ffff:ffff:: scope:LINK
        ethernet Link:on MAC: 00:0c:29:50:10:59 autoneg:on duplex:Full
            port:TP speed:1000Mbps
        statistics collisions:0 multicast:0 rx_bytes:20200 rx_compressed:0
            rx_crc_errors:0 rx_dropped:0 rx_errors:0 rx_fifo_errors:0
            rx_frame_errors:0 rx_length_errors:0 rx_missed_errors:0
            rx_over_errors:0 rx_packets:178 tx_aborted_errors:0
            tx_bytes:132465 tx_carrier_errors:0 tx_compressed:0
            tx_dropped:0 tx_errors:0 tx_fifo_errors:0
            tx_heartbeat_errors:0 tx_packets:211 tx_window_errors:0
Console> _
```

The IP address of the appliance is shown by the “**inet addr**” prefix (in the above example **192.168.192.132**)

Write down the IP address of the appliance in your environment here:

XC10 IP Address: < ________________________________________>

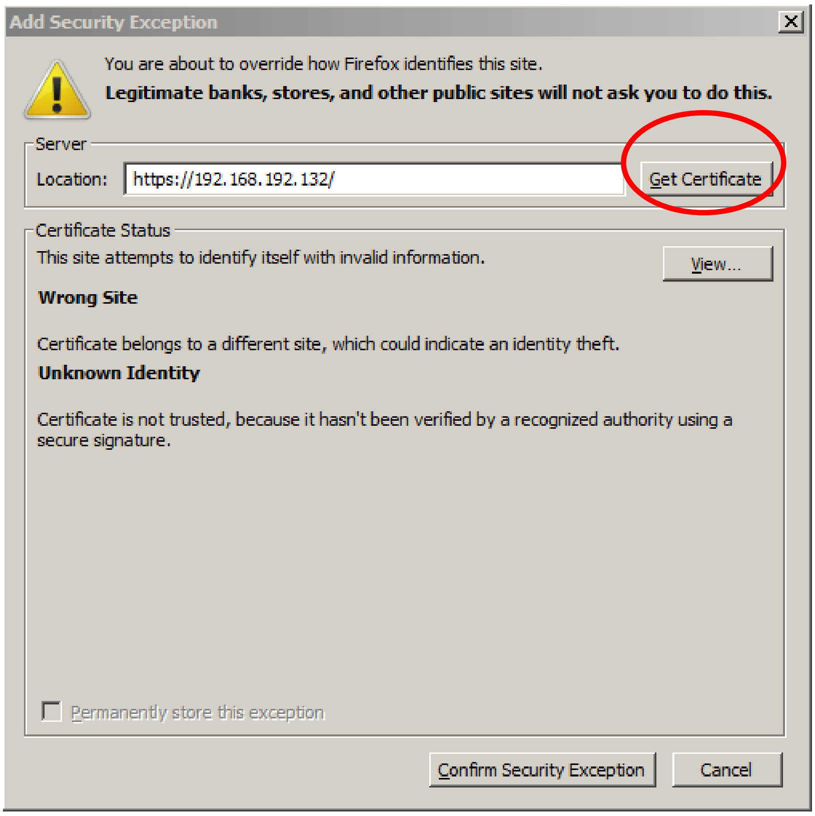
### 3.2.1.2 Define a non-administrator XC10 user

5. Switch back to the IBM Integration Bus environment and open a firefox browser. You may need to release the cursor from the virtual appliance console using <Ctrl> and <Alt> before you can switch to the IBM Integration Bus environment.

6. Enter the IP address of the virtual XC10 appliance in the Firefox browser.

You will be prompted that the connection is “Untrusted”, click “I understand the Risks” and the “Add Exception” button.

In the “Add security Exception” window, Click the “Get Certificate” button.



You are about to override how Firefox identifies this site.

**Legitimate banks, stores, and other public sites will not ask you to do this.**

Server

Location: https://192.168.192.132/

Certificate Status

This site attempts to identify itself with invalid information.

**Wrong Site**

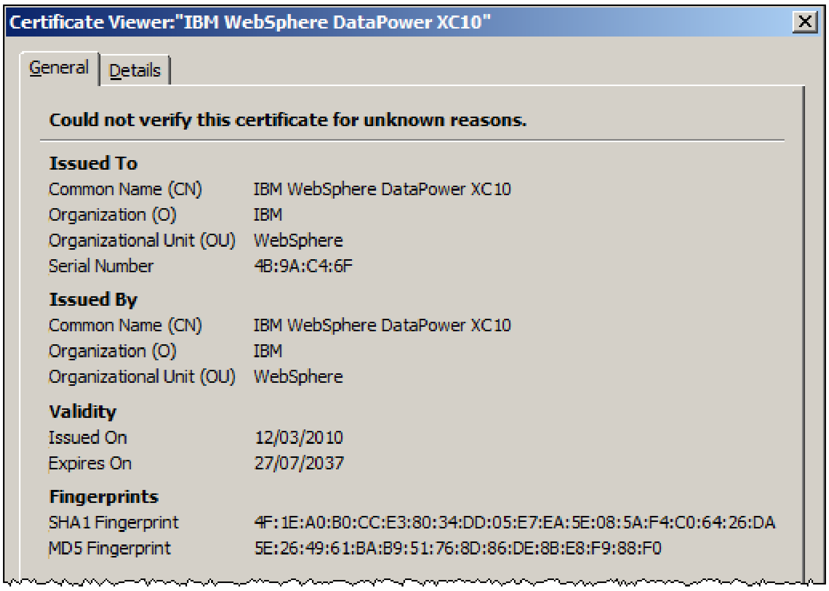
Certificate belongs to a different site, which could indicate an identity theft.

**Unknown Identity**

Certificate is not trusted, because it hasn't been verified by a recognized authority using a secure signature.

□ Permanently store this exception

7. Click the “View” button to make sure the ip address you entered is the XC10 virtual appliance, it will look similar to this:



**Could not verify this certificate for unknown reasons.**

**Issued To**

| | |
|---|---|
| Common Name (CN) | IBM WebSphere DataPower XC10 |
| Organization (O) | IBM |
| Organizational Unit (OU) | WebSphere |
| Serial Number | 4B:9A:C4:6F |

**Issued By**

| | |
|---|---|
| Common Name (CN) | IBM WebSphere DataPower XC10 |
| Organization (O) | IBM |
| Organizational Unit (OU) | WebSphere |

**Validity**

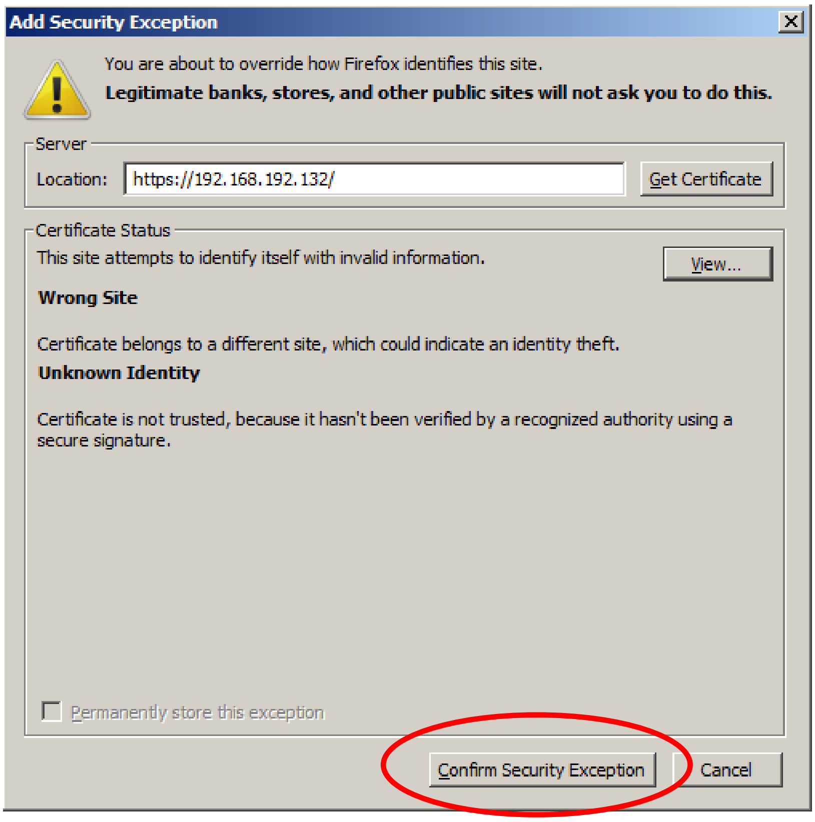
| | |
|---|---|
| Issued On | 12/03/2010 |
| Expires On | 27/07/2037 |

**Fingerprints**

| | |
|---|---|
| SHA1 Fingerprint | 4F:1E:A0:B0:CC:E3:80:34:DD:05:E7:EA:5E:08:5A:F4:C0:64:26:DA |
| MD5 Fingerprint | 5E:26:49:61:BA:B9:51:76:8D:86:DE:8B:E8:F9:88:F0 |

Close the viewer window.

8. Back in the “Add security Exception” window, click “Confirm Security Exception”:



You are about to override how Firefox identifies this site.

**Legitimate banks, stores, and other public sites will not ask you to do this.**

Location: https://192.168.192.132/

This site attempts to identify itself with invalid information.

**Wrong Site**

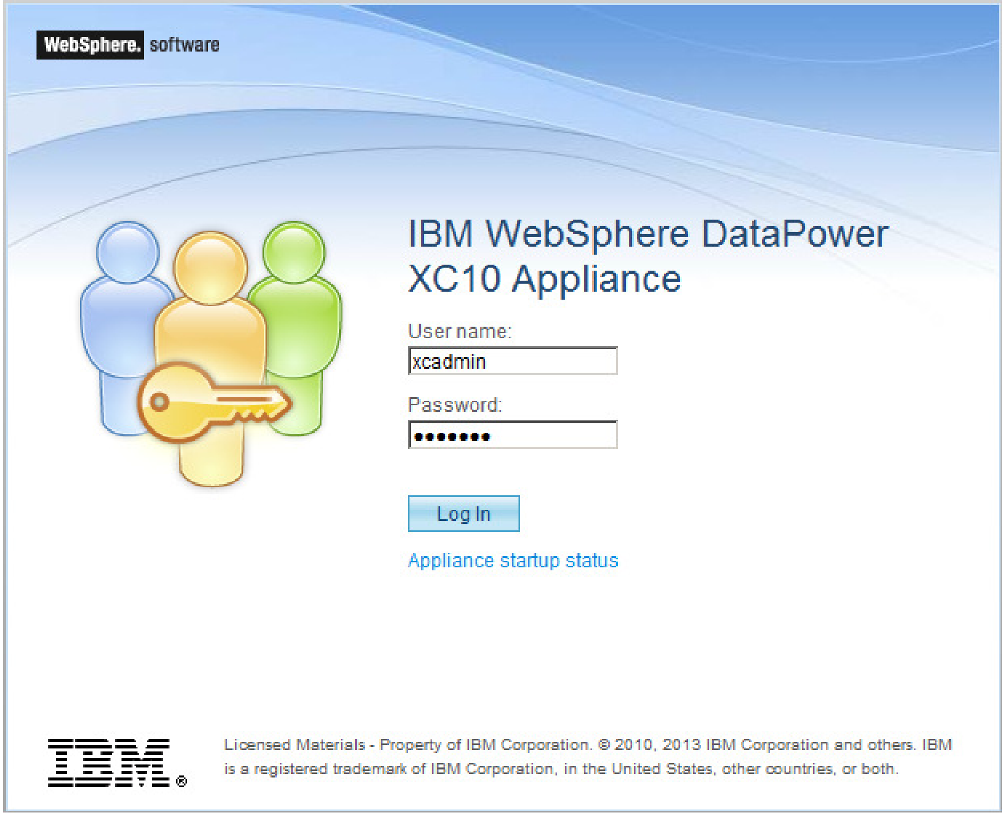
Certificate belongs to a different site, which could indicate an identity theft.

**Unknown Identity**

Certificate is not trusted, because it hasn't been verified by a recognized authority using a secure signature.

□ Permanently store this exception

9. The Web Administration GUI sign-on screen will appear. Log in using “User name” `xcadmin` and password `xcadmin` :

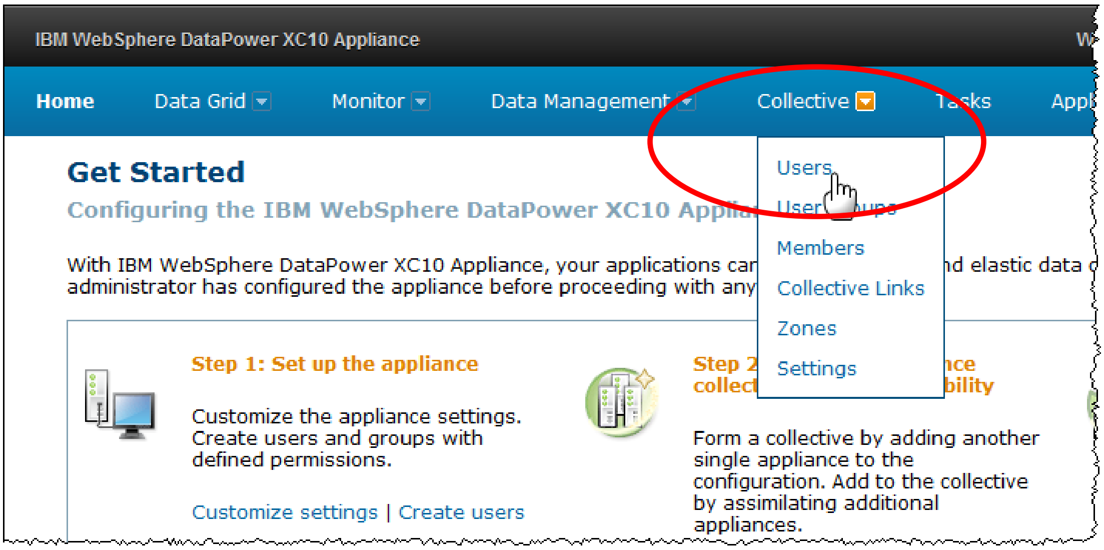


User name: xcadmin

Password: •••••••

Licensed Materials - Property of IBM Corporation. © 2010, 2013 IBM Corporation and others. IBM is a registered trademark of IBM Corporation, in the United States, other countries, or both.

10. On the main XC10 Web Administration GUI window click “Collective” then “Users”



Get Started

Configuring the IBM WebSphere DataPower XC10 Appli

With IBM WebSphere DataPower XC10 Appliance, your applications car administrator has configured the appliance before proceeding with any nd elastic data c

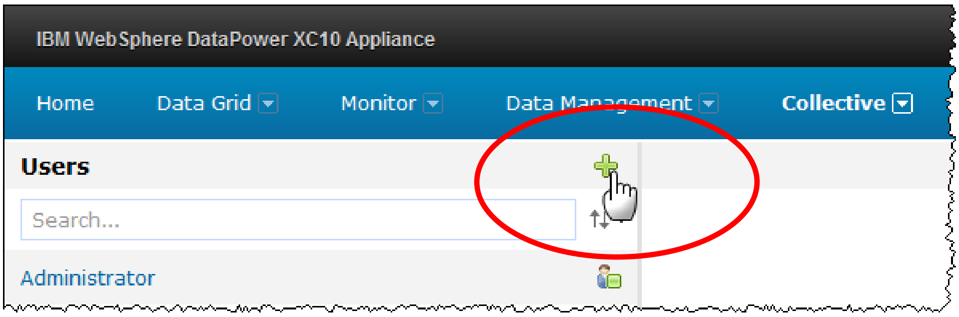
Step 1: Set up the appliance

Customize the appliance settings. Create users and groups with defined permissions.

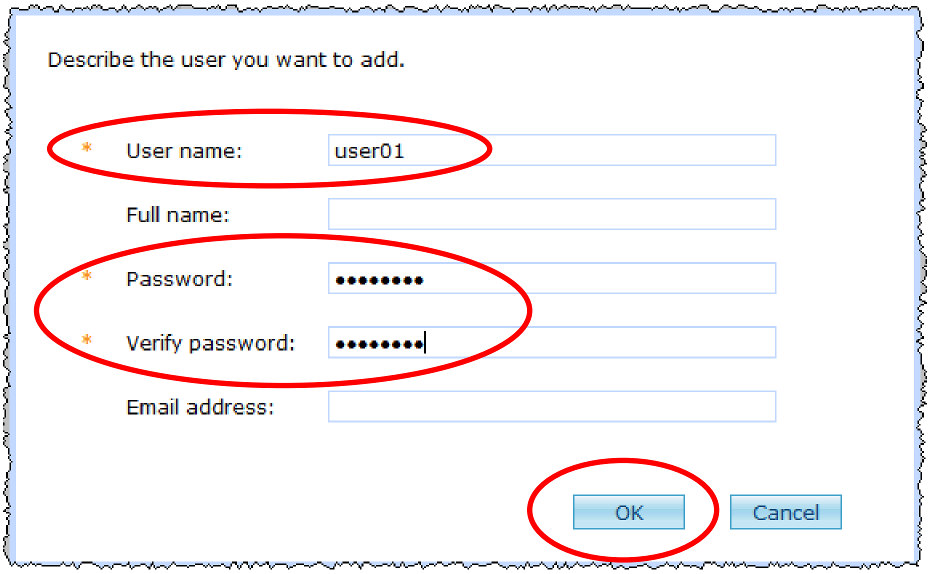
Step 2 collect nce bility

Form a collective by adding another single appliance to the configuration. Add to the collective by assimilating additional appliances.

11. In the “Users” screen click the green plus symbol (to add a user):



12. Type a User name (`user01` in the case below), with a password of “`passw0rd`” (password with a zero instead of the ‘o’), click OK when complete :



Describe the user you want to add.

* User name: user01

Full name:

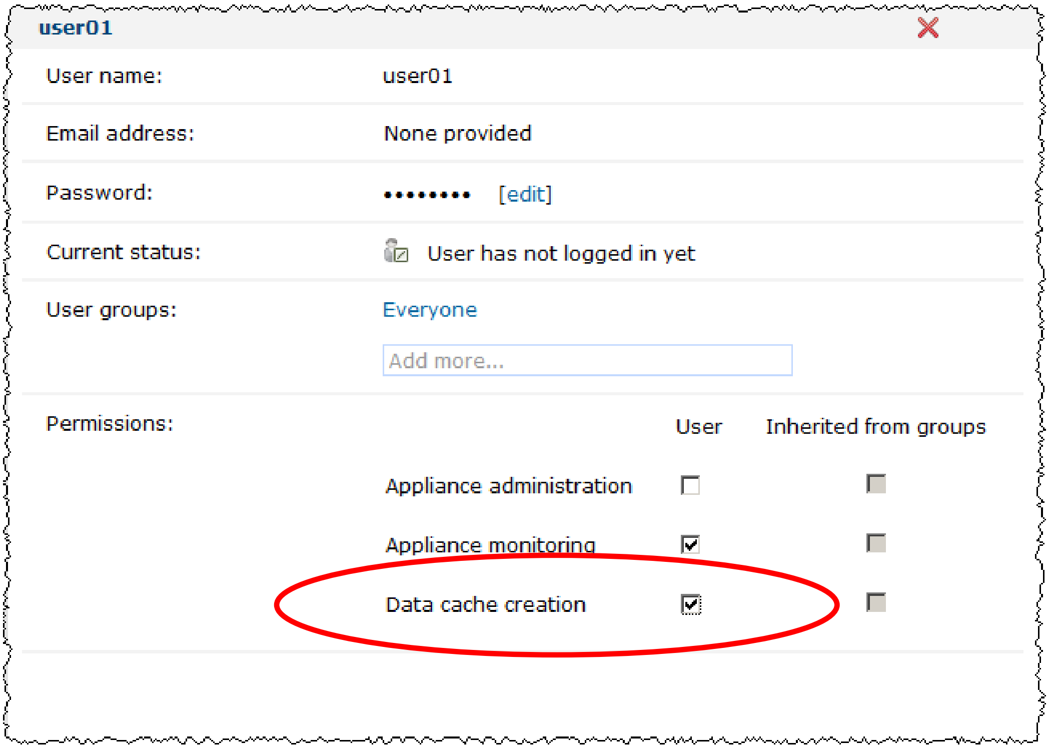
* Password: ••••••••

* Verify password: ••••••••

Email address:

### 3.2.1.3 Define permissions for user

13. When added to the list of users on the virtual appliance, enable the user with “Data cache creation” permission by check marking the box for “Data cache creation” :



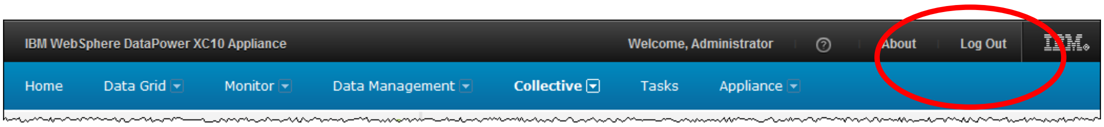
**user01**

| | |
|---|---|
| User name: | user01 |
| Email address: | None provided |
| Password: | •••••••• [edit] |
| Current status: | User has not logged in yet |
| User groups: | Everyone<br>Add more... |

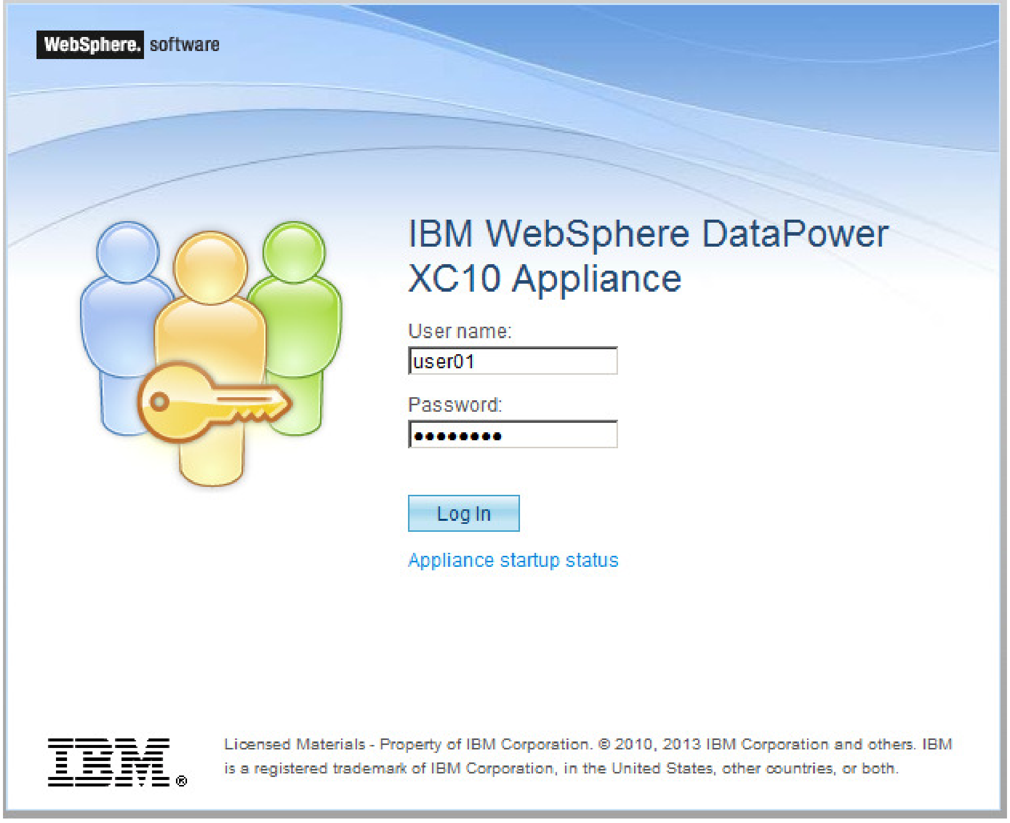
Permissions:

| | User | Inherited from groups |
|---|---|---|
| Appliance administration | □ | □ |
| Appliance monitoring | ☑ | □ |
| Data cache creation | ☑ | □ |

14. Log out of the Web Administration tool by clicking the “Log Out” link :



15. Sign on using the user that you just added:



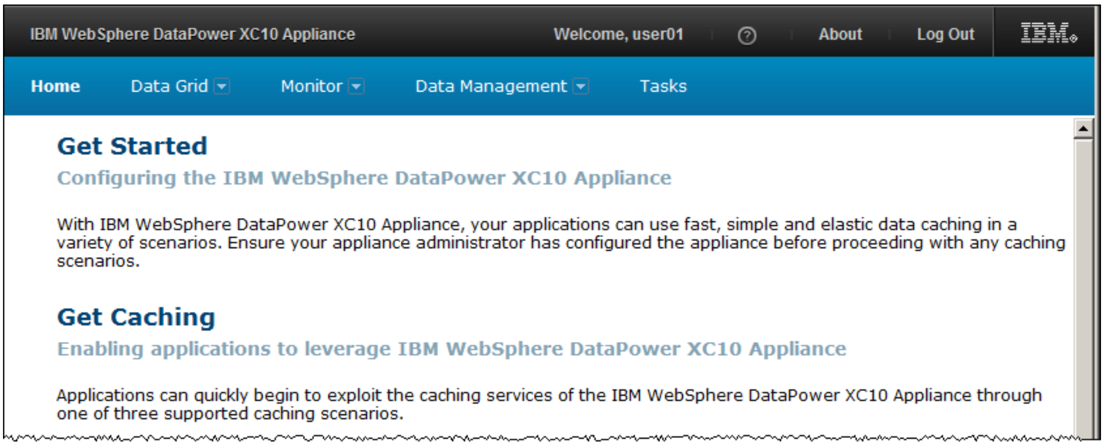
IBM WebSphere DataPower XC10 Appliance

User name: user01

Password: ••••••••

Licensed Materials - Property of IBM Corporation. © 2010, 2013 IBM Corporation and others. IBM is a registered trademark of IBM Corporation, in the United States, other countries, or both.

16. Notice the user has the ability to create and monitor a data grid however can not administer the appliance (ie can not change appliance settings or add users):



**Get Started**

Configuring the IBM WebSphere DataPower XC10 Appliance

With IBM WebSphere DataPower XC10 Appliance, your applications can use fast, simple and elastic data caching in a variety of scenarios. Ensure your appliance administrator has configured the appliance before proceeding with any caching scenarios.

**Get Caching**

Enabling applications to leverage IBM WebSphere DataPower XC10 Appliance

Applications can quickly begin to exploit the caching services of the IBM WebSphere DataPower XC10 Appliance through one of three supported caching scenarios.

17. You are now ready to proceed with the instructions in this Lab Guide. This is the XC10 user that you will use throughout this lab guide, the remainder of this lab guide will refer to this user as userXX.

### 3.2.2 Using a physical DataPower XC10 appliance

**If you are using a Virtual DataPower XC10 appliance you can ignore this section.**

If you are using a physical XC10 device, ask your XC10 administrator for:

1) The IP address of the DataPower XC10 administration interface.

2) An XC10 user called userXX (XX can be a number between 0 and 99 if multiple users are required )

3) Permissions: userXX must have “Data cache creation” and “Appliance monitoring” permissions.

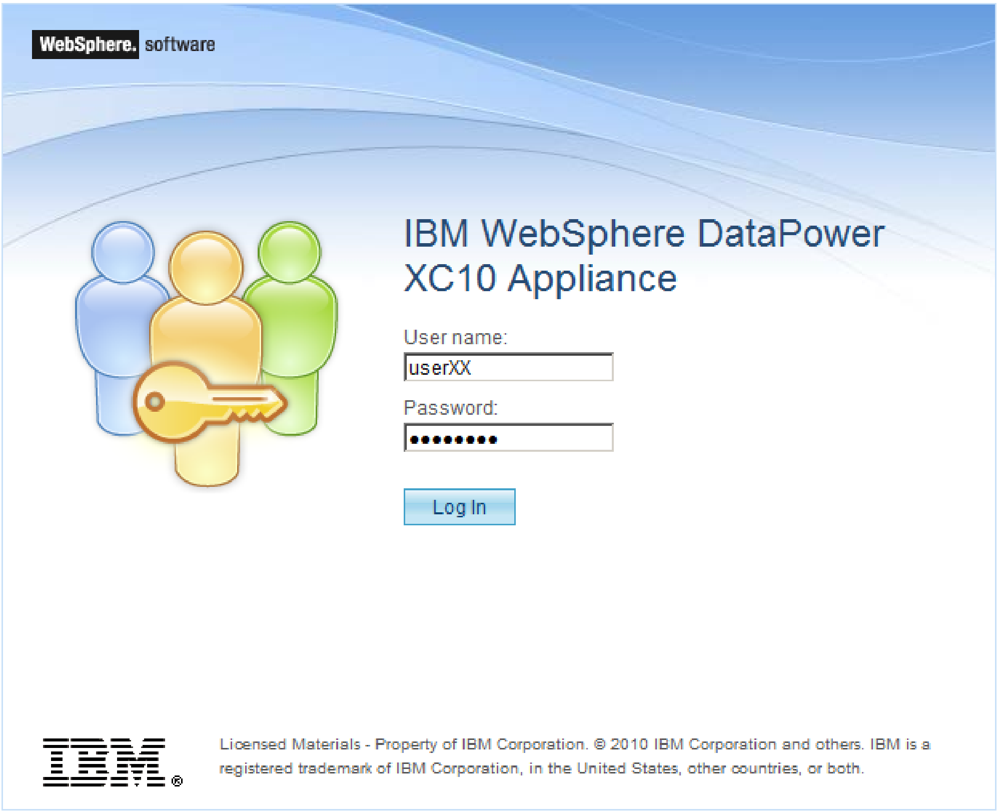
NB: If you need to see how to add a user on a physical appliance the process is exactly the same as documented in the previous section where you add a user to the virtual XC10 appliance (*note passwords on your physical device may be different*).

# 4. Configure your environment

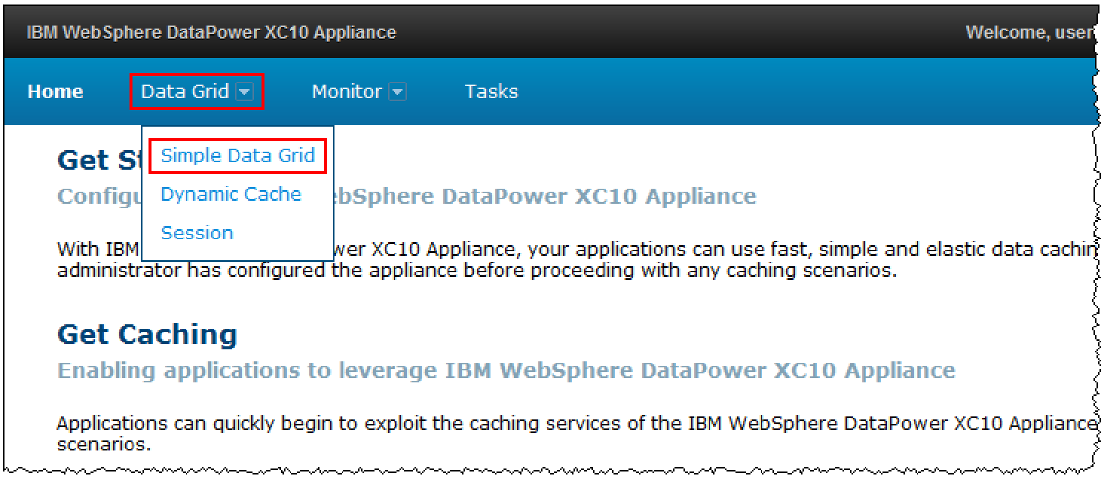
This section and remainder of this lab guide is relevant to both XC10 options (physical and virtual).

## 4.1 Create a data grid on the XC10 appliance

The XC10 appliance provides a web based configuration tool. You will now use this web based tool to create a simple data grid to store information used in this lab guide.

| | |
|---|---|
| 18. | Point a browser at the IP address of your XC10 Appliance.<br>The Configuration tool login screen will appear.<br>Sign on with your XC10 user credentials. (*you will either have been given this user if using a physical appliance – or added a user in the previous section if you are using the supplied virtual appliance - for the purposes of illustration throughout this lab we will use userXX)* :<br><br>User name: userXX<br>Password: ●●●●●●●●●<br>Licensed Materials - Property of IBM Corporation. © 2010 IBM Corporation and others. IBM is a registered trademark of IBM Corporation, in the United States, other countries, or both.  WebSphere. software<br>IBM WebSphere DataPower XC10 Appliance<br>Log In<br>IBM |

19. On the Home page select “Simple Data Grid” from the “Data Grid” drop down:



Get St

Configu bSphere DataPower XC10 Appliance

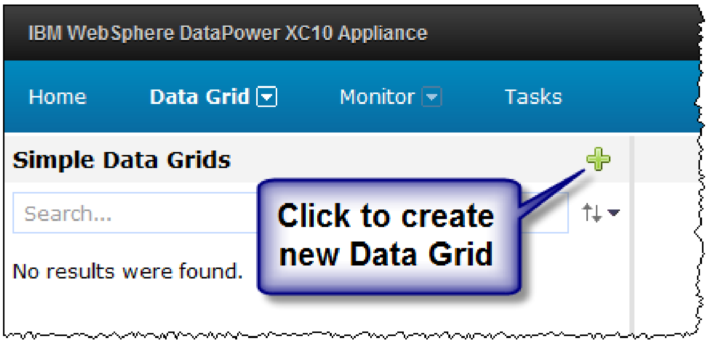
With IBM wer XC10 Appliance, your applications can use fast, simple and elastic data cachin administrator has configured the appliance before proceeding with any caching scenarios.

Get Caching

Enabling applications to leverage IBM WebSphere DataPower XC10 Appliance

Applications can quickly begin to exploit the caching services of the IBM WebSphere DataPower XC10 Appliance scenarios.

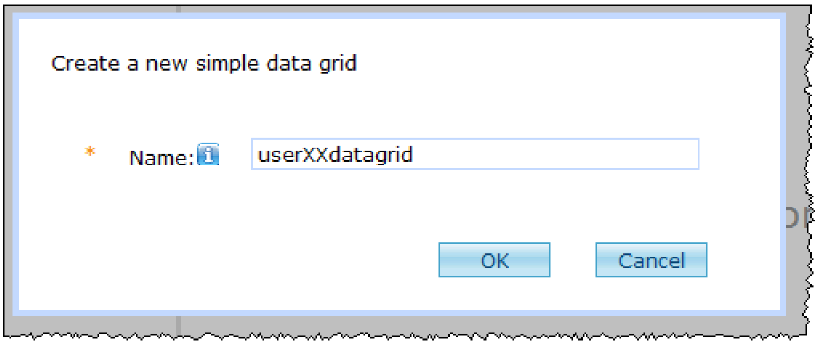
20. On the “Simple Data Grids” screen:



No results were found.

Click to create new Data Grid

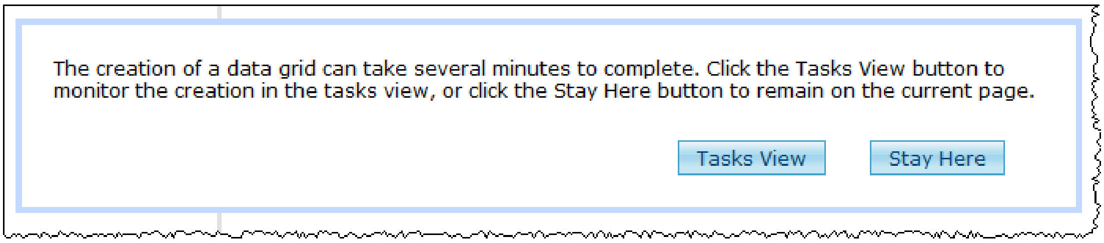
21. Enter a name for the grid, and click OK (*make the grid name specific to the user you are using by replacing XX with the numeric digits applicable to your user for example user01datagrid if you are user01*):



Create a new simple data grid

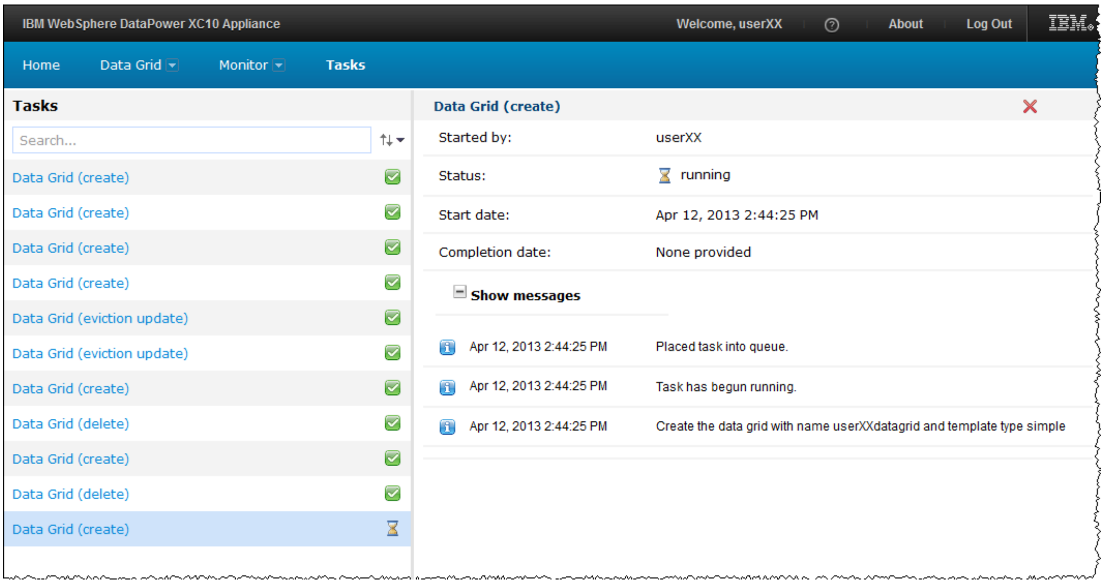
* Name: userXXdatagrid

22. On the next pop up click the “Tasks View”:



The creation of a data grid can take several minutes to complete. Click the Tasks View button to monitor the creation in the tasks view, or click the Stay Here button to remain on the current page.

23. In the “Tasks” view you will see that your request has been scheduled and is “running”. On A light use system a data grid will take 1 or 2 minutes to create on the appliance:



| Tasks | |
|---|---|
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (eviction update) | ✓ |
| Data Grid (eviction update) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (delete) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (delete) | ✓ |
| Data Grid (create) | ⌛ |

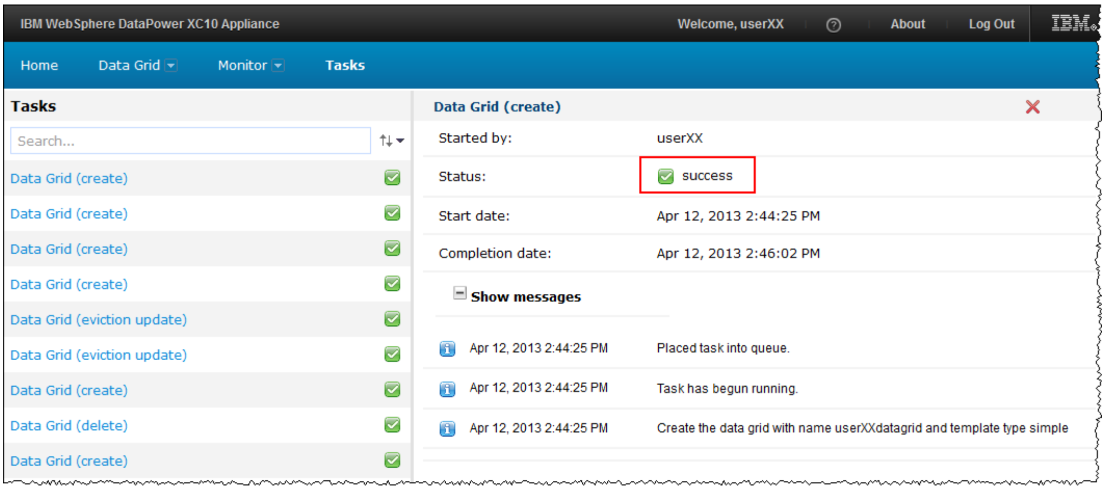
**Data Grid (create)**

| | |
|---|---|
| Started by: | userXX |
| Status: | ⌛ running |
| Start date: | Apr 12, 2013 2:44:25 PM |
| Completion date: | None provided |

**Show messages**

| | |
|---|---|
| Apr 12, 2013 2:44:25 PM | Placed task into queue. |
| Apr 12, 2013 2:44:25 PM | Task has begun running. |
| Apr 12, 2013 2:44:25 PM | Create the data grid with name userXXdatagrid and template type simple |

24. When complete the “Tasks” view will update with the completion status for the request. Ensure your grid is created successfully (status: Success in the task view):



| Tasks | |
|---|---|
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (eviction update) | ✓ |
| Data Grid (eviction update) | ✓ |
| Data Grid (create) | ✓ |
| Data Grid (delete) | ✓ |
| Data Grid (create) | ✓ |

**Data Grid (create)**

| | |
|---|---|
| Started by: | userXX |
| Status: | ✓ success |
| Start date: | Apr 12, 2013 2:44:25 PM |
| Completion date: | Apr 12, 2013 2:46:02 PM |

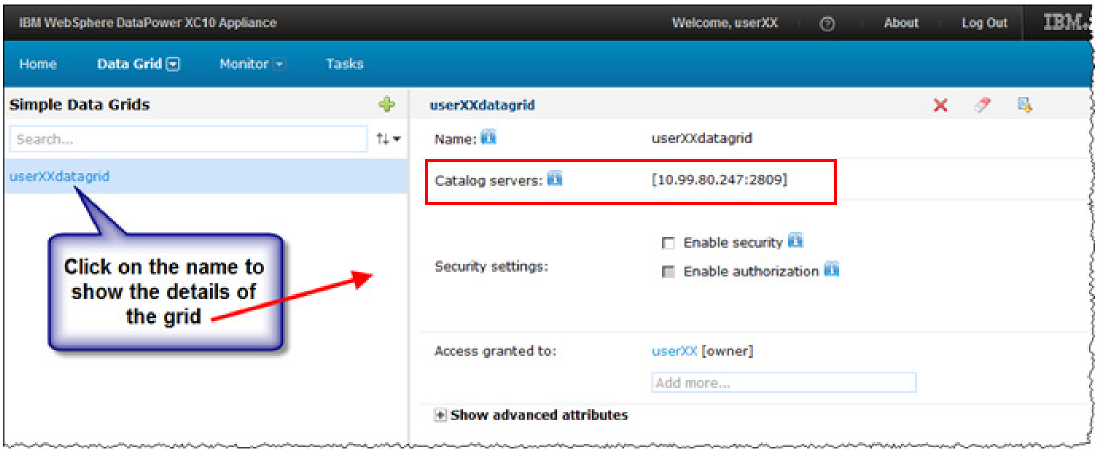
**Show messages**

| | |
|---|---|
| Apr 12, 2013 2:44:25 PM | Placed task into queue. |
| Apr 12, 2013 2:44:25 PM | Task has begun running. |
| Apr 12, 2013 2:44:25 PM | Create the data grid with name userXXdatagrid and template type simple |

25. Navigate back to the Simple Data Grid menu and click on the link to the grid you have just created.

Make a note of the Catalog Server (ip address and port):

< ________________________________________ >



Name: userXXdatagrid

Catalog servers: [10.99.80.247:2809]

Security settings: □ Enable security □ Enable authorization

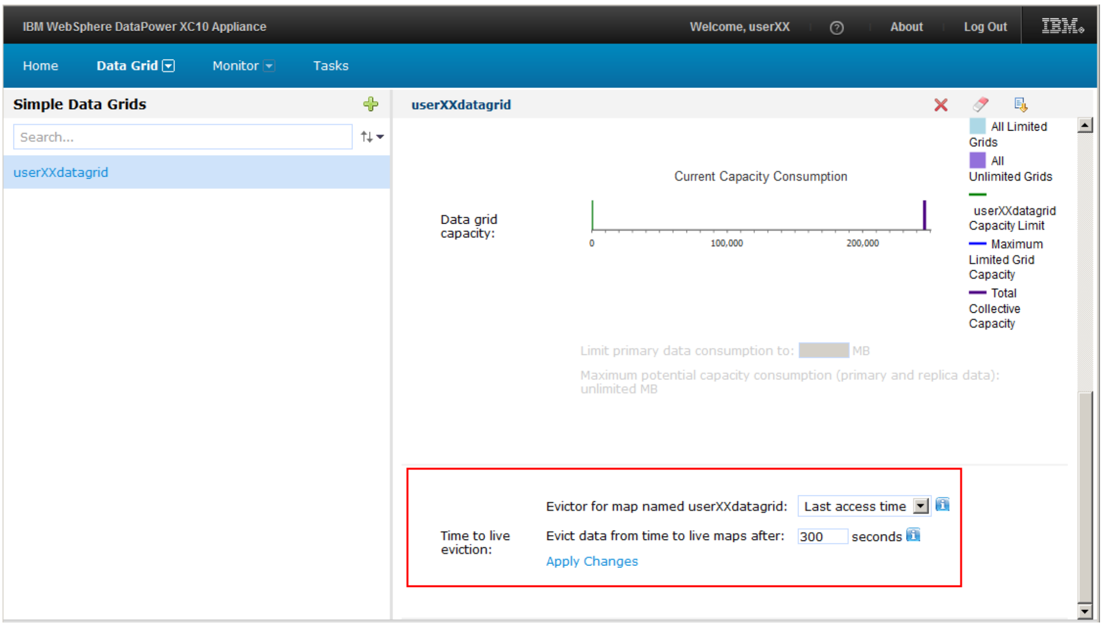
Access granted to: userXX [owner]

Add more...

Click on the name to show the details of the grid

26. Click “**Show advanced attributes**”, scroll to the bottom of the advanced attributes for the data grid you created. In the “Time to live eviction” section, set

1) the “Evictor” for the map to “Last access time” and
2) “Evict data from time to live maps after” to “300” seconds:



Limit primary data consumption to: MB

Maximum potential capacity consumption (primary and replica data): unlimited MB

Time to live eviction:

Evictor for map named userXXdatagrid: Last access time

Evict data from time to live maps after: 300 seconds

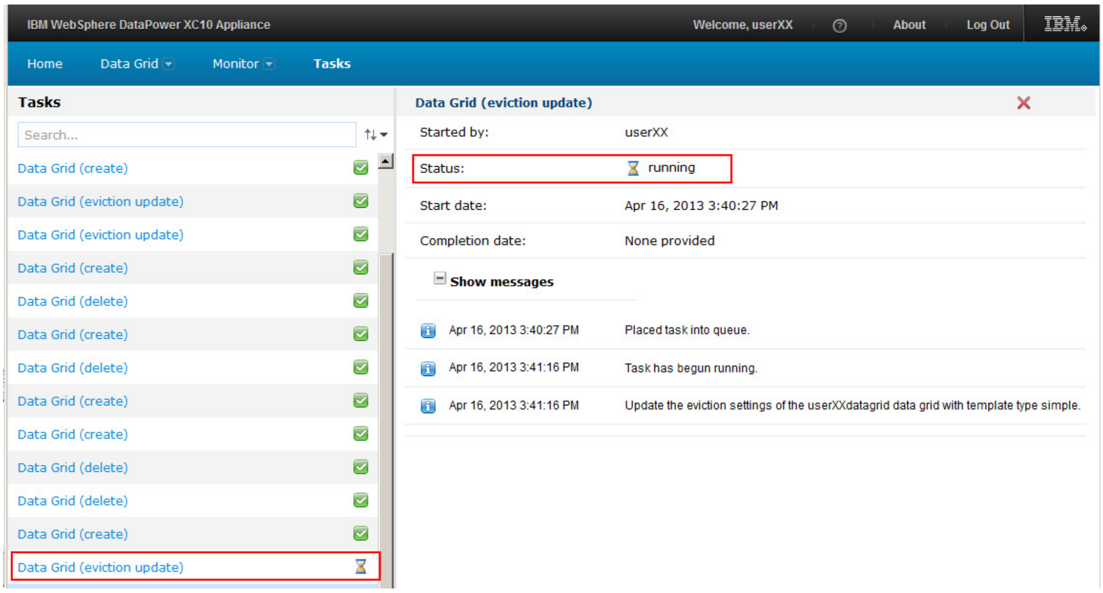
Apply Changes

When an XC10 data grid is created the default eviction policy is set so that data is kept until the grid is restarted.

Changing the Time to Live eviction at this level means that data not accessed for 5 minutes (300 seconds) or longer will be automatically removed from data grid.

Click “Apply changes” and then OK to confirm the changes (and then “Tasks View”). Note that applying this action will restart the data grid (any existing data in the data grid is removed on a restart).

27. In the “Task View” you will see the “eviction update” with a status (queued or running). The screen will automatically update when the task has completed (the status will change to “success”):



Tasks

- Data Grid (create) ✓
- Data Grid (eviction update) ✓
- Data Grid (eviction update) ✓
- Data Grid (create) ✓
- Data Grid (delete) ✓
- Data Grid (create) ✓
- Data Grid (delete) ✓
- Data Grid (create) ✓
- Data Grid (create) ✓
- Data Grid (delete) ✓
- Data Grid (delete) ✓
- Data Grid (create) ✓
- Data Grid (eviction update) ⌛

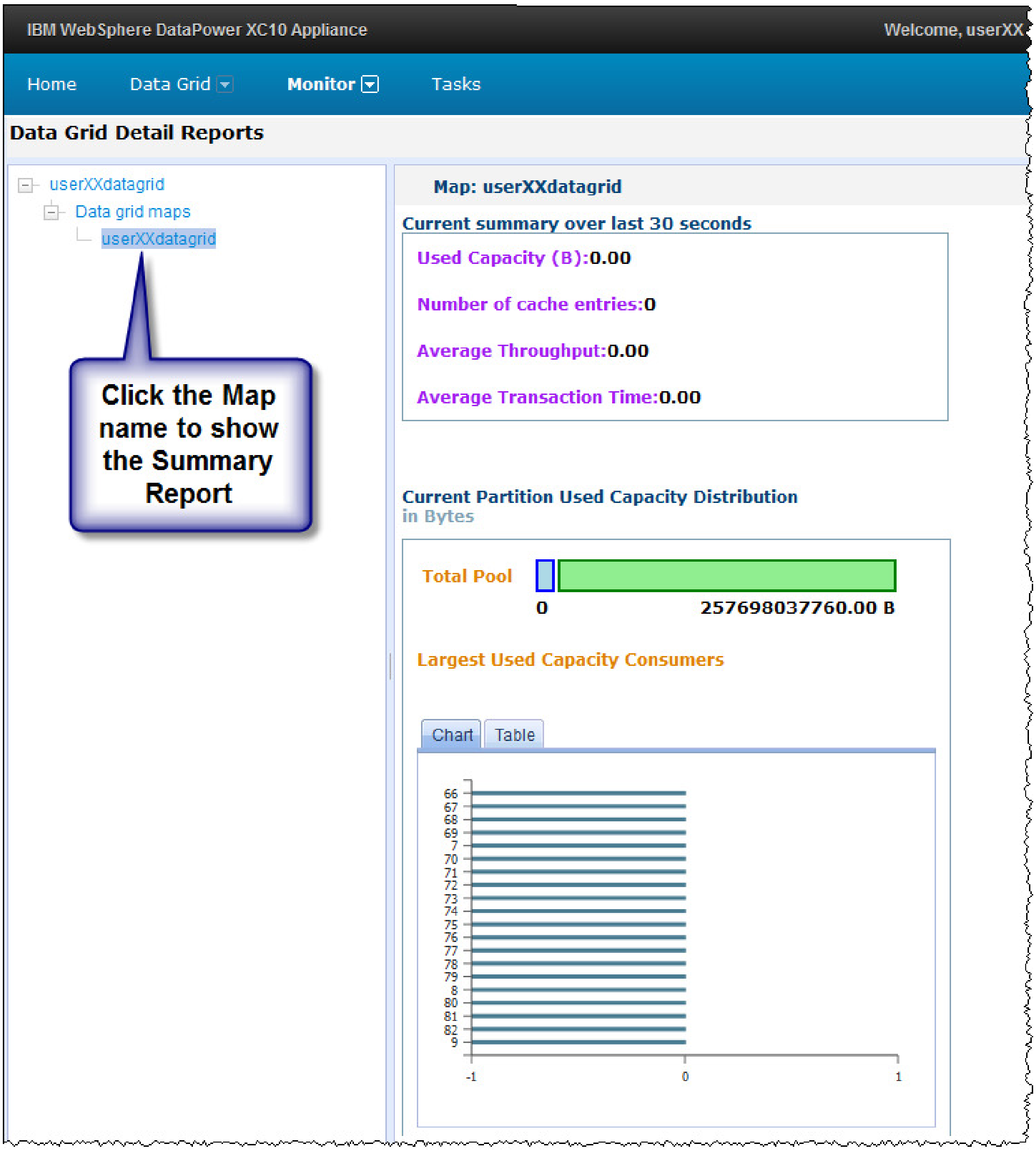
Data Grid (eviction update)

| | |
|---|---|
| Started by: | userXX |
| Status: | ⌛ running |
| Start date: | Apr 16, 2013 3:40:27 PM |
| Completion date: | None provided |

⊟ Show messages

| | |
|---|---|
| Apr 16, 2013 3:40:27 PM | Placed task into queue. |
| Apr 16, 2013 3:41:16 PM | Task has begun running. |
| Apr 16, 2013 3:41:16 PM | Update the eviction settings of the userXXdatagrid data grid with template type simple. |

28. The XC10 configuration tool also provides summary statistics on data grid usage. Click on “Monitor” and choose “Data Grid Detail Reports”. Note that there will have been no usage on your grid at this stage:



Data Grid Detail Reports

Click the Map name to show the Summary Report

Map: userXXdatagrid

Current summary over last 30 seconds

Used Capacity (B):0.00

Number of cache entries:0

Average Throughput:0.00

Average Transaction Time:0.00

Current Partition Used Capacity Distribution in Bytes

Total Pool 0 257698037760.00 B

Largest Used Capacity Consumers

29. Your XC10 data grid is now ready to be used. The next configuration task is to make IIB aware of the data grid.

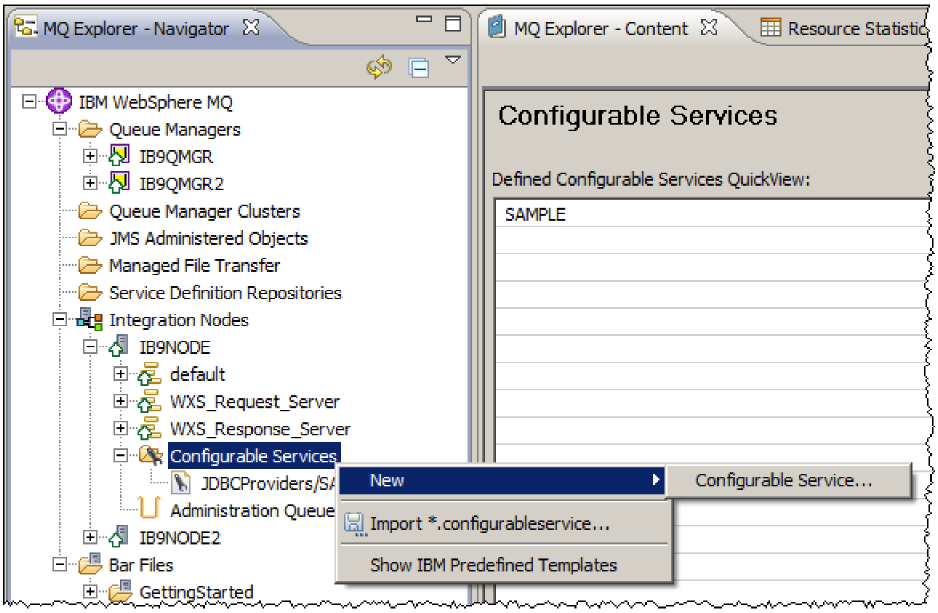
Move to the next section “Create WXSServer configurable service”.

## 4.2 Create WXSServer configurable service

Details of the data grid and connectivity details for the XC10 server are stored in an Integration Bus configurable service of type "WXSServer". Note this configurable service defines the grid that you will be using (not just the connection specific details of the XC10 appliance. Since multiple data grids can exist on an XC10 appliance more than one definition of type "WXSServer" can exist in an environment (with the same XC10 connection details).

For the purposes of this lab guide, this configuration will be done using the Integration Explorer plug-in (in WebSphere MQ Explorer), the mqsicreateconfigurableservice command can be used as an alternative.

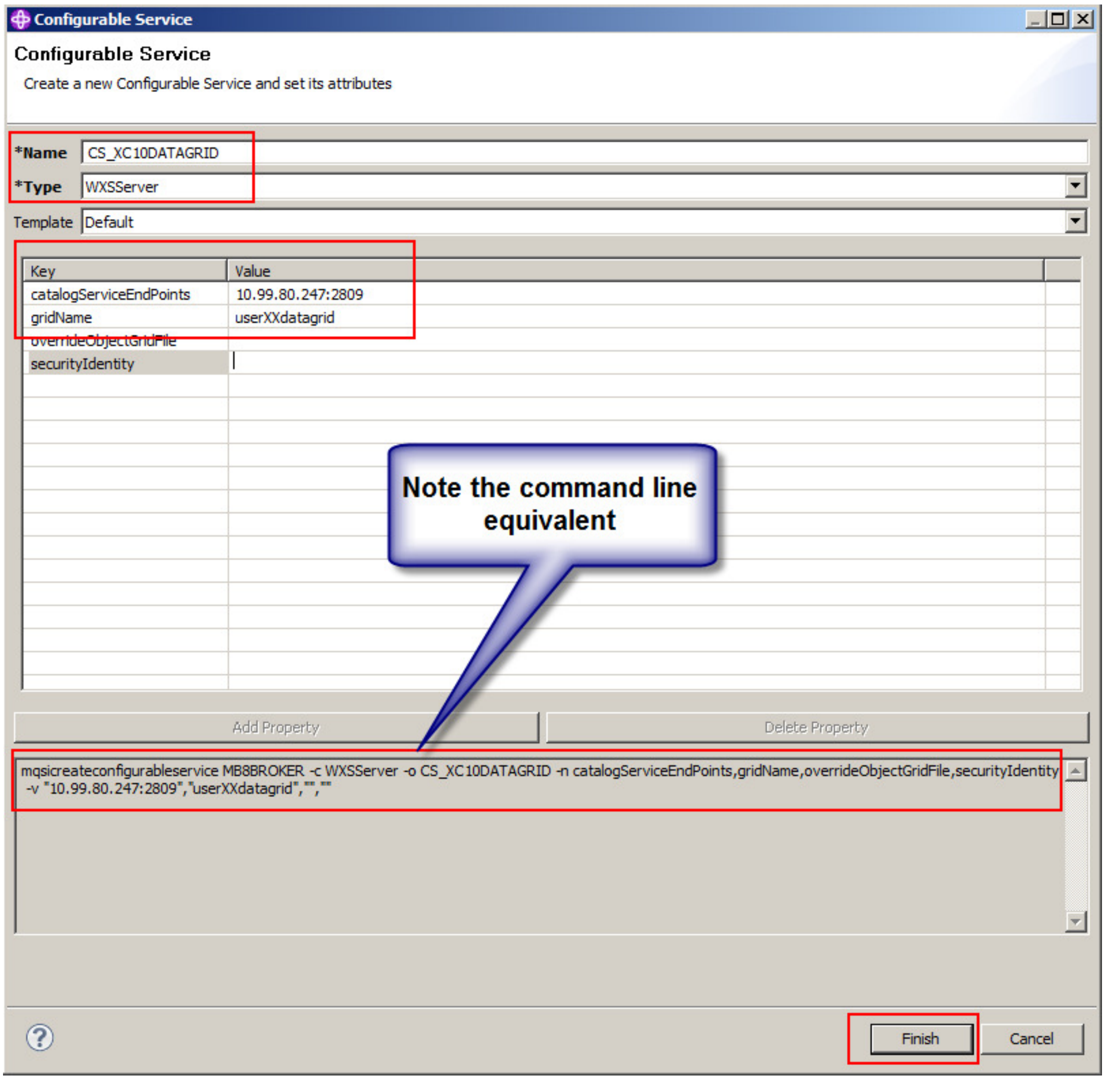
You will now use the Integration Explorer to define a configurable service that the Integration Node can use to connect to the XC10 data grid:

| | |
|---|---|
| 30. | In Integration Explorer, expand IB9NODE and Right click on "Configurable Services", choose "New" then "Configurable Service…" from the options:<br><br> MQ Explorer - Navigator<br>IBM WebSphere MQ<br>Queue Managers<br>IB9QMGR<br>IB9QMGR2<br>Queue Manager Clusters<br>JMS Administered Objects<br>Managed File Transfer<br>Service Definition Repositories<br>Integration Nodes<br>IB9NODE<br>default<br>WXS_Request_Server<br>WXS_Response_Server<br>Configurable Services<br>JDBCProviders/S/<br>Administration Queue<br>IB9NODE2<br>Bar Files<br>GettingStarted<br>MQ Explorer - Content<br>Resource Statistic<br>Configurable Services<br>Defined Configurable Services QuickView:<br>SAMPLE<br>New<br>Configurable Service…<br>Import *.configurableservice…<br>Show IBM Predefined Templates |

31. In the Wizard that opens change the "Type" to "`WXSServer`" and enter:

1. the IP and port details of the XC10 catalog server (you made a note of it earlier)
2. the name of the data grid ("`userXXdatagrid`") (replace userXX with your details)
3. Name the service "`CS_XC10DATAGRID`", (*case is important since we will be using this value in the Java compute nodes later in the lab*)

Click Finish to save:



Configurable Service

Create a new Configurable Service and set its attributes

*Name CS_XC10DATAGRID

*Type WXSServer

Template Default

| Key | Value |
|---|---|
| catalogServiceEndPoints | 10.99.80.247:2809 |
| gridName | userXXdatagrid |
| overrideObjectGridFile | |
| securityIdentity | |

Note the command line equivalent

```
mqsicreateconfigurableservice MB8BROKER -c WXSServer -o CS_XC10DATAGRID -n catalogServiceEndPoints,gridName,overrideObjectGridFile,securityIdentity
 -v "10.99.80.247:2809","userXXdatagrid","",""
```

The Configurable service CS_XC10DATAGRID will appear in the list of configurable services for the IB9NODE integration node:

## 4.3 Configure ports and listener for the Integration Node

Integration Nodes use the embedded cache to connect to the XC10 DataPower appliance. The embedded cache requires configuring even if you decide not to use the embedded Global Cache.

The default port range used by the default Integration Node for the embedded eXtreme Scale component is 2800-2819. In the event that this port range conflicts with any other applications, you can change this port range using the Integration Explorer.

In the case of the pre-built system which includes WAS and Information Server, this is necessary to avoid a port conflict.

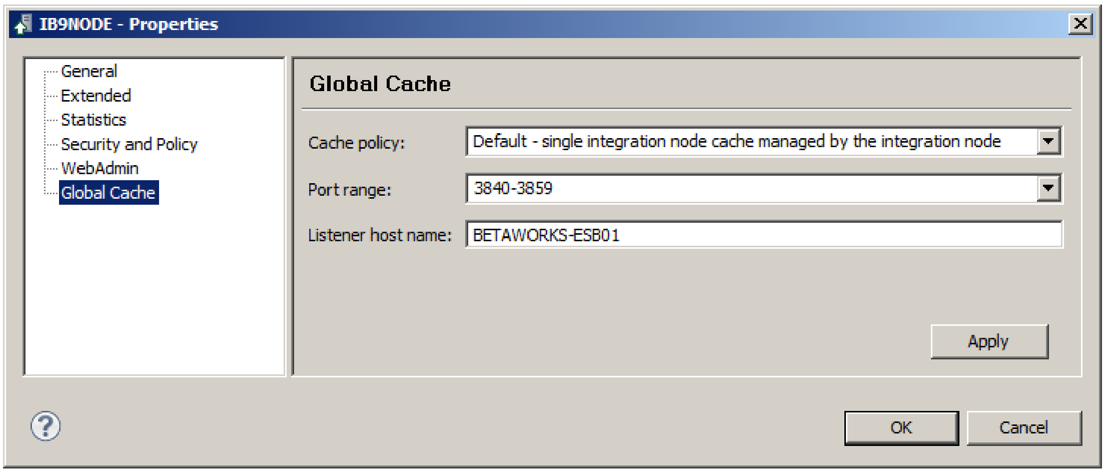
1. In Integration Explorer, right click on IB9NODE, and select Properties. Select Global Cache.

2. Set Cache policy to “`Default - single broker cache managed by the broker`”.

   Change the port range to `3840-3859`.

   Set the Listener host name to “`BETAWORKS-ESB01`” (the  host name of your VM machine – check this value by entering “hostname” in a command window)

   Click **Apply**, then OK:



IB9NODE - Properties

Global Cache

Cache policy: Default - single integration node cache managed by the integration node

Port range: 3840-3859

Listener host name: BETAWORKS-ESB01

3. Stop and restart the Integration Node.

# 5. Import the applications

The application that you will use to investigate IBM Integration Bus support of the DataPower XC10 Appliance is provided for you. In this section, you will import the application, and investigate certain aspects of the flow logic. The application that is used to demonstrate IBM Integration Bus use of the XC10 DataPower appliance for caching is very similar to the application used with the embedded global cache. The following sections will highlight the differences when using the XC10 appliance.

4. In the Integration Toolkit, import the Project Interchange file at the following location:

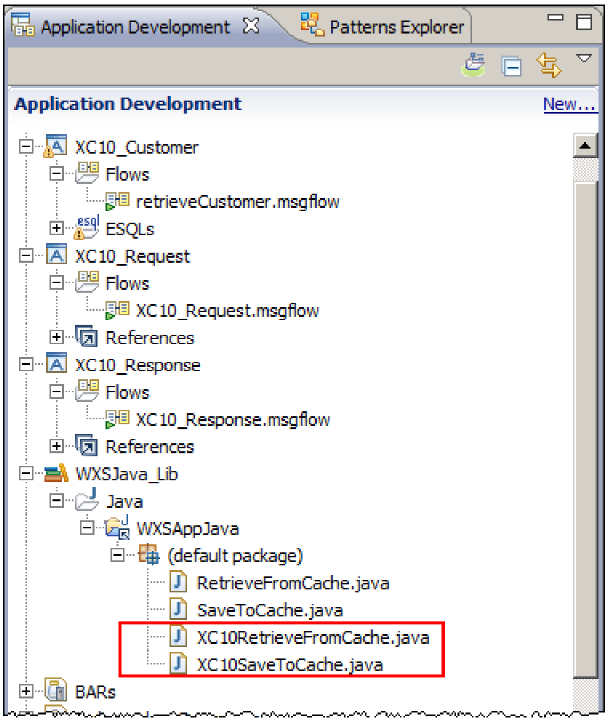
```
c:\student\global_cache\resources\ XC10_Start.zip
```

5. Explore the items have been defined in the Application Development navigator.

Things to note:
There are three applications, each containing one message flow. The application has been organized in this way in order that each application (and therefore message flow) can be deployed independently to separate Integration Servers, or to separate Integration Nodes.

The `XC10_Request` and `XC10_Response` applications have a reference to the WXSJava_Lib library. The WXSJava_Lib library contains four java compute nodes, two of these are relevant to the XC10 cache scenario: One for writing information to the XC10 cache, and one for reading from it.

The `XC10_Customer` application has no reference. The `retrieveCustomer` message flow is independent, and uses a simple Compute node to read a DB2 database.

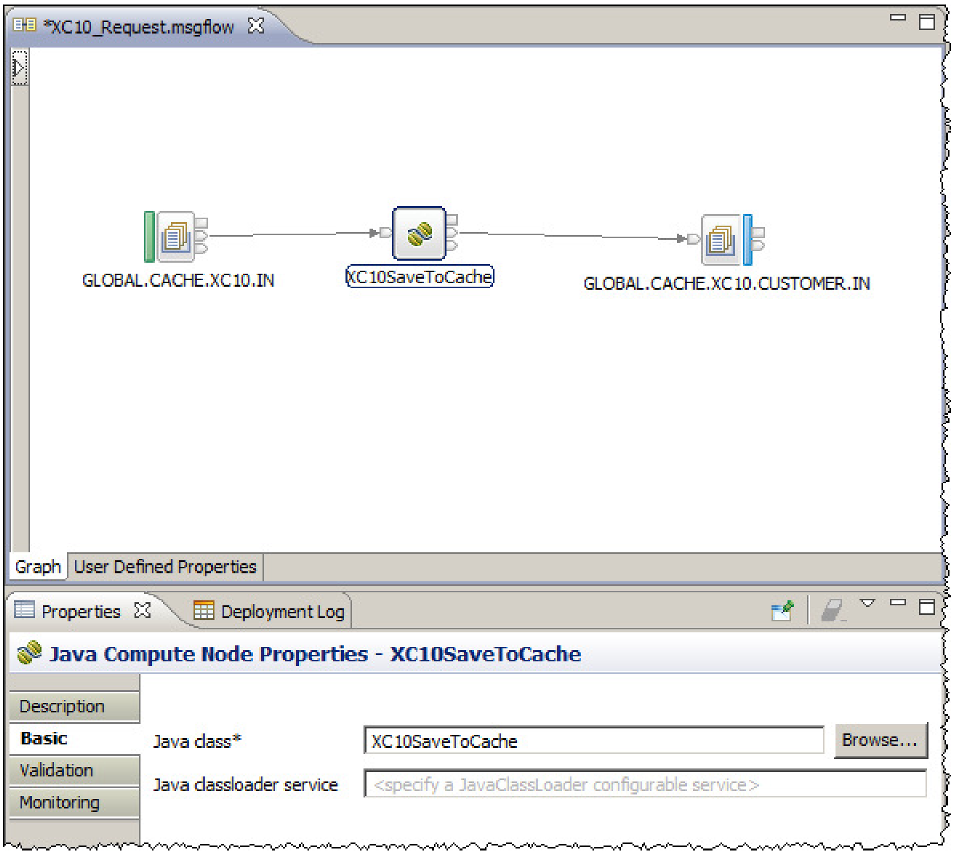


## 5.1 Review the XC10_Request application

6. Open the **XC10_Request** message flow:

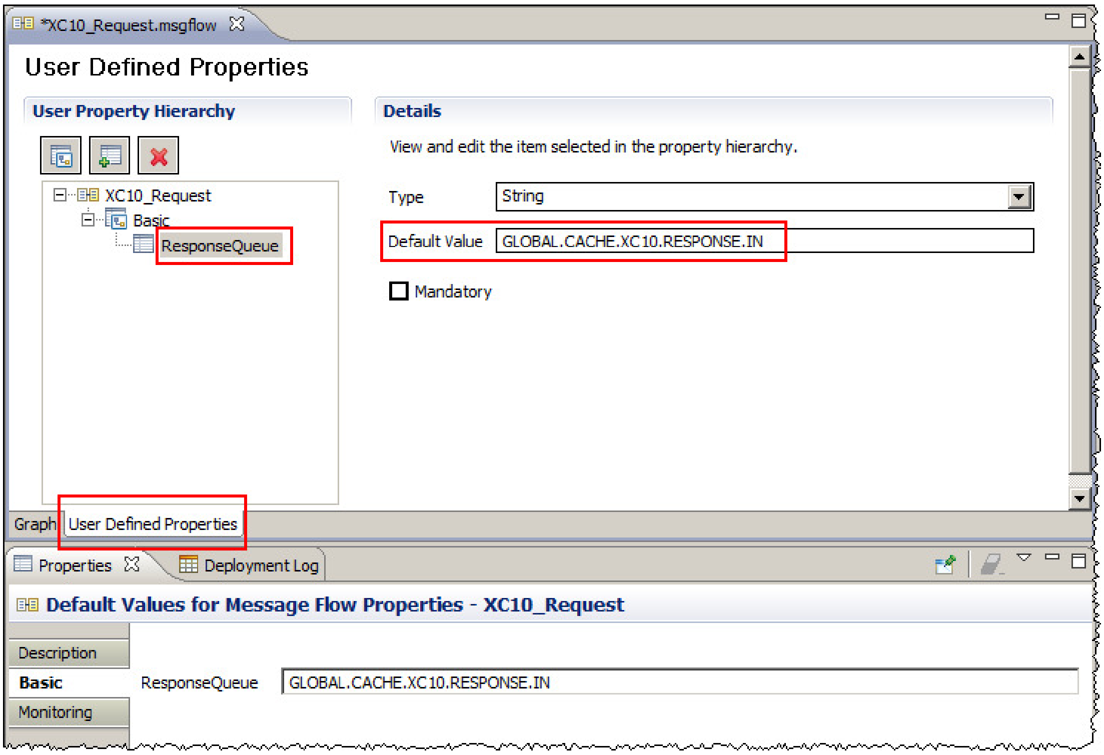
This message flow reads an MQ message from queue `GLOBAL.CACHE.XC10.IN`, and uses a java compute node to store the MQ MessageID and Reply Queue Name (required to correlate the final MQ response message) into the cache.

The message is then sent to the `XC10_retrieveCustomer` message flow by writing it to the `GLOBAL.CACHE.XC10.CUSTOMER.IN` queue.



Java class* XC10SaveToCache

Java classloader service <specify a JavaClassLoader configurable service>

| | |
|---|---|
| 7. | In the flow editor, click "`User defined properties`".<br>You will see that this flow has a user-defined property called `ResponseQueue`. This UDP is used to determine whether the output from the retrieveCustomer flow is sent to a local queue, or to a remote queue (for the purpose of demonstrating caching across multiple brokers). The default value is `GLOBAL.CACHE.XC10.RESPONSE.IN`, which is a local MQ queue.<br>When finished, switch back to the Graph view.<br> |

*XC10_Request.msgflow

User Defined Properties

User Property Hierarchy

- XC10_Request
  - Basic
    - ResponseQueue

Details

View and edit the item selected in the property hierarchy.

Type: String

Default Value: GLOBAL.CACHE.XC10.RESPONSE.IN

□ Mandatory

Default Values for Message Flow Properties - XC10_Request

ResponseQueue: GLOBAL.CACHE.XC10.RESPONSE.IN

### 5.1.1 Modify XC10SaveToCache.java source

8. Double-click on the `XC10SaveToCache.java` source file in WXSJava_Lib to see how the Java logic writes information to the cache. There are three key sections of java code:

This section of code retrieves the MQ MsgId and ReplyToQ from the MQMD (same code is in the default global cache example):

```
// Get the original MsgId and ReplyToQ of the incoming request

MbElement rootEl = outAssembly.getMessage().getRootElement();
MbElement replyToQEl =
rootEl.getFirstElementByPath("/MQMD/ReplyToQ");
String  replyToQ = replyToQEl.getValueAsString();

MbElement msgIdEl= rootEl.getFirstElementByPath("/MQMD/MsgId");
String msgId = msgIdEl.getValueAsString();
```

9. This section of code (unique to using the XC10 appliance) saves this information in the global cache:

```
/*
 * We're about to overwrite the original ReplyToQ of this message.
 * So, write it to the XC10 cache for safe keeping. This will be
 * used subsequently by the XC10 Response message flow to send the
 * response back to the original reply queue.
 */

MbGlobalMap xc10Map=
 MbGlobalMap.getGlobalMap("userXXdatagrid",“CS_XC10DATAGRID”);

xc10Map.put(msgId, replyToQ);
```

**Action**: Overtype the value `"userXXdatagrid"` to reflect the details of the data grid you created on the XC10.

10. ------------------------------------------------------------------------------------------------------------

**A Note on data grids and Maps**: an XC10 data grid can contain multiple “Maps”. When you create a data grid, a single map (with the same name as the grid name – in our case `"userXXdatagrid")` is created by default. It is this map name ( known as a GlobalMap object) that getGlobalMap requires.

It is possible to dynamically create additional maps in a data grid by the WSX client attempting to access a specifically named map in this field. For example a map name of `"userXXdatagrid.LAT"` specified in the code would:

a) dynamically create a map within the data grid if it didn’t exist
b) since the name consists of “<anyname>.LAT” a default data eviction policy (this is when data will be removed from the map) would be based on “Last Access Time” of the entry
c) a default time to live (TTL) of 1 hour. (this would mean data not accessed for 1 hour would be automatically deleted from the map.

More information on XC10 dynamic map configuration options can be found at the following URL: http://publib.boulder.ibm.com/infocenter/wdpxc/v2r0/topic/com.ibm.websphere.datapower.xc.doc/rdynmap.html

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

11. Finally, this section of code reads the value of a user-defined property “ResponseQueue”. The value of this user-defined property determines whether the reply queue of the retrieveCustomer flow will be a local or remote queue.

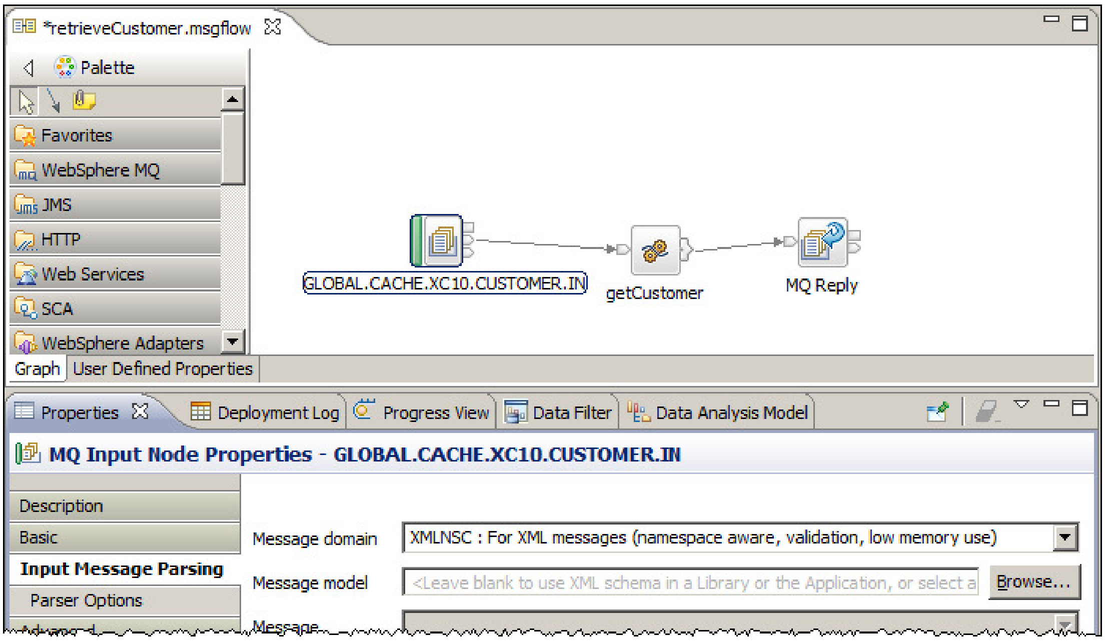
```
responseQueue = getUserDefinedAttribute("ResponseQueue").toString();

outMessage.getRootElement().getFirstElementByPath("/MQMD/ReplyToQ").
setValue(responseQueue + instance);
```

12. Save the java code and exit the java editor when finished, and close the Request message flow.

## 5.2 Review the XC10_Customer application

13. Now open the retrieveCustomer message flow.

On the MQ Input node, the Message Parsing domain has been set to XMLNSC. This is because the flow needs access to the input data (CustomerID) to retrieve a record from the CUSTOMER table.



MQ Input Node Properties - GLOBAL.CACHE.XC10.CUSTOMER.IN

Message domain: XMLNSC : For XML messages (namespace aware, validation, low memory use)

Message model: <Leave blank to use XML schema in a Library or the Application, or select a

Message

14. Open the getCustomer Compute node. The ESQL code will retrieve a row from the CUSTOMER table, putting it temporarily in Environment.Variables.

The FirstName and LastName fields are then stored in the output message.

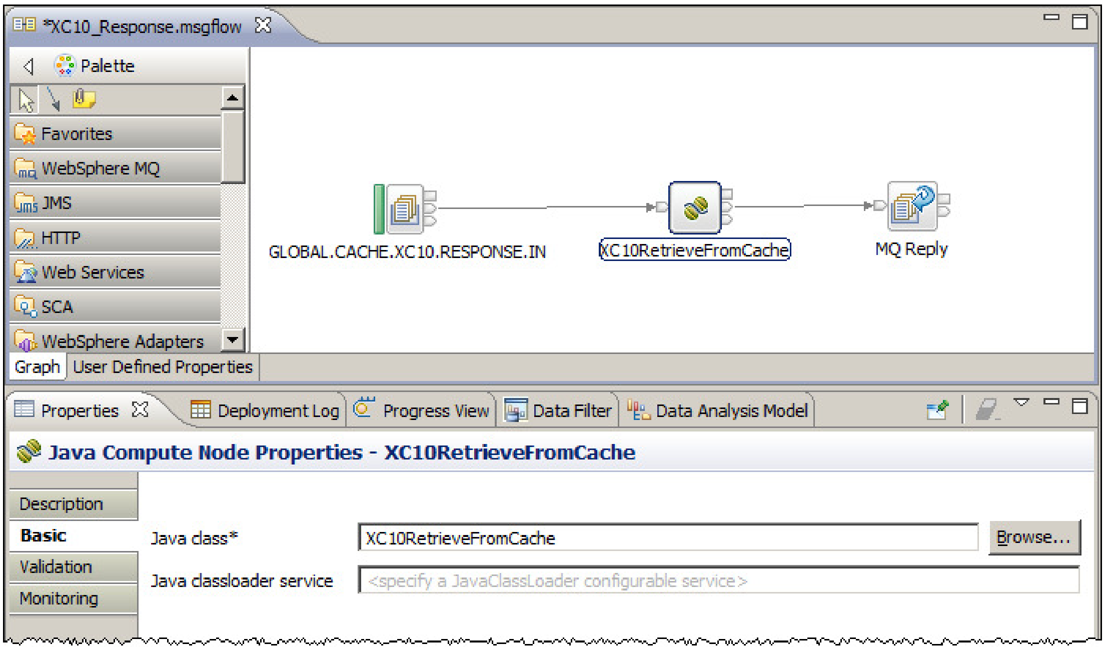
```
BEGIN
-- CALL CopyMessageHeaders();
SET OutputRoot = InputRoot;
SET OutputRoot.XMLNSC.CUSTOMER = NULL;
-- populate the environment with passenger info from the database
SET Environment.Variables =
      THE (SELECT T.* FROM Database.CUSTOMER AS T
      WHERE T.CUSTOMERID = InputRoot.XMLNSC.CUSTOMER.CUSTOMERID);

-- populate the output message with info from the database query
CREATE FIELD OutputRoot.XMLNSC.CUSTOMER;
DECLARE outpass REFERENCE TO OutputRoot.XMLNSC.CUSTOMER;
SET outpass.FirstName = Environment.Variables.FIRSTNAME;
SET outpass.LastName = Environment.Variables.LASTNAME;
RETURN TRUE;
END;
```

Close the ESQL editor.

## 5.3 Review the XC10_Response application

15. Now open the Response message flow.

First, note that the MQ input node reads the queue `GLOBAL.CACHE.XC10.RESPONSE.IN`.



Java Compute Node Properties - XC10RetrieveFromCache

Java class* XC10RetrieveFromCache

Java classloader service <specify a JavaClassLoader configurable service>

16. Open the **XC10RetrieveFromCache** java node.

The following are the important parts of the java code.

This section retrieves the MessageID from the MQMD.

```
      String msgId = inAssembly
            .getMessage()
            .getRootElement()
            .getFirstElementByPath("/MQMD/CorrelId")
            .getValueAsString();
```

17. The following code uses the MessageID to retrieve the name of the reply queue from the Global Cache.

**Action**: Overtype the data grid value "***userXXdatagrid***" to reflect the details of the data grid you created on the XC10.

```
// Now we can restore the original ReplyToQ by looking it up in the
cache.
MbGlobalMap xc10Map =
      MbGlobalMap.getGlobalMap ("UserXXdatagrid","CS_XC10DATAGRID");
String replyToQ = (String)globalMap.get(msgId);
```

**Note:** the MessageID retrieved is NOT deleted from the XC10 data grid in this scenario using the code. This data will be “cleaned up” using the data eviction policy set for the data grid.

18. This section sets the ReplyToQ field in the output message, and resets the value of the MessageID to the original value.

```
// Set the ReplyToQ field
outMessage.getRootElement().getFirstElementByPath("/MQMD/ReplyToQ").
setValue(replyToQ);

// Set the MsgId back to what it was before
outMessage.getRootElement().getFirstElementByPath("/MQMD/MsgId").set
Value(getBytes(msgId));
```

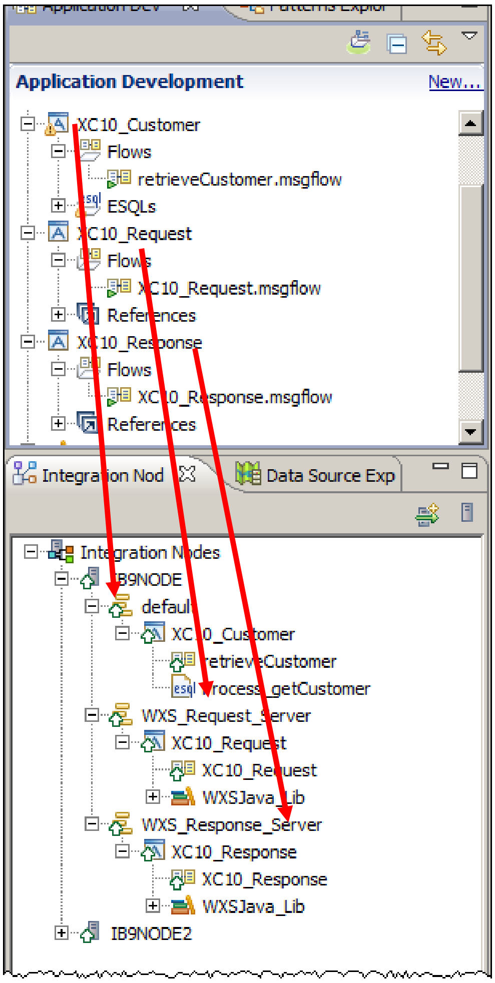
19. **Save** and close the RetrieveFromCache java code.

# 6. Deploy and test the applications

This part of the lab will test the caching applications, with all applications deployed to a single Integration Node. To demonstrate the global cache, each of the three applications will be deployed into a separate Integration Servers, as follows:

1. Default – XC10_Customer
2. WXS_Request_Server – XC10_Request
3. WXS_Response_Server – XC10_Response

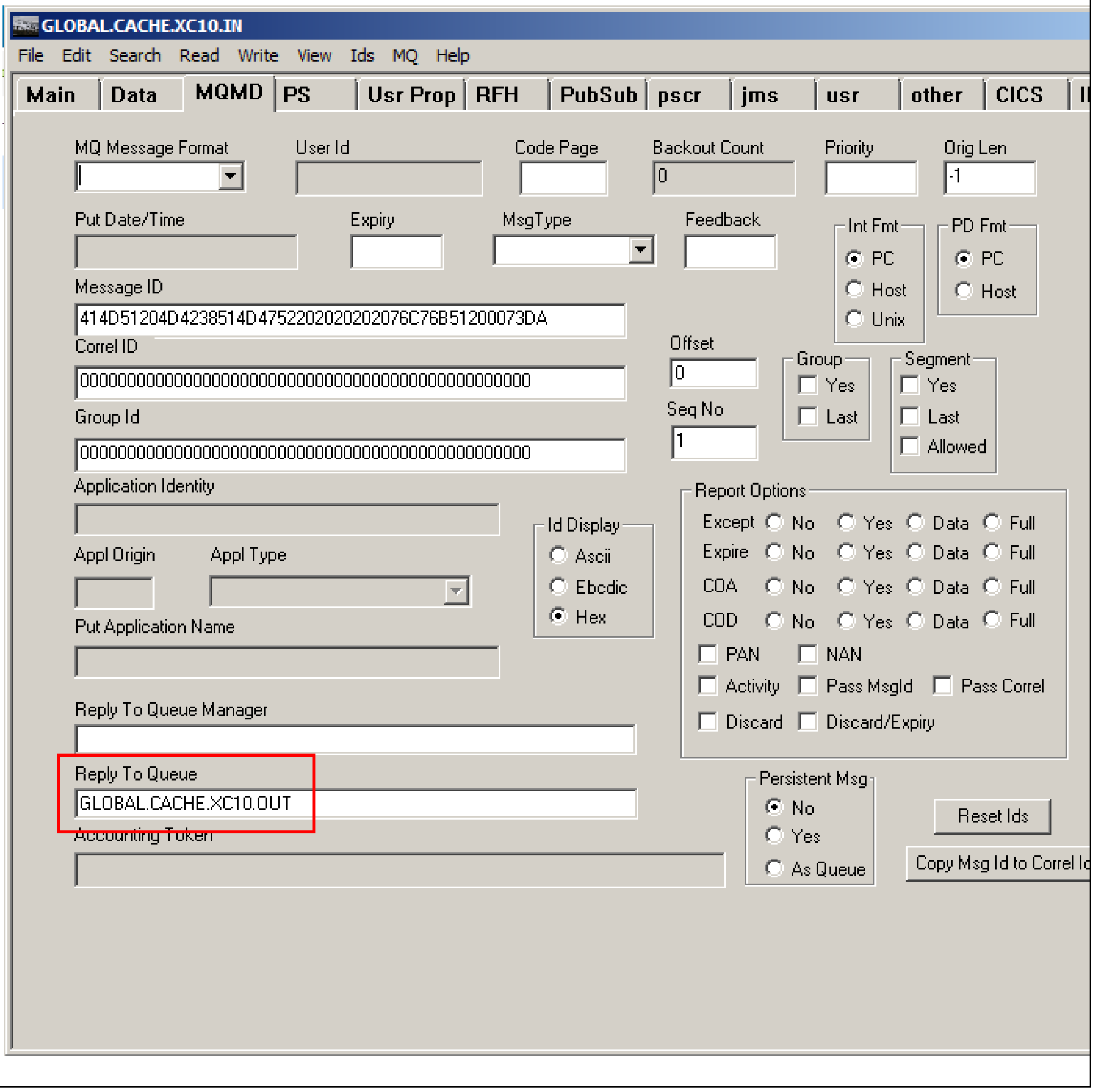
Note: If you have previously deployed the embedded global cache applications into the Integration Servers please delete these applications from the Integration Servers so that you have a "clean" working environment for this XC10 lab.

| | |
|---|---|
| 4. | (If they don't already exist), create two new Integration Servers , WXS_Request_Server and WXS_Response_Server. (Right-click the IB9NODE, and select "New Integration Server".) |
| 5. | Deploy the applications as shown above. Drag and drop the application into the required Integration Servers. |
| 6. | When all three applications are deployed, the Integration Node should look something like this:<br><br> Application Development<br>New...<br>XC10_Customer<br>Flows<br>retrieveCustomer.msgflow<br>ESQLs<br>XC10_Request<br>Flows<br>XC10_Request.msgflow<br>References<br>XC10_Response<br>Flows<br>XC10_Response.msgflow<br>References<br>Integration Nod<br>Data Source Exp<br>Integration Nodes<br>IB9NODE<br>default<br>XC10_Customer<br>retrieveCustomer<br>process_getCustomer<br>WXS_Request_Server<br>XC10_Request<br>XC10_Request<br>WXSJava_Lib<br>WXS_Response_Server<br>XC10_Response<br>XC10_Response<br>WXSJava_Lib<br>IB9NODE2 |
| 7. | Open RFHUtil, and open the file named `c:\student\global_cache\data\Customer0001.xml`. |

8. This data contains a CustomerID with a value of 0001. The CustomerID will be used to retrieve customer data from the SUBREG1 database, by the XC10_Customer application. To ensure the request/reply scenario works, you must specify the name of the reply queue, which is the name of the queue where the final output will appear (ie. the Response application will write to this queue).

On the MQMD tab, set the “Reply to Queue” to `GLOBAL.CACHE.XC10.OUT`.

On the Main tab, write the message to the input queue `GLOBAL.CACHE.XC10.IN`.



Backout Count: 0

Orig Len: -1

Int Fmt: ● PC ○ Host ○ Unix

PD Fmt: ● PC ○ Host

Message ID: 414D51204D4238514D4752202020202076C76B51200073DA

Correl ID: 000000000000000000000000000000000000000000000000

Group Id: 000000000000000000000000000000000000000000000000

Offset: 0

Seq No: 1

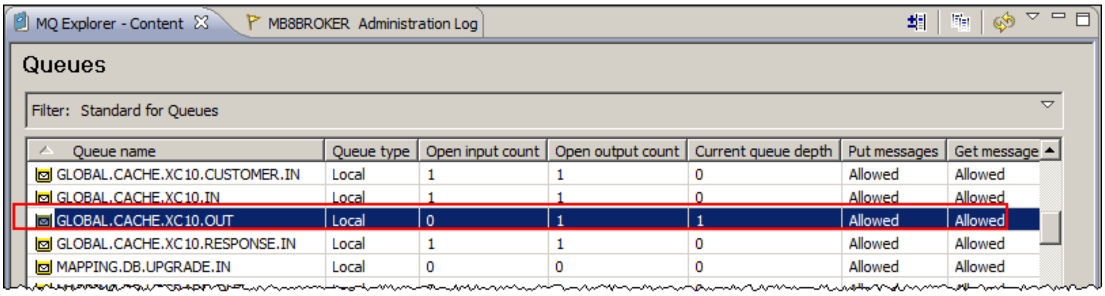
Id Display: ○ Ascii ○ Ebcdic ● Hex

Reply To Queue: GLOBAL.CACHE.XC10.OUT

Persistent Msg: ● No ○ Yes ○ As Queue

9. In MQ Explorer, you should see that the message count for `GLOBAL.CACHE.XC10.OUT` has been incremented by 1.

This demonstrates that the Request and Response applications both have access to the same shared data (the name of the reply queue), even though they are running in separate execution groups. This is provided by the built-in global cache.



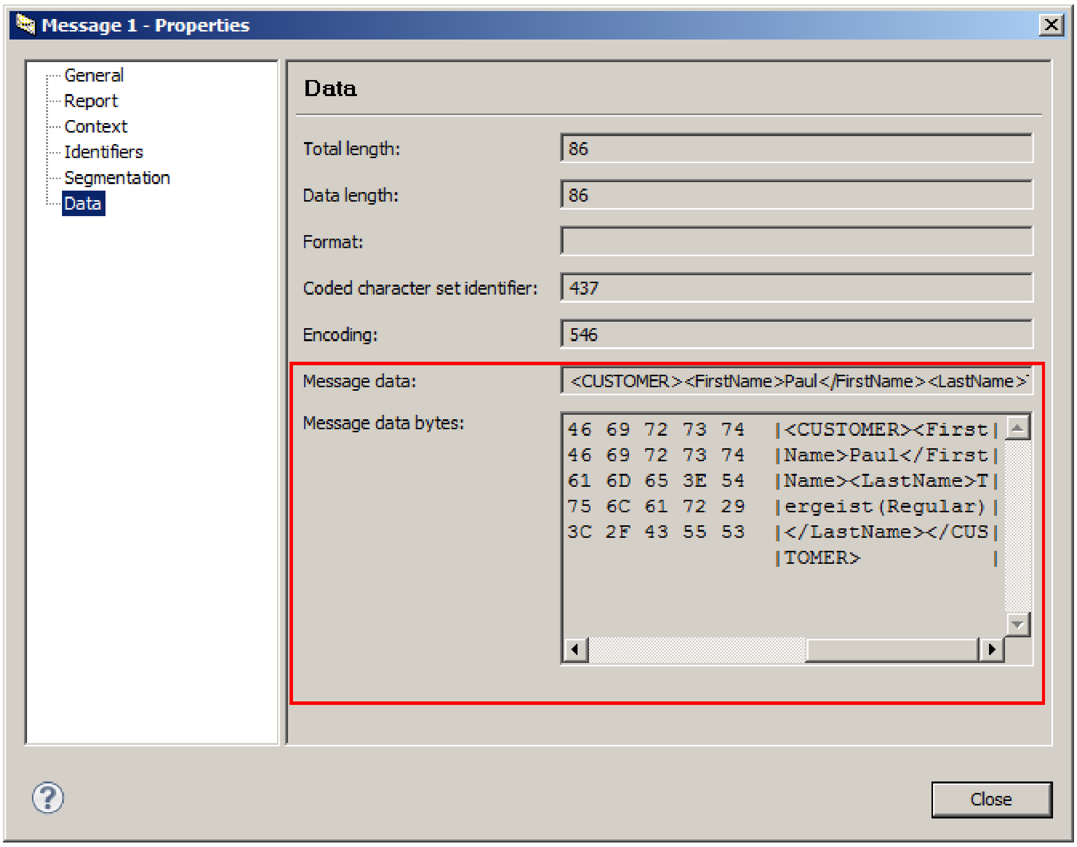
Queues

Filter: Standard for Queues

| Queue name | Queue type | Open input count | Open output count | Current queue depth | Put messages | Get message |
|---|---|---|---|---|---|---|
| GLOBAL.CACHE.XC10.CUSTOMER.IN | Local | 1 | 1 | 0 | Allowed | Allowed |
| GLOBAL.CACHE.XC10.IN | Local | 1 | 1 | 0 | Allowed | Allowed |
| GLOBAL.CACHE.XC10.OUT | Local | 0 | 1 | 1 | Allowed | Allowed |
| GLOBAL.CACHE.XC10.RESPONSE.IN | Local | 1 | 1 | 0 | Allowed | Allowed |
| MAPPING.DB.UPGRADE.IN | Local | 0 | 0 | 0 | Allowed | Allowed |

10. Right click on `GLOBAL.CACHE.XC10.OUT` to browse the messages. Double click on the message to open the message properties.

In the data section you will see a record from the CUSTOMER table.



Data

Total length: 86

Data length: 86

Format:

Coded character set identifier: 437

Encoding: 546

Message data: <CUSTOMER><FirstName>Paul</FirstName><LastName>

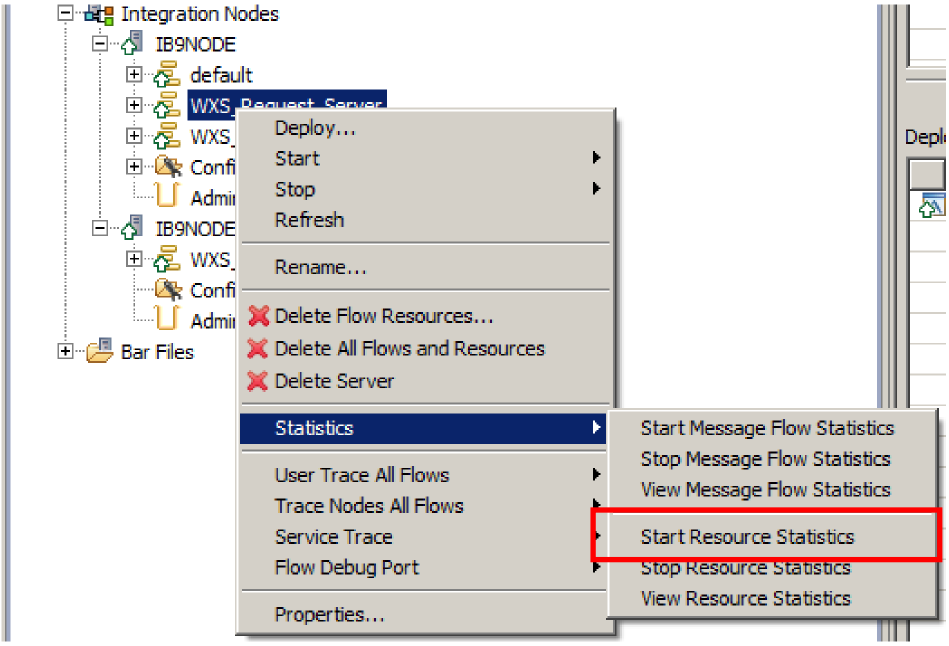
Message data bytes:

```
46 69 72 73 74    |<CUSTOMER><First|
46 69 72 73 74    |Name>Paul</First|
61 6D 65 3E 54    |Name><LastName>T|
75 6C 61 72 29    |ergeist(Regular)|
3C 2F 43 55 53    |</LastName></CUS|
                  |TOMER>          |
```

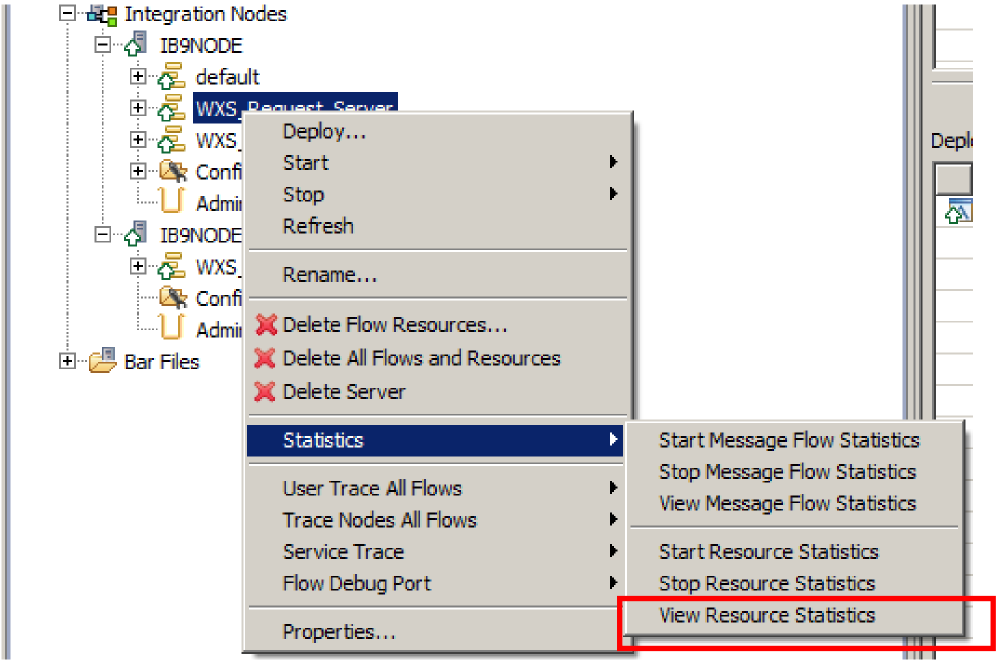
## 6.1 Show the Cache Resource Statistics in Integration Explorer

1. Now check the Resource Statistics for the three execution groups.

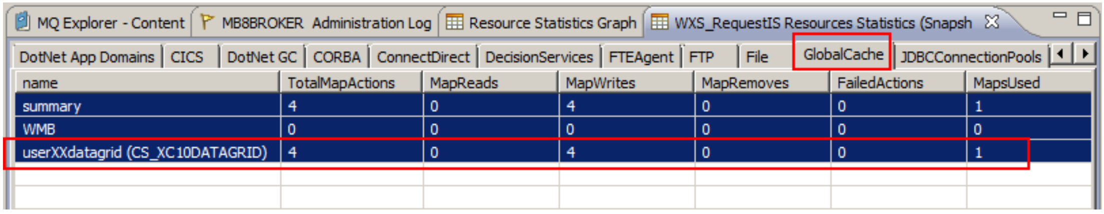
In Integration Explorer, right-click the WXS_Request_Server execution group, and select Statistics, Start Resources Statistics:



2. Open the Resource Views, (right-click  WXS_Request_Server again, and select Statistics > View Resource Views:

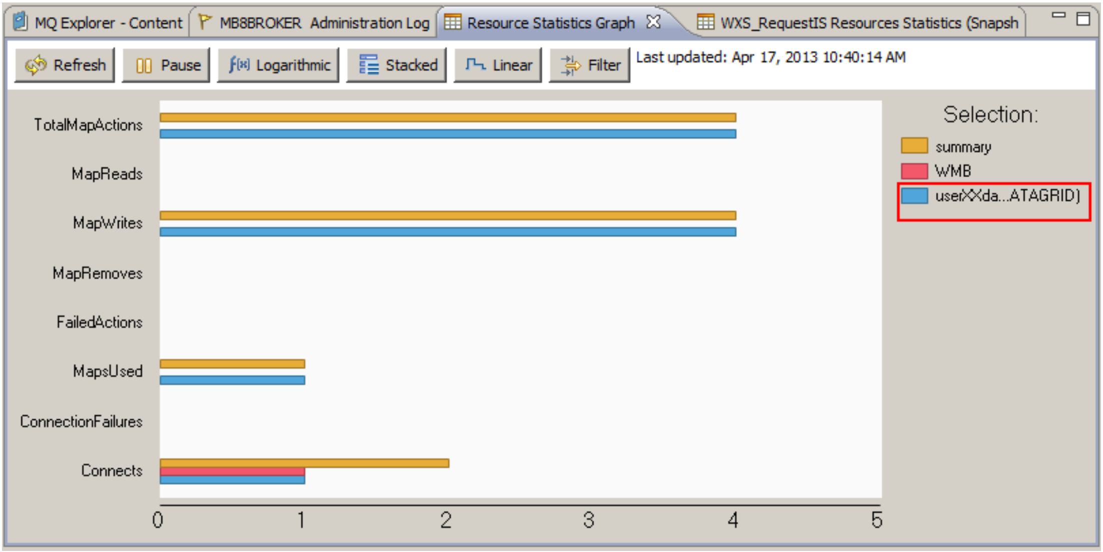


3. Both the graphical and tabular views will open. On the tabular view, click the GlobalCache tab. You will see that the MapWrites count for the XC10 data grid that you are writing to will increment (*try using writing a few more messages using RFHUTIL*) :



| name | TotalMapActions | MapReads | MapWrites | MapRemoves | FailedActions | MapsUsed |
|---|---|---|---|---|---|---|
| summary | 4 | 0 | 4 | 0 | 0 | 1 |
| WMB | 0 | 0 | 0 | 0 | 0 | 0 |
| userXXdatagrid (CS_XC10DATAGRID) | 4 | 0 | 4 | 0 | 0 | 1 |

And the corresponding Resource statistics Graphical view:



Last updated: Apr 17, 2013 10:40:14 AM

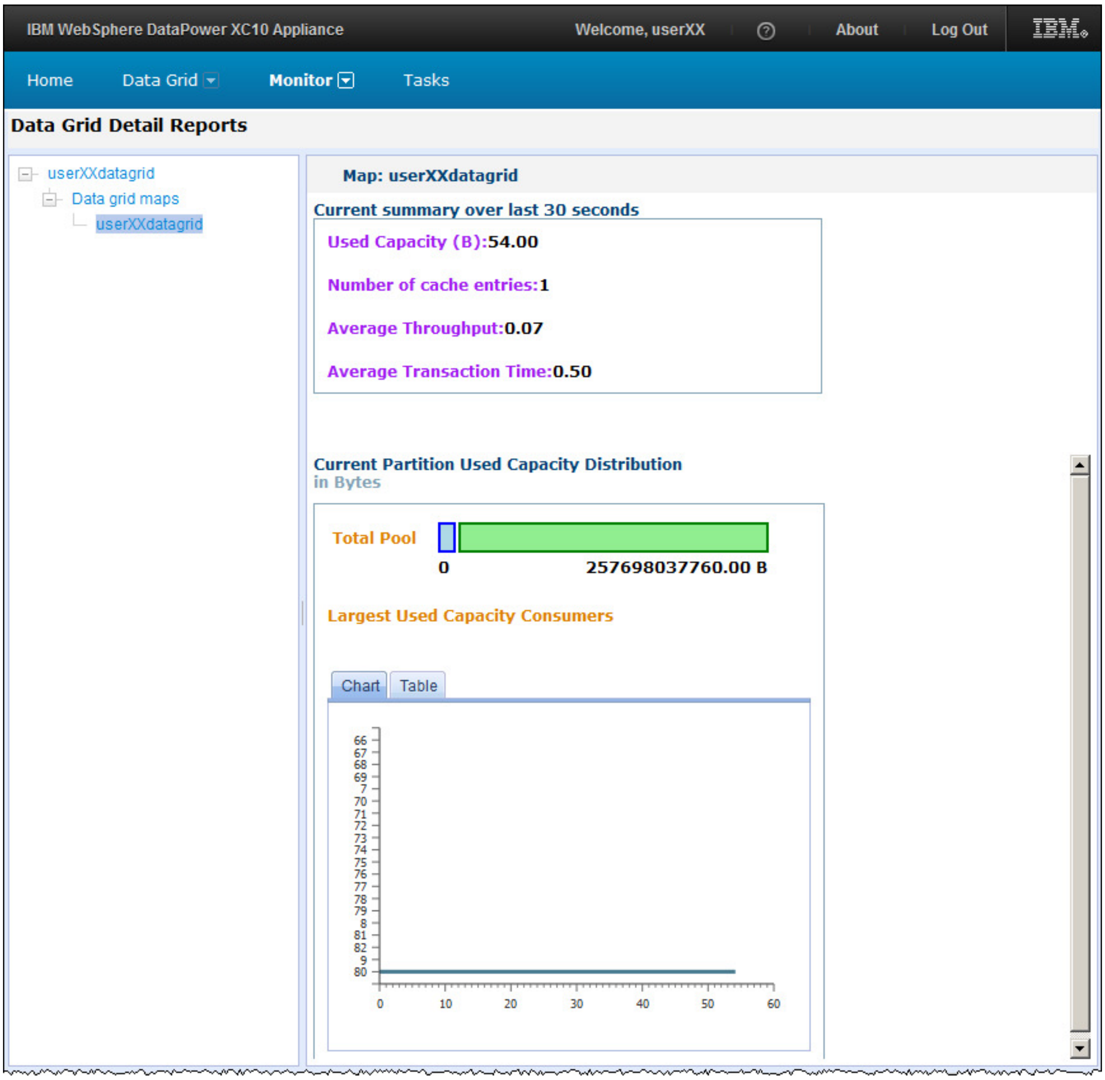
4. Performing the same tasks on the `WXS_Response_Server` will show the GlobalCache Reads count being incremented.

## 6.2 View the XC10 data grid/map statistics.

When using the XC10 data grid for caching the mqsicacheadmin command is not useful since the command shows statistics for the use of the embedded global cache. The web based XC10 configuration tool provides monitoring statistics reports for the data grids on your XC10 device.

1. Sign on to the XC10 using the userid/password you have been using for the XC10.

   Click the "`Monitor`" link and choose > "`Data Grid Detail Reports`", In the left navigation expand `userXXdatagrid > Data grid maps` and click on "`userXXdatagrid`" to show the detailed report for your data grid:



Data Grid Detail Reports

Map: userXXdatagrid

Current summary over last 30 seconds

Used Capacity (B):54.00

Number of cache entries:1

Average Throughput:0.07

Average Transaction Time:0.50

Current Partition Used Capacity Distribution
in Bytes

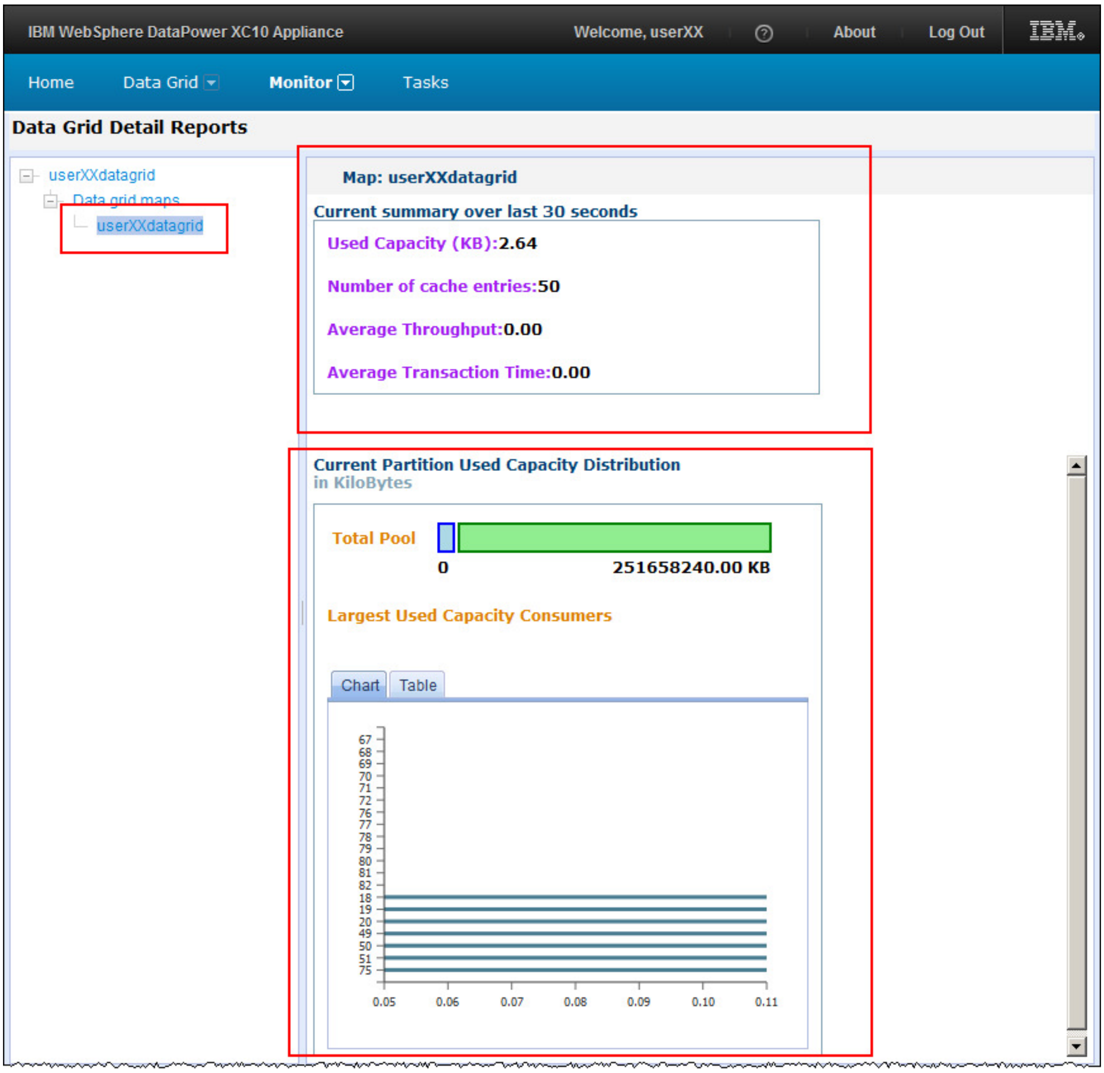
Total Pool 0 257698037760.00 B

Largest Used Capacity Consumers

2. In the Integration Explorer have the Resource statistics views open.

3. Using RFHUTIL write lots of messages (>30) to the `GLOBAL.CACHE.XC10.IN` queue. This will cause the XC10 application to write entries to the XC10 data grid. (ensure all the iterations of the application result in messages being written to the queue `GLOBAL.CACHE.XC10.OUT`

4. On the web based XC10 configuration tool you will begin to see the "Detailed Report" for your map fill with data (click the name of the report to see the updates):



Data Grid Detail Reports

Map: userXXdatagrid

Current summary over last 30 seconds

Used Capacity (KB):2.64

Number of cache entries:50

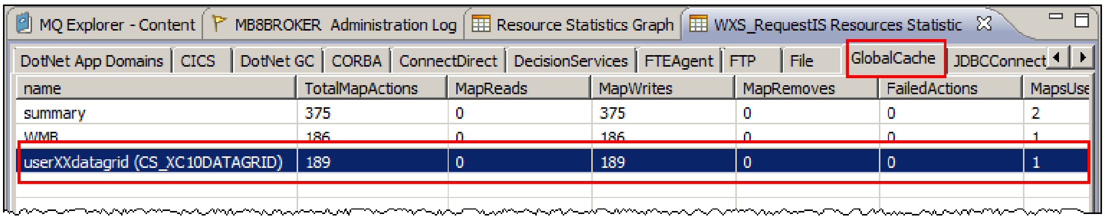
Average Throughput:0.00

Average Transaction Time:0.00

Current Partition Used Capacity Distribution in KiloBytes

Largest Used Capacity Consumers

5. Switch to the Integration Explorer and review the WXS_Request_Server statistics. You will also see that the statistics for Mapwrites for "`userXXdatagrid`" will have increased:

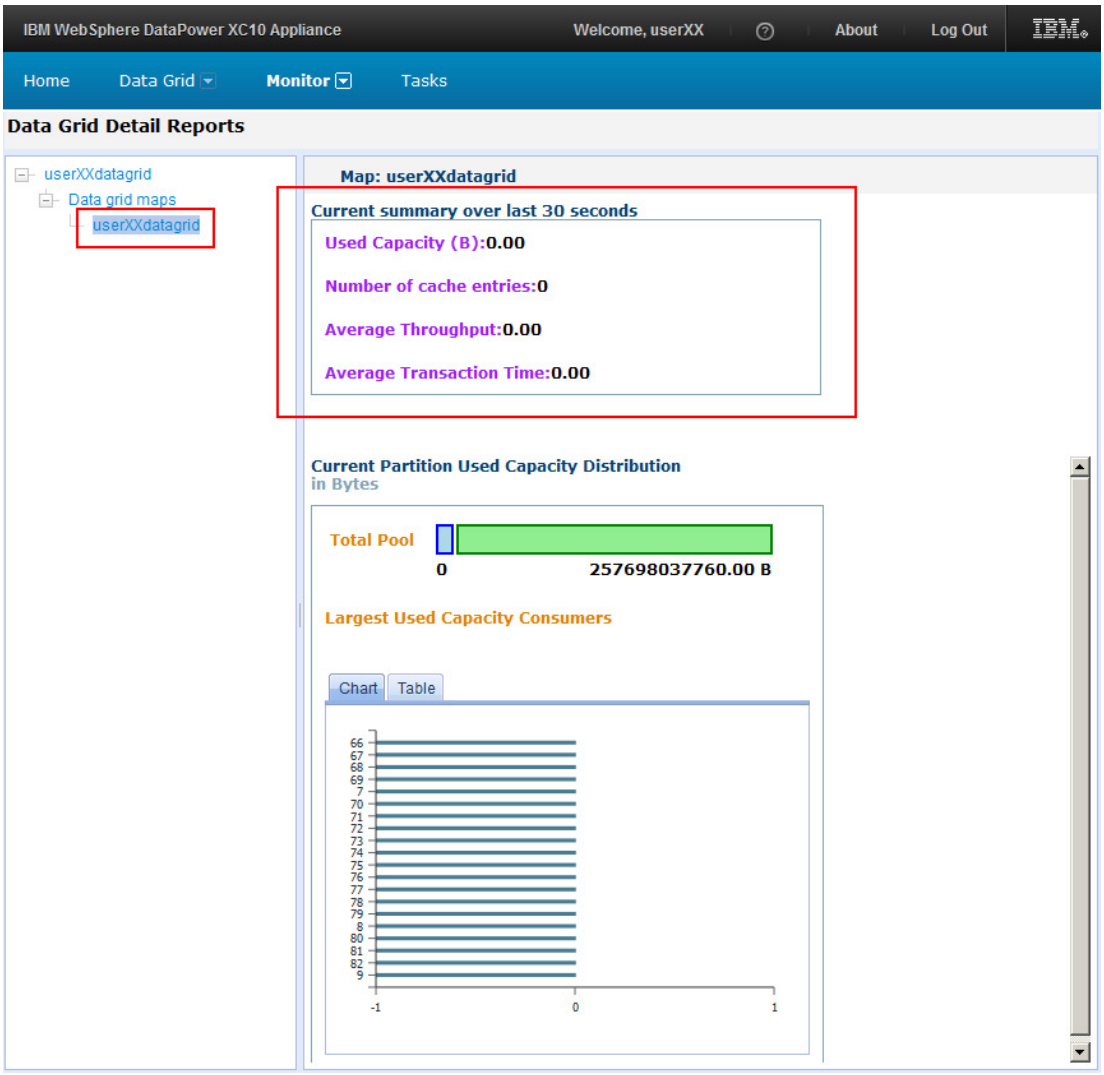


| name | TotalMapActions | MapReads | MapWrites | MapRemoves | FailedActions | MapsUse |
|---|---|---|---|---|---|---|
| summary | 375 | 0 | 375 | 0 | 0 | 2 |
| WMB | 186 | 0 | 186 | 0 | 0 | 1 |
| userXXdatagrid (CS_XC10DATAGRID) | 189 | 0 | 189 | 0 | 0 | 1 |

6. Now take a 5 minute break (no really you've earned it, you got this far :o)

Remember when we set up the data grid – we set the "Time to Live" in the data grid (Advanced attributes section). We set the value at 300 seconds (5 minutes) with an eviction policy of "Last Access Time" meaning that data not accessed for more than 5 minutes will be deleted from the map by the XC10.

7. When you come back from your break (shouldn't be hard for it to last 5 minutes or longer ;-) Refresh the Detailed Map Report on the XC10 web admin tool (sign on again if it prompts you). You will see that the data entries and Chart details in the report have changed showing that the grid contains no data:



Data Grid Detail Reports

Map: userXXdatagrid

Current summary over last 30 seconds

Used Capacity (B):0.00

Number of cache entries:0

Average Throughput:0.00

Average Transaction Time:0.00

Current Partition Used Capacity Distribution
in Bytes

Total Pool 0 257698037760.00 B

Largest Used Capacity Consumers

Write a few more messages to the `GLOBAL.CACHE.XC10.IN` queue using RFHUTIL and refresh the detailed report again to see that the cache is still functioning correctly.

This concludes the Global Cache using a DataPower XC10 appliance Lab

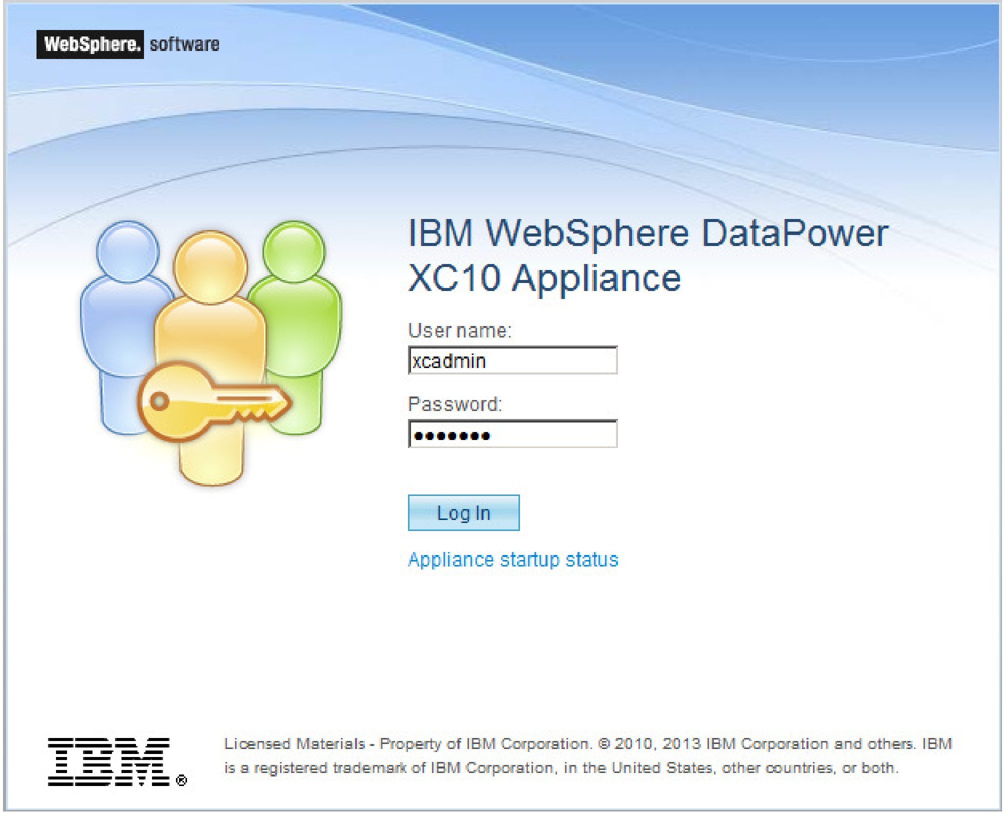
# 7. Appendix A

## 7.1 Using DataPower XC10 at V2.5 Firmware

**If your XC10 firmware is at V2.5 there is a small change to the configuration settings that is required before IBM Integration Bus V9.0 can communicate with your XC10 appliance. Changing this setting requires a restart of the XC10 appliance services which will result in loss of all cached data in all Data Grids on the appliance. Ensure nobody else is actively using the appliance before you attempt this change.**

**This section of the lab guide assumes that you have already done "Connectivity to a DataPower XC10 appliance" (*ie you know the details of the XC10 appliance you are using and have administrator access to the appliance*).**

1. Sign on to the XC10 Web Administration GUI. Log in using "User name" `xcadmin` and password `xcadmin` :



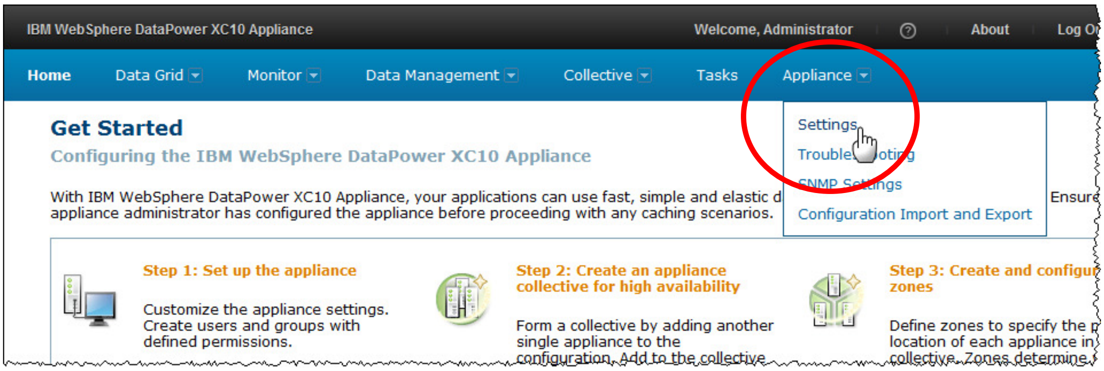
IBM WebSphere DataPower XC10 Appliance

User name: xcadmin

Password: •••••••

Licensed Materials - Property of IBM Corporation. © 2010, 2013 IBM Corporation and others. IBM is a registered trademark of IBM Corporation, in the United States, other countries, or both.

2. On the main menu, click “Appliance” then “Settings”:



**Get Started**

**Configuring the IBM WebSphere DataPower XC10 Appliance**

With IBM WebSphere DataPower XC10 Appliance, your applications can use fast, simple and elastic d… Ensure
appliance administrator has configured the appliance before proceeding with any caching scenarios.

**Step 1: Set up the appliance**

Customize the appliance settings. Create users and groups with defined permissions.

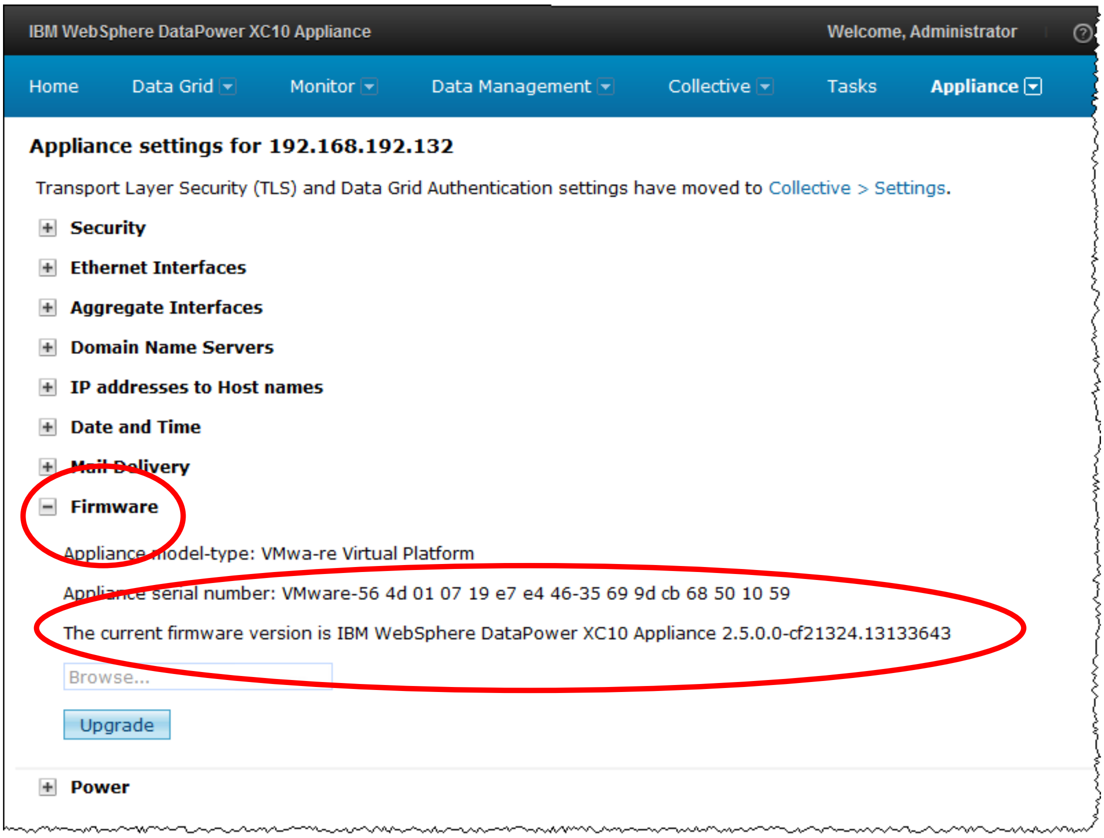
**Step 2: Create an appliance collective for high availability**

Form a collective by adding another single appliance to the configuration. Add to the collective

**Step 3: Create and configu zones**

Define zones to specify the p location of each appliance in collective. Zones determine

3. Click Firmware to verify the version:



**Appliance settings for 192.168.192.132**

Transport Layer Security (TLS) and Data Grid Authentication settings have moved to Collective > Settings.

⊞ **Security**

⊞ **Ethernet Interfaces**

⊞ **Aggregate Interfaces**

⊞ **Domain Name Servers**

⊞ **IP addresses to Host names**

⊞ **Date and Time**

⊞ **Mail Delivery**

⊟ **Firmware**

Appliance model-type: VMwa-re Virtual Platform

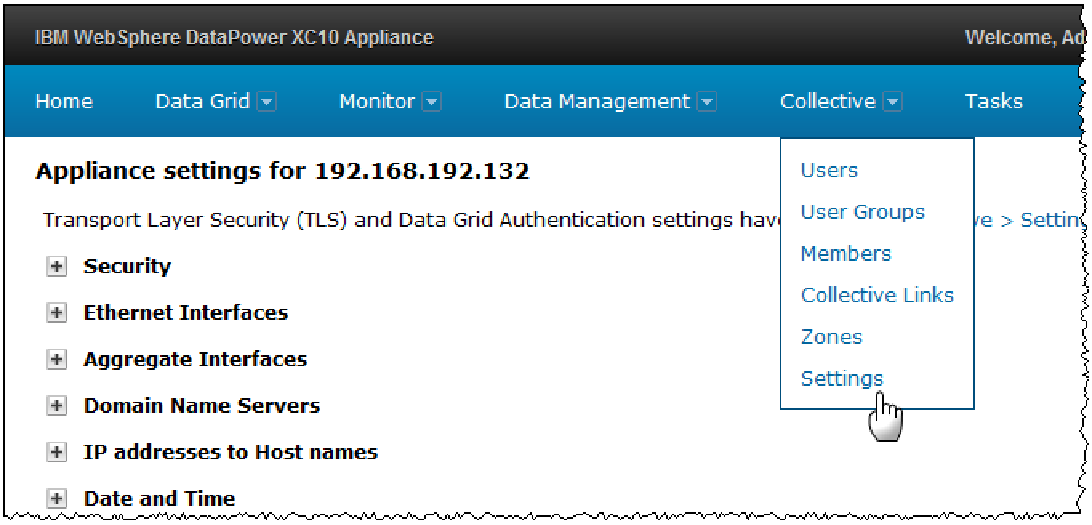
Appliance serial number: VMware-56 4d 01 07 19 e7 e4 46-35 69 9d cb 68 50 10 59

The current firmware version is IBM WebSphere DataPower XC10 Appliance 2.5.0.0-cf21324.13133643

⊞ **Power**

4. If you are at version 2.5 you will need to configure the “Transport and storage mode” (in Collective settings) in order for IBM Integration Bus to communicate with the XC10 appliance.

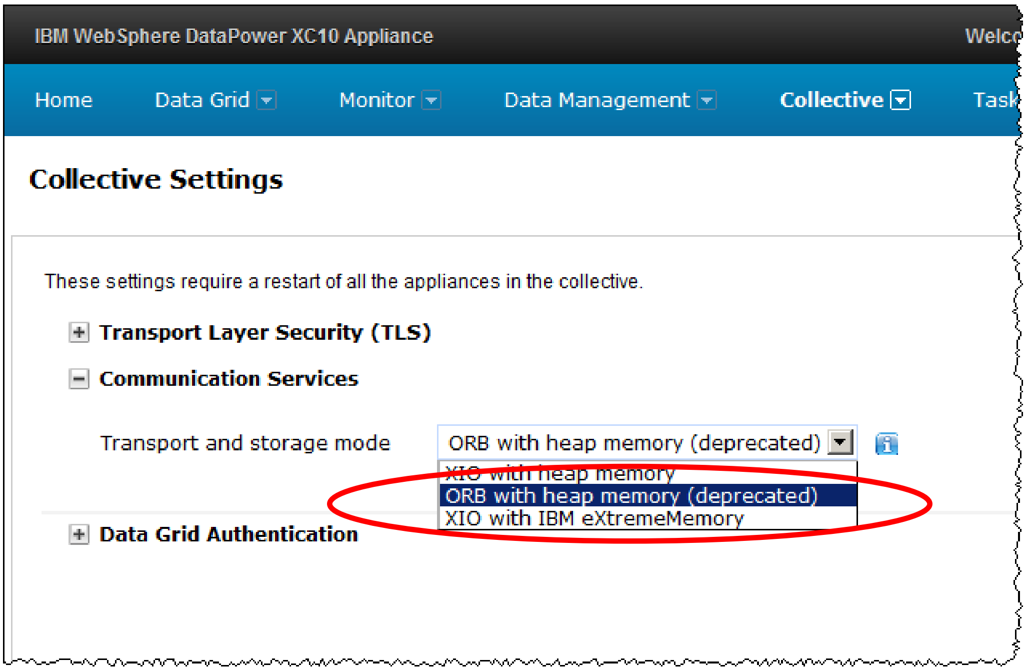
5. Click “Collective” then “settings”:



Appliance settings for 192.168.192.132

Transport Layer Security (TLS) and Data Grid Authentication settings hav ve > Settin

6. Change the “Transport and storage mode” to “ORB with heap memory (deprecated)”:

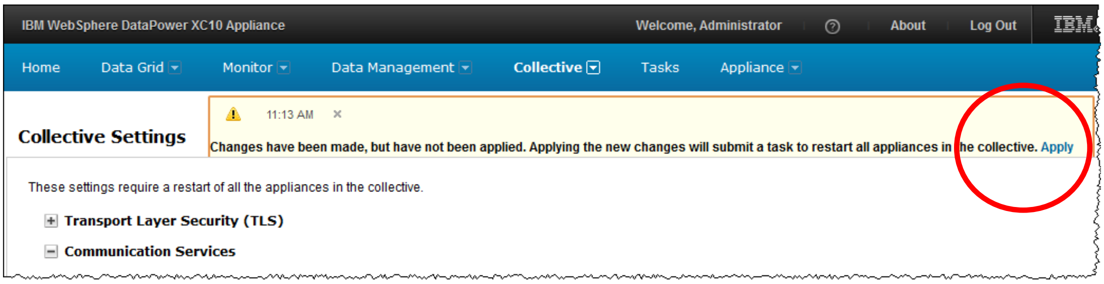

IBM WebSphere DataPower XC10 Appliance
Welco
Home
Data Grid
Monitor
Data Management
Collective
Task
Transport Layer Security (TLS)
Communication Services
XIO with heap memory
ORB with heap memory (deprecated)
XIO with IBM eXtremeMemory
Data Grid Authentication

Collective Settings

These settings require a restart of all the appliances in the collective.

Transport and storage mode: ORB with heap memory (deprecated)

7. Click “Apply” when the message appears to warn about restarting the machine:

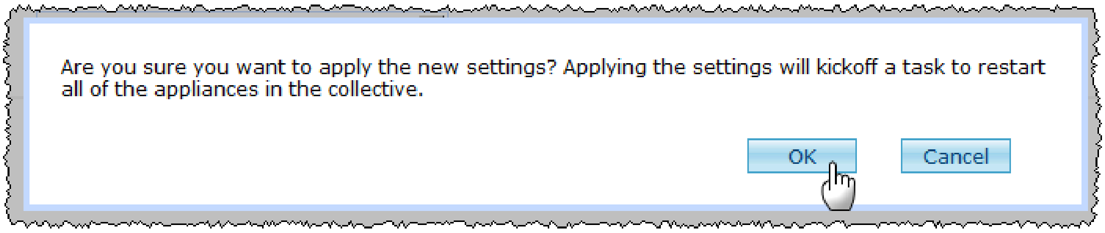


Collective Settings

Changes have been made, but have not been applied. Applying the new changes will submit a task to restart all appliances in the collective. Apply

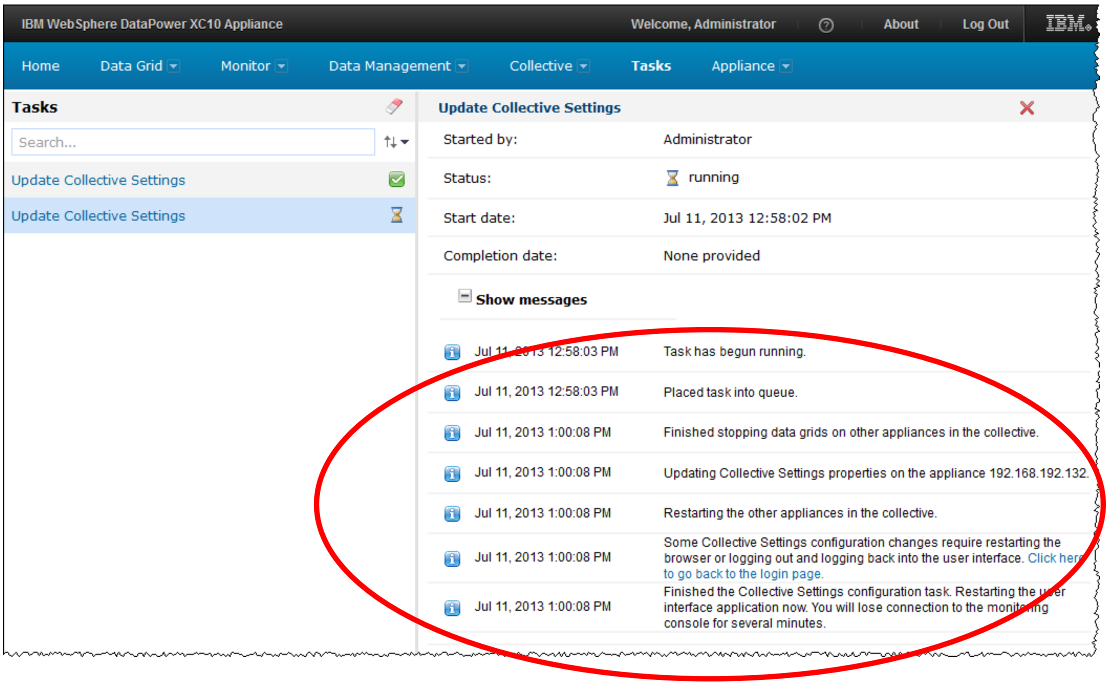
These settings require a restart of all the appliances in the collective.

8. Click OK when prompted:



Are you sure you want to apply the new settings? Applying the settings will kickoff a task to restart all of the appliances in the collective.

9. The gui will redirect you to the task view where you will see that the appliance will be restarted. After a few minutes the task page will indicate that the ,monitoring user interface will be lost as a result of the configuration change:



Update Collective Settings

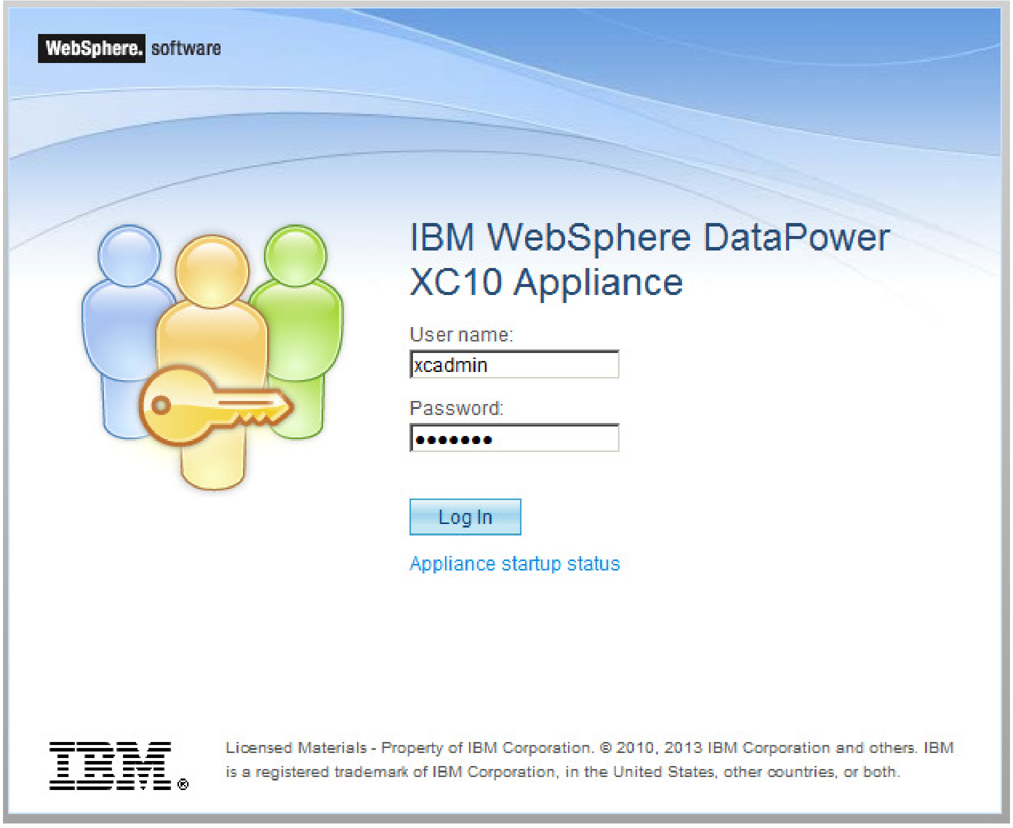
| | |
|---|---|
| Started by: | Administrator |
| Status: | running |
| Start date: | Jul 11, 2013 12:58:02 PM |
| Completion date: | None provided |

Show messages

| | |
|---|---|
| Jul 11, 2013 12:58:03 PM | Task has begun running. |
| Jul 11, 2013 12:58:03 PM | Placed task into queue. |
| Jul 11, 2013 1:00:08 PM | Finished stopping data grids on other appliances in the collective. |
| Jul 11, 2013 1:00:08 PM | Updating Collective Settings properties on the appliance 192.168.192.132. |
| Jul 11, 2013 1:00:08 PM | Restarting the other appliances in the collective. |
| Jul 11, 2013 1:00:08 PM | Some Collective Settings configuration changes require restarting the browser or logging out and logging back into the user interface. Click here to go back to the login page. |
| Jul 11, 2013 1:00:08 PM | Finished the Collective Settings configuration task. Restarting the user interface application now. You will lose connection to the monitoring console for several minutes. |

10. Reload the web interface pages until you are promoted with the sign on page again (this may take a few minutes whilst the web ui interface on the appliance is restarting).

Sign on using xcadmin/xcadmin:

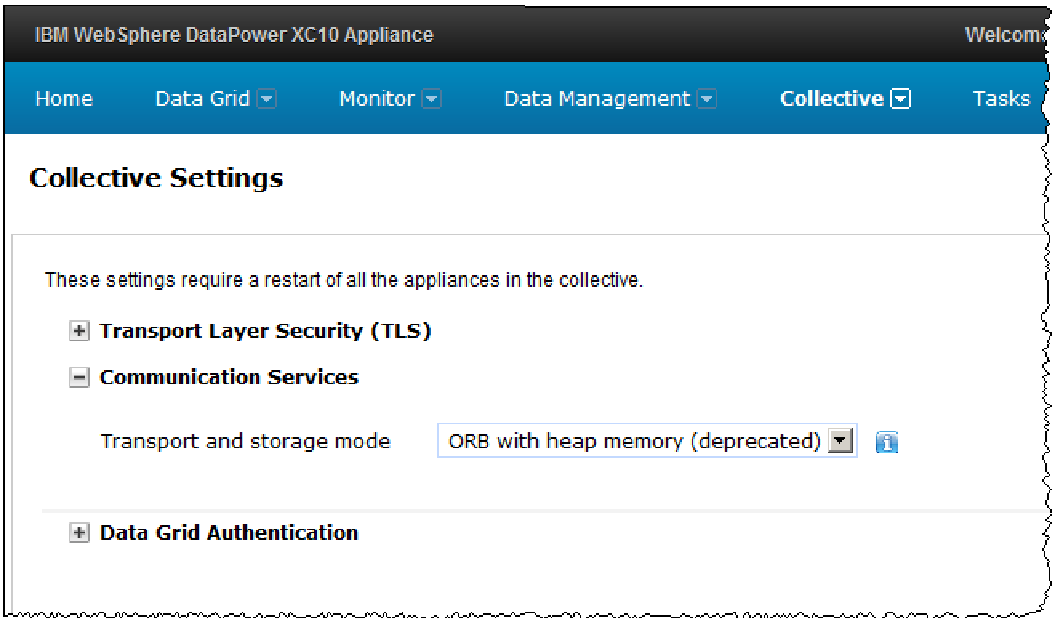


User name: xcadmin

Password: •••••••

Licensed Materials - Property of IBM Corporation. © 2010, 2013 IBM Corporation and others. IBM is a registered trademark of IBM Corporation, in the United States, other countries, or both.

11. Click “Collective”, then “settings” again and ensure that the Transport and storage mode is set to “ORB with heap memory (deprecated)”:



Collective Settings

These settings require a restart of all the appliances in the collective.

⊞ Transport Layer Security (TLS)

⊟ Communication Services

Transport and storage mode: ORB with heap memory (deprecated)

⊞ Data Grid Authentication

12. You are now ready to connect IBM Integration Bus with your XC10 appliance at V2.5 level.