

IBM WebSphere® Partner Gateway V6.1 – Lab exercise

## Partner migration

What this exercise is about .....	1
Lab requirements .....	1
What you should be able to do .....	1
Exercise instructions .....	2
Part 1: Introduction.....	3
Part 2: Extract and configure the migration tool.....	6
Part 3: Import WebSphere Partner Gateway configuration data .....	8
Part 4: Check the imported configuration.....	9
Part 5: Create independent and dependent configurations .....	12
What you did in this exercise .....	16

### What this exercise is about

The objective of this lab is to provide you with an understanding on how to selectively export and import WebSphere Partner Gateway configuration data

### Lab requirements

List of system and software required for the student to complete the lab.

- WebSphere Application Server Base version 6.1 Installed
- WebSphere Partner Gateway V6.1 installed (Simple or Distributed Mode)
- DB2® Enterprise Server V8.2 or higher

### What you should be able to do

At the end of this lab you should be able to selectively:

- Configure the Migration tool
- Export WebSphere Partner Gateway configuration data
- Import WebSphere Partner Gateway configuration data

## Exercise instructions

Some instructions in this lab may be Windows® operating-system specific. If you plan on running the lab on an operating-system other than Windows, you will need to run the appropriate commands, and use appropriate files (.sh vs. .bat) for your operating system. The directory locations are specified in the lab instructions using symbolic references, as follows:

Reference variable	Windows location	Linux® location
<WAS_HOME>	C:\IBM\WAS61	/opt/IBM/WAS61
<WPG_HOME>	C:\IBM\WPG61\wpghubsimple – for simple mode install  C:\IBM\WPG61\wpghubappsprofile – for distributed mode install	/opt/IBM/WPG61
<LAB_FILES>	C:\WPG61\Labfiles	/tmp/WPG61\Labfiles
<MIGRATION_UPLITY_ROOT>	C:\IBM\Migration\bcgmigrate	/opt/IBM/Migration/bcgmigrate
<JAVA_HOME>	C:\IBM\WAS61\java	/opt/IBM/WAS61/java
<TEMP>	C:\temp	/tmp

**Windows users' note:** When directory locations are passed as parameters to a Java program such as EJBdeploy or wsadmin, it is necessary to replace the backslashes with forward slashes to follow the Java convention. For example, C:\LabFiles61\ would be replaced by C:/LabFiles61/

---

## Part 1: Introduction

WebSphere Partner Gateway V6.1 provides configuration migration utility that enables you to selectively export and import WebSphere Partner Gateway configuration data. This differs from other data-moving options like database backup and restore because data is selectively extracted when it is exported with the utility and a database backup is not normally selective. The configuration migration utility enables you to import configuration without overwriting configuration that is already present on a system, while database restoration from a backup typically overwrites existing data. The configuration migration utility exports selected partner and system definitions into an XML file and a set of supporting files. You can subsequently import these files into another system, moving the configuration across systems.

The migration utility is used primarily to move configuration data from a development and testing system to a production system. But you can also load configuration data from an XML file that you create based on the XML schema provided.

There are two options available for running the migration utility

1. A command line interface is provided so the migration utility can be run using a script
2. An API is provided so user-written Java programs can invoke the migration utility

The migration utility is implemented as a stand-alone Java application that calls WebSphere Partner Gateway remotely. The utility is packaged in a .zip file named **BCGMigrationUtil.zip**. This file is installed by the hub installer in this directory **<WPG\_HOME>/console/support**

---

**NOTE:** When transferring data from one system to another, both systems must use the same version of WebSphere Partner Gateway.

---

### Exporting WebSphere Partner Gateway configuration

If you are exporting data, an export options file is required. The export options file specifies what types of data are to be extracted by the utility. A sample export options file named **export.zip** is available in the **BCGMigrationUtil.zip**. This file exports all of the supported configuration data types from a system. The options file can be an XML file or a zip file holding an XML file. The XML must conform to the XML schema **bcgMigrationExport.xsd** located in migration **BCGMigrationUtil.zip**

Some configuration types have no dependencies.

For example, a partner definition can be created without referring to any other configured entity in the system. Other configuration types cannot exist by themselves because they depend on other entities in the system.

For example, a destination is associated with a partner, so it cannot exist unless the partner also exists.

To ensure that dependency items are always available, the content and ordering of the items in export file is important. When an export is performed, any item that has dependencies must be exported after any dependency items. The export XML file reflects this ordering. If you selectively export configuration types, you must ensure that you specify dependency types for all dependent types

### Importing WebSphere Partner Gateway configuration

If you are importing data, you must have data to import. The import file can be produced by exporting data, or you can write your own that contains definitions you want to load into a system. After exporting, the exported

data is contained in a zip file. The zip file includes an XML file that conforms to the XML schema **bcgMigrationImport.xsd** located in migration **BCGMigrationUtil.zip**.

The import zip file contains an XML file, folders for validation, transformation and FA maps and **RoutingObjects** folder. The XML file that conforms to the XML schema **bcgMigrationImport.xsd** located in **BCGMigrationUtil.zip**. This XML file includes data that can be used by the import code to re-create the configuration.

Folders for validation, transformation and FA maps contain the details for the maps.

**RoutingObjects** folder contains **RoutingObjects.zip** that contains the internal representations of the exported routing objects (packages, protocols, and document types)

To ensure that dependency items are always available, the content and ordering of the items in export and import files are important. When an export is performed, any item that has dependencies must be exported after any dependency items. The XML file reflects this ordering. Using the same logic, when the import is performed, the dependency items are imported before the dependent items. So if you define an import file incorrectly, for example, you forget to provide a dependency item or incorrectly defining a dependency item, that item will fail when you attempt an import of it

### **Invoking migration utility from command line:**

```
bcgmigrate [-h host name:bootstrap_port] [-a import|export] [-f file name] [-u userid] [-p password] [-o] [-d 1..5] [-r root_path]
```

h is the host name:bootstrap port where the console component is running

-a is the activity (which can be either import or export)

-f is the fully qualified file name of the export option file or the import configuration file

-u is the WebSphere Partner Gateway userid that has migration permission.

-p is the WebSphere Partner Gateway user password

-o is the overwrite.

-d is the debug level from 1 to 5 where 5 provides the most debug output

-r is the root path where exported data is stored and the log file is written

#### ***Example for Import:***

```
bcgMigrate.bat -h localhost:58809 -a import -f C:\WPG61Labfiles\Migration\import.zip -u hubadmin -p hub1admin -d 5 -r C:\WPG61LabFiles\Migration\output
```

#### ***Example for Export:***

```
bcgMigrate.bat -h localhost:58809 -a export -f C:\WPG61Labfiles\Migration\export.xml -u hubadmin -p hub1admin -d 5 -r C:\WPG61LabFiles\Migration\output
```

```
Usage:
bcgmigrate -h <hostname:portNumber> -a <import | export> -f <fileName.xml> -u <userName> -p
<password> [-r <rootPath>] [-o ] [-d <0..5> ]

Where options are defined as:
-h The URL of the ejb server including the bootstrap port
-a Specify the action: either import or export
-f Specify the full path to the option file
-u Specify the user name of Operator company
-p Specify the password of the hubadmin user
-r Specify the directory name where the log files and any output files will be put.
-o Include this parameter if you wish to overwrite during import.
-d Specify the debug logging level to use: (0-Error, 5-Finest)

Example:
bcgmigrate -h localhost:58809 -a export -f c:/export.xml -u hubadmin -p abc123 -r c:/migrat
ion -d 2
bcgmigrate -h localhost:58809 -a import -f c:/migration/import.zip -u hubadmin -p abc123 -o
-d 2
```

## Part 2: Extract and configure the migration tool

Before you can use the migration utility, you must extract the **BCGMigrationUtil.zip** file on the machine where you will run the utility. After the utility files are extracted to your local file system, perform the following prerequisite steps:

- \_\_\_ 1. In the file system, navigate to <WPG\_HOME>\<profile path>\console\support
- \_\_\_ 2. Locate the **BCGMigrationUtil.zip** file and extract the contents to **C:\IBMMigration**
- \_\_\_ 3. The console component for the WebSphere Partner Gateway system that you will export from or import into must be running. So make sure that the WebSphere Partner Gateway servers are started and the WebSphere Partner Gateway applications are started.

---

**NOTE:** The utility can be run on a different machine than where the console component is installed. This is because the utility accesses the console using the IIOOP (EJB™) protocol over a network. Connectivity between the machines is required, and the console's IIOOP port (typically 58809) has to be available from the machine where the utility runs.

---

- \_\_\_ 4. You need to have Java 5 available on the machine where the utility will run. When you run the command line script to run the utility, it obtains the system environment variable **JAVA\_HOME** for this location. If **JAVA\_HOME** is undefined, then the script will prompt for you to enter the home location for Java 5. To configure the System variable, do the following
  - \_\_\_ a. Right Click on the **My computer** icon and select **properties**
  - \_\_\_ b. In the dialog box that open up, select the **Advanced** tab
  - \_\_\_ c. Click on the **Environment Variables** button
  - \_\_\_ d. In the new dialog box, under **System variables** look for **JAVA\_HOME** variable.
    - 1) If the variable is present and it points to different location than Java 5, select the **JAVA\_HOME** variable and click **Edit** and provide the location as **<WAS\_HOME>\java**
    - 2) the variable is not present, then you need to create the variable. Click **New** under System Variables and provide **Variable name** as **JAVA\_HOME** and **Variable Value** as **<WAS\_HOME>\java**. Click **OK** on the **New Variable** dialog box, Click **OK** on the **Environment Variables** dialog box. Finally click **Ok** on the **System Properties** dialog box
- \_\_\_ 5. Another system environment variable named **MIGRATION\_PATH** needs be set to point to the location where **BCGMigrationUtil.zip** extracted. If the **MIGRATION\_PATH** is undefined, then the script will prompt you to enter a path to this directory. **MIGRATION\_PATH** should point to the directory where the zip file is extracted.

For example, if you extracted the files to the directory **C:\IBMMigration**, then you should set **MIGRATION\_PATH** to **C:\IBMMigration\bcgmigrate**

- \_\_\_ 6. To configure the System variable, do the following
  - \_\_\_ a. Right Click on the **My computer** icon and select **properties**
  - \_\_\_ b. In the dialog box that open up, select the **Advanced** tab
  - \_\_\_ c. Click on the **Environment Variables** button

- \_\_\_ d. In the new dialog box, under **System variables**. Click **New** under System Variables and provide **Variable name** as **MIGRATION\_PATH** and **Variable Value** as **C:\IBM\migration\bcgmigrate**
- \_\_\_ e. Click **OK** on the **New Variable** dialog box
- \_\_\_ f. Click **OK** on the **Environment Variables** dialog box.
- \_\_\_ g. Finally click **Ok** on the **System Properties** dialog box

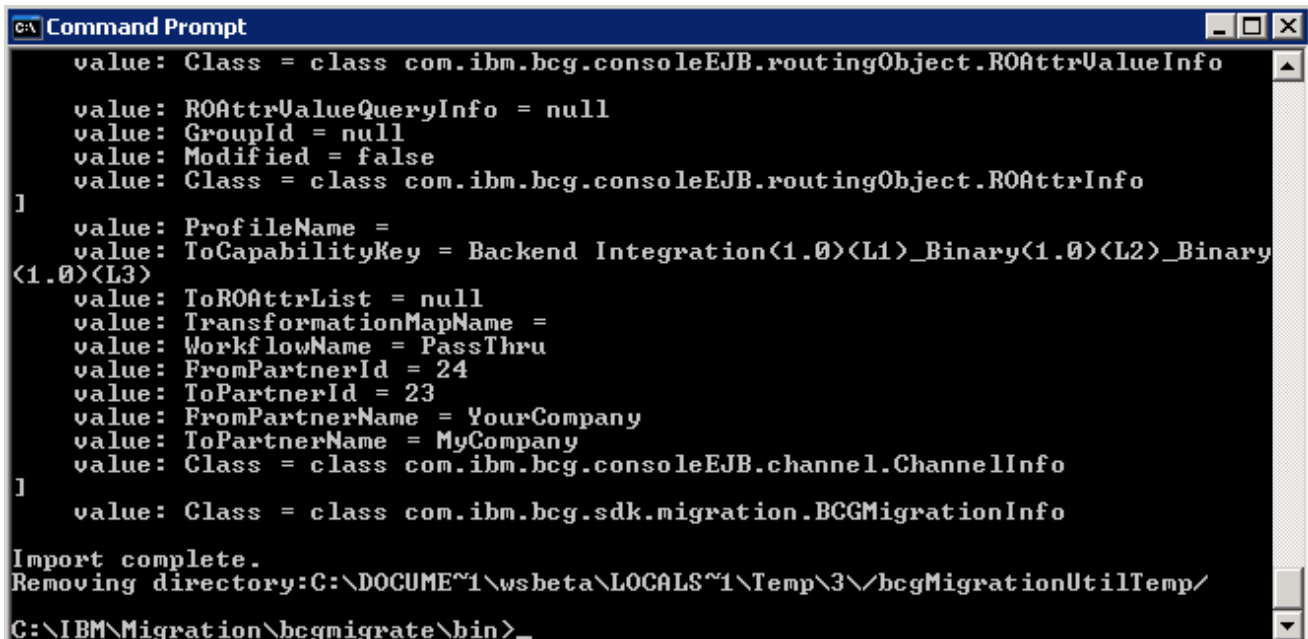
## Part 3: Import WebSphere Partner Gateway configuration data

If you are importing data, you must have data file to import. The import file can be produced by exporting data, or you can write your own that contains definitions you want to load into a system. In this lab scenario, you will use a zip file that is generated by exporting data from a WebSphere Partner Gateway system.

- \_\_\_ 1. Go to <LAB\_FILES>/Migration
- \_\_\_ 2. Locate the **import.zip** file.
- \_\_\_ 3. Open a command prompt window and change directories to **C:\IBM\Migration\bcmigrate\bin**
- \_\_\_ 4. Use the following command to import the configuration data. You will need to specify user credentials that have the permission to import the data. **hubadmin** user has the permission by default

```
bcgMigrate.bat -h localhost:58809 -a import -f C:\WPG61Labfiles\Migration\import.zip -u hubadmin
-p hub1admin -d 5 -r C:\WPG61LabFiles\Migration\output
```

- \_\_\_ 5. If the import is successful, you will see something similar to the picture below



```

C:\ Command Prompt
value: Class = class com.ibm.bcg.consoleEJB.routingObject.ROAttrValueInfo
value: ROAttrValueQueryInfo = null
value: GroupId = null
value: Modified = false
value: Class = class com.ibm.bcg.consoleEJB.routingObject.ROAttrInfo
]
value: ProfileName =
value: ToCapabilityKey = Backend Integration<1.0><L1>_Binary<1.0><L2>_Binary
<1.0><L3>
value: ToROAttrList = null
value: TransformationMapName =
value: WorkflowName = PassThru
value: FromPartnerId = 24
value: ToPartnerId = 23
value: FromPartnerName = YourCompany
value: ToPartnerName = MyCompany
value: Class = class com.ibm.bcg.consoleEJB.channel.ChannelInfo
]
value: Class = class com.ibm.bcg.sdk.migration.BCGMigrationInfo
Import complete.
Removing directory:C:\DOCUME~1\wsbeta\LOCALS~1\Temp\3\bcgMigrationUtilTemp/
C:\IBM\Migration\bcmigrate\bin>

```

- \_\_\_ 6. If there were errors during the import, you can check the log messages in your output location which is **C:\WPG61LabFiles\Migration\output\bcmigration.log**



## Part 4: Check the imported configuration

The import file you used contains configuration data for

- two receivers (one HTTP and one File Directory),
- two partners (MyCompany and YourCompany) each with two Destinations (one HTTP and one File Directory) and one connection.

In this section you will check for the existence of the data imported.

\_\_\_ 1. Open a Web browser and type the following URL:

Unsecured: **http://<host name>.<domain>:58080/console**

Secure: **https://<host name>.<domain>:58443/console**


Where <host name> and <domain> are the name and location of the computer hosting the Community Console component.

\_\_\_ 2. Log into the Community console providing the appropriate credentials for **hubadmin** user


\_\_\_ 3. Check for the creation of partners and the partner ID's

\_\_\_ a. In the WebSphere Partner Gateway Community Console, navigate to **Account Admin → Profiles → Partner**

\_\_\_ b. Click on the **Search** button. This will list all of the partners defined. You should see the Hub Operator, **YouCompany** and **MyCompany** partners along with any other partners you have already defined.



Partner Search		
Partner Name	<input type="text"/>	
Business ID	DUNS <input type="text"/>	
	<input type="button" value="Search"/>	
	Partner Name	Partner Type
	Hub Operator	Hub Administrator
	IBM	External Partner
	LenPartner	External Partner
	MyCompany	Internal Partner
	YourCompany	External Partner

\_\_\_ 4. Check for the creation of **Destinations** of **YourCompany** and **MyCompany** partners

\_\_\_ a. Select the partner **MyCompany** by clicking on the  icon next to **MyCompany**. This will list the properties of the trading partner **MyCompany**


\_\_\_ b. Click the Destinations option in the top menu. This will list all the Destinations created for the partner **MyCompany**.

\_\_\_ c. You should see two destinations created, one is a file gateway and the other HTTP gateway.

Profile » MyCompany » Destination List			
<a href="#">Create</a> <a href="#">Forward Proxy Support</a> <a href="#">Global Transport Attributes</a>			
Destination Name	Transport	Address	
 MyCompanyFileDestination	File Directory	file:///c:/MyCompany/destination	
 MyCompanyHTTPDestination	HTTP/1.1	http://MyCompany.com/destination	



\_\_ d. Navigate to **Account Admin → Profiles → Partner**

\_\_ e. Click on the **Search** button. This will list all of the partners defined. You should see the Hub Operator, **YouCompany** and **MyCompany** partners along with any other partners you have already defined

\_\_ f. Select the partner **YourCompany** by clicking on the  icon next to **YourCompany**. This will list the properties of the trading partner **YourCompany**




\_\_ g. Click the Destinations option in the top menu. This will list all the Destinations created for the partner **YourCompany**.

\_\_ h. You should see two destinations created, one is a file gateway and the other HTTP gateway

Profile » YourCompany » Destination List			
<a href="#">Create</a> <a href="#">Forward Proxy Support</a> <a href="#">Global Transport Attributes</a>			
Destination Name	Transport	Address	
 FileDestination	File Directory	file:///c:/YourCompany/destination	
 HttpDestination	HTTP/1.1	http://YourCompany.com/destination	

\_\_\_ 5. Check for the creation of **Receivers**

\_\_ a. Navigate to **Hub Admin → Hub Configuration → Receivers**. This should list all the receivers defined. You should see HTTP Receiver and File Receiver along with any others you have already defined

Receiver List			
<a href="#">Create Receiver</a>			
Receiver Name	Transport	Address	
 LenPartnerReceiver	HTTP/S	/bcgreceiver/LenPartnerReceiver	
 HTTPReceiver	HTTP/S	bcgreceiver/receive	
 FileReceiver	File Directory	file:///c:/receive	

\_\_\_ 6. Check for the Participant Connections

\_\_ a. Navigate to **Account Admin → Connections → Partner connections**

\_\_ b. Select **MyCompany** as **Source** and **YourCompany** as **Target**

\_\_ c. Click **Search**. This should list the participant connections you imported

Source		Target	
<input type="text" value="MyCompany"/>	<input type="button" value="Search"/>	<input type="button" value="Reset"/>	<input type="text" value="YourCompany"/>
<hr/>			
Enabled	<input type="button" value="B2B Capabilities"/>	<input type="button" value="Connection Details"/>	<input type="button" value="B2B Capabilities"/> Deactivate • Help
<hr/>			
<input type="button" value="Package: Backend Integration (1.0)&lt;br/&gt;Protocol: Binary (1.0)&lt;br/&gt;Document Type: Binary (1.0)"/>	<input type="button" value="Activate"/>	<input type="button" value="Package: AS (N/A)&lt;br/&gt;Protocol: Binary (1.0)&lt;br/&gt;Document Type: Binary (1.0)"/>	

You have successfully checked that all the configuration data in the import file has been imported properly.

## Part 5: Create independent and dependent configurations

As discussed in the introduction, some configuration types have no dependencies.

For example, a partner definition can be created without referring to any other configured entity in the system. Other configuration types cannot exist by themselves because they depend on other entities in the system. For example, a destination is associated with a partner, so it cannot exist unless the partner also exists.

In this section, you are going to create a new partner, one file gateway for the partner. Partner profile is an independent configuration type but the destination you create for the partner is dependent on the partner definition. You can export the partner definition individually but you cannot export just the destination without exporting the partner definition

- \_\_\_ 1. In the Community console, navigate to **Account Admin → Profiles → Partner**
- \_\_\_ 2. Click the **Create** link on the right corner
- \_\_\_ 3. In the partner creation screen, provide **OtherCompany** as the **Company Log in Name** and **Partner Display Name**
- \_\_\_ 4. Accept the default **External Partner** for **Partner Type**
- \_\_\_ 5. Click **New** button under **Business ID**
- \_\_\_ 6. Select Type as **Duns**
- \_\_\_ 7. Provide **876534678** as **Identifier** and click **Save**
- \_\_\_ 8. Click the **Destinations** link from the menu and Click **Create** in the next screen
- \_\_\_ 9. Provide **Destination Name** as **OtherCompanyFileDestination**
- \_\_\_ 10. Select **Transport Type** as **File Directory**
- \_\_\_ 11. Provide **file://C:\tmp\otherfgw** as **Address**
- \_\_\_ 12. Click **Save**

In order to export the configuration data, you need to define an export file which should conform to the **bcgMigrationExport.xsd** located in migration **BCGMigrationUtil.zip**. Provided in the <LAB\_FILES>\Migration are two export files **export\_err.xml** and **export.xml**.

You will use these export files to export the data. Parts of the **export\_err.xml** are shown below.

You are exporting

- all the receivers, all the maps(transformation, FA and validation)
- MyCompany profile, MyCompany partners gateway information, contact info, and others
- OtherCompany partner info but nothing else
- Connection info from MyCompany to Your Company.

As you can see below, you are not exporting the **YourCompany** partner configuration

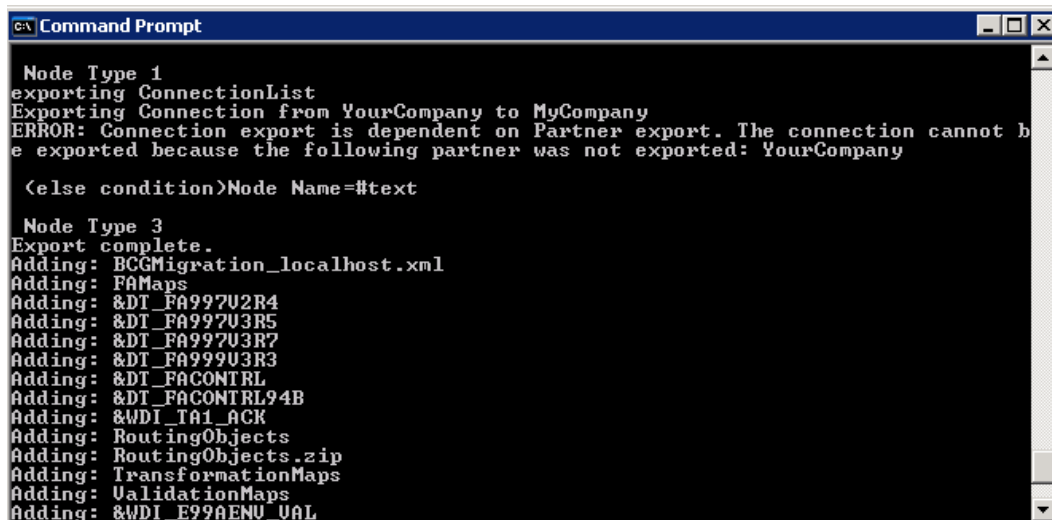
```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ExportInfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="bcgExport.xsd">
  <EnveloperSchedulingInfo>all</EnveloperSchedulingInfo>
  <TransportInfo>all</TransportInfo>
  <DestinationTypeInfo>all</DestinationTypeInfo>
  .....
  .....
  <ValidationMapInfo>all</ValidationMapInfo>
  <TransformationMapInfo>all</TransformationMapInfo>
  <FaMapInfo>all</FaMapInfo>
  <ReceiverInfo>all</ReceiverInfo>
  .....
  <PartnerList>
    <PartnerInfo>
      <Partner>MyCompany</Partner>
      <ContactInfo>all</ContactInfo>
      <PartnerAddressInfo>all</PartnerAddressInfo>
      <ParticipantControllInfo>all</ParticipantControllInfo>
      <GatewayInfo>all</GatewayInfo>
      <DefaultGatewayInfo>all</DefaultGatewayInfo>
    </PartnerInfo>
    <PartnerInfo>
      <Partner>OtherCompany</Partner>
      <ContactInfo>all</ContactInfo>
      <PartnerAddressInfo>all</PartnerAddressInfo>
      <ParticipantControllInfo>all</ParticipantControllInfo>
    </PartnerInfo>
  </PartnerList>
  .....
  <connectionList>
    <ConnectionInfo>
      <FromPartner>YourCompany</FromPartner>
      <ToPartner>MyCompany</ToPartner>
    </ConnectionInfo>
```

```
</connectionList>

</ExportInfo>
```

- \_\_\_ 1. Go to <LAB\_FILES>/Migration
- \_\_\_ 2. Open a command prompt window and change directories to **C:\IBM\Migration\bcgmigrate\bin**
- \_\_\_ 3. Use the following command to import the configuration data. You will need to specify user credentials that have the permission to export the data. **hubadmin** user has the permission by default
 

```
bcgMigrate.bat -h localhost:58809 -a export -f C:\WPG61Labfiles\Migration\export_err.xml -u hubadmin -p hub1admin -d 5 -r C:\WPG61LabFiles\Migration\output
```
- \_\_\_ 4. The messages you see on the console are also logged in the **bcgMigration.log** file in **C:\WPG61LabFiles\Migration\output\** (Which is the output location provided in the command)
- \_\_\_ 5. Also generated is the zip file in the output location that contains the configuration data you exported which can be used to import into another WebSphere Partner Gateway System.
- \_\_\_ 6. Once the export process is completed, if you look at the messages on the command prompt you should see something similar to the picture below.



```

Node Type 1
exporting ConnectionList
Exporting Connection from YourCompany to MyCompany
ERROR: Connection export is dependent on Partner export. The connection cannot be
e exported because the following partner was not exported: YourCompany

<else condition>Node Name=#text

Node Type 3
Export complete.
Adding: BCGMigration_localhost.xml
Adding: FAMaps
Adding: &DT_FA997U2R4
Adding: &DT_FA997U3R5
Adding: &DT_FA997U3R7
Adding: &DT_FA999U3R3
Adding: &DT_FACONTRL
Adding: &DT_FACONTRL94B
Adding: &WDI_TA1_ACK
Adding: RoutingObjects
Adding: RoutingObjects.zip
Adding: TransformationMaps
Adding: ValidationMaps
Adding: &WDI_E99AENU_VAL
```

Notice the Error. You tried to export the connection info between **MyCompany** and **YourCompany** but you only exported the **MyCompany** details and excluded **YourCompany** in the export file. **Participant Connections** belong to the category of the dependent configuration items. In order for them to be exported, you need to export the other dependencies like the Source and Target Partners, capabilities, interactions. In this case you just included the Source Partner but no the Target partner so the export failed.

Also notice that in the export file, you included the **OtherCompany** partner but you didn't include the **Destination** you created for the **OtherCompany**. There were no errors reported in this case as the Partner definition is independent and partner can exist on its own with out the Destination. This demonstrates the selective export capability of the migration tool.

- \_\_\_\_ 7. You will now use the **export.xml** under <LAB\_FILES>\Migration which includes the **YourCompany** details for export.
- \_\_\_\_ 8. Open a command prompt window and change directories to **C:\IBM\Migration\bcgmigrate\bin**
- \_\_\_\_ 9. Use the following command to import the configuration data. You will need to specify user credentials that have the permission to export the data. **hubadmin** user has the permission by default  
  

```
bcgMigrate.bat -h localhost:58809 -a export -f C:\WPG61Labfiles\Migration\export.xml -u  
hubadmin -p hub1admin -d 5 -r C:\WPG61LabFiles\Migration\output
```
- \_\_\_\_ 10. Check the **C:\WPG61LabFiles\Migration\output** directory for the export zip file and log file **bcgMigration.log**

## What you did in this exercise

In the Lab exercise, you have imported the WebSphere Partner Gateway configuration data in the import file and familiarized yourself on how to selectively export the WebSphere Partner Gateway configuration data.