IOC Enablement
Data integration Lab

Contents

Before You Begin

Introduction

This IOC Example Lab tutorial consists of a combination of presentations and hands-on exercises. By focusing on the end user experience, this tutorial will demonstrate many of the features and functions of the IOC Example.

Documentation Conventions

This document is logically organized to help you understand and use **IOC Example**. Additionally, the following documentation conventions are used to assist in performing each task:

Convention	Explanation
Bold	Words that appear in boldface represent menu options, buttons, icons, or any object you click to cause the software to perform a task. This typeface also represents anything that you must type or enter.
italics	In addition to book titles, italics are used to emphasize certain words, especially new terms when they are first introduced.
Note	This signifies information that emphasizes or supplements important points of the main text.
Important	This signifies information essential to the completion of a task. You can disregard information in a note and still complete a task, but you should not disregard an important note.
Caution	This alerts you to follow a recommended procedure carefully. Failure to do so may result in installation or configuration problems or other preventable conditions.
Tip	This suggests alternative methods that may not be obvious and helps you understand the benefits and capabili-

	ties of a feature or function. A tip is not essential to the basic understanding of the text.
☐	This symbol indicates the end of a note, caution, or tip.

Convention	Explanation
Presentation	The presentation provides conceptual information and background knowledge. Presentations take many forms: formal presentations, instructor lecture, or discussion.
Exercise	These are hands-on exercises used to reinforce the concepts and information covered in a presentation.

Chapter 1

# Data integration

In this lab you will Install and implement a file based data integration customization that imports Stadium entrance gate data and processes it.

What you'll learn in this section:

1. Develop Integrate Gate (CSV) data and publish (CAP) into IOC via Broker

2. Create new Broker Flow

3. Map CSV to CAP elements

4. Install MBroker Flow EAR

5. Run script to feed sample data
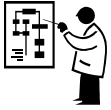
Exercise 1 – Load CSVtoXML Pluggin

The Smart Stadium pack delivers a set of artifacts for use with ABC WebSphere Message Broker to support the acquisition of data from external sources, for example to feed KPIs within IOC.

The first of these artifacts is a MQ Broker "Pattern" that outlines the parsing of a CSV input file, formatting it as a CAP XML message, and passing it to the IOC input queue. This pattern may be used as a basis for the import of any format of CSV file, and the production of CAP message, with simple customization of the mapping specification of the input CSV fields to the desired locations within the target CAP message (for example within the <info> block).

Loading the KPI data integration pattern into the Message Broker Toolkit:

This pattern may be used within the Message Broker Toolkit, simply by extracting the broker pattern (CSVtoXMLPattern.zip) into the broker toolkit plugins folder (usually /opt/ABC/WBMT700/plugins). If the broker toolkit is already running, the toolkit must be restarted. The Message Broker is on the Application and integration Server, Node 3

Step	Action
1	Connect into the Dev server.
2	Go to the plugin directory Cd /opt/ABC/WMBT700/plugins
3	Unzip the CSVtoXMLPattern.zip file from C:\IOC15\labs\data\ to the C:\Program Files\ABC\WMBT800\plugins directory unzip ~/CSVtoXMLPattern.zip *(Note: If the files already exist answer to the prompt with an [A] to replace the files.)*
4	Start Message Broker Toolkit. Click on **Start.** You should see the ABC **WebSphere Message Broker Toolkit.** If you do not see it, open it from the **All Programs** Menu**.**

Summary

In this section you:

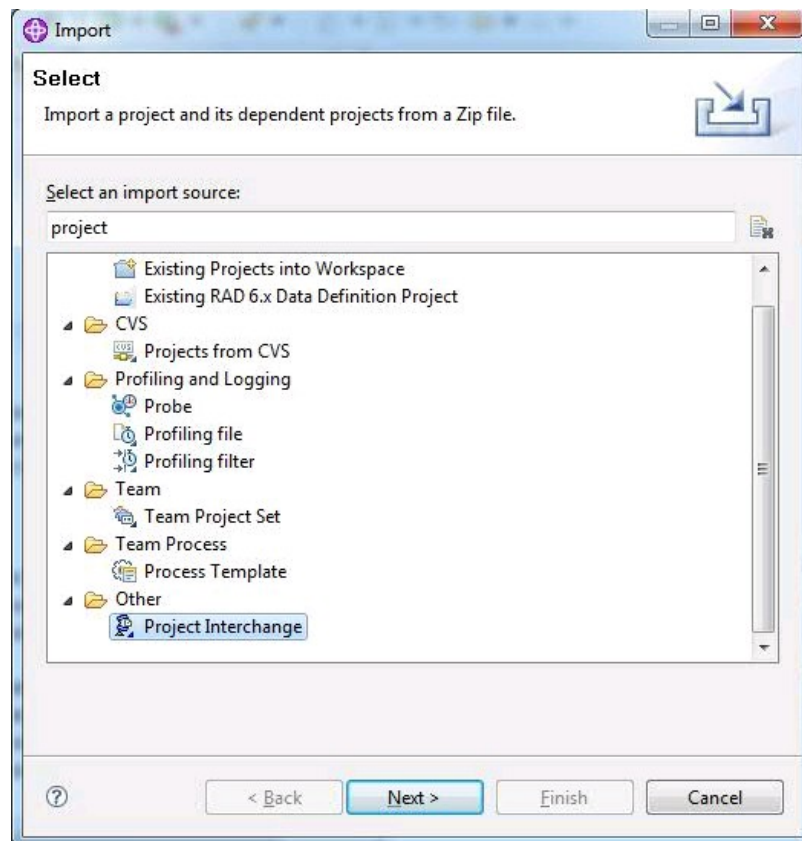- Unzipped the Stadium integration file into the Plugin directory.

-

Exercise 2 – Install Message Broker Flow EAR.

Pre Defined Message Flows:
The Smart Stadium pack also includes two pre-defined message flows based on this pattern; one for import of till sales aggregate data, and a second for the import of aggregated turnstile data, in support of the till revenue and gate flow KPI's respectively.

These flows are included in the "SmarterStadiumBrokerPIF.zip" project interchange file, which may be imported into the Broker Toolkit.

Step	Action
1	In The Message Broker Toolkit, click on **File > Import**.

| 2 | Click on P**roject Interchange** and select **Next**. |
| 3 | Select the drop down box next to From Zip File and choose **SmarterStadium-BrokerPIF.zip**. It should be located in the c:\labs\data\ directory. |

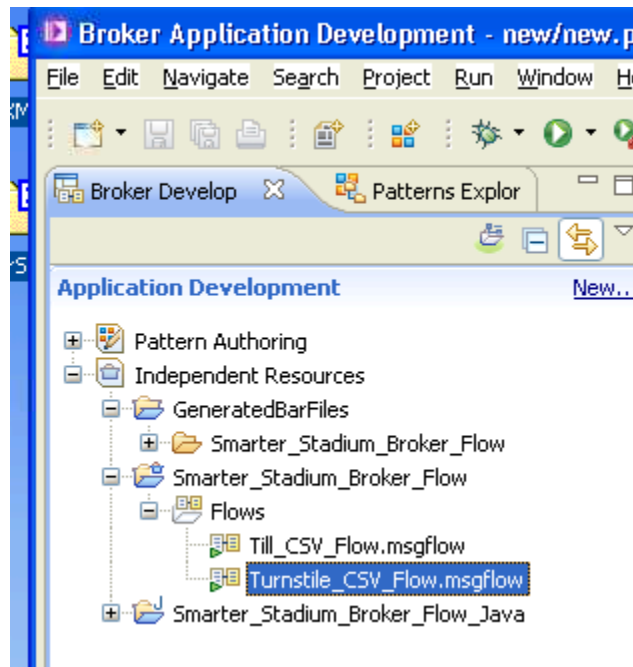| Step	Action
4	Select Both of the options, **Smarter Stadium Broker Flow** and **Smarter Stadium Broker Flow Java**. (Image below)
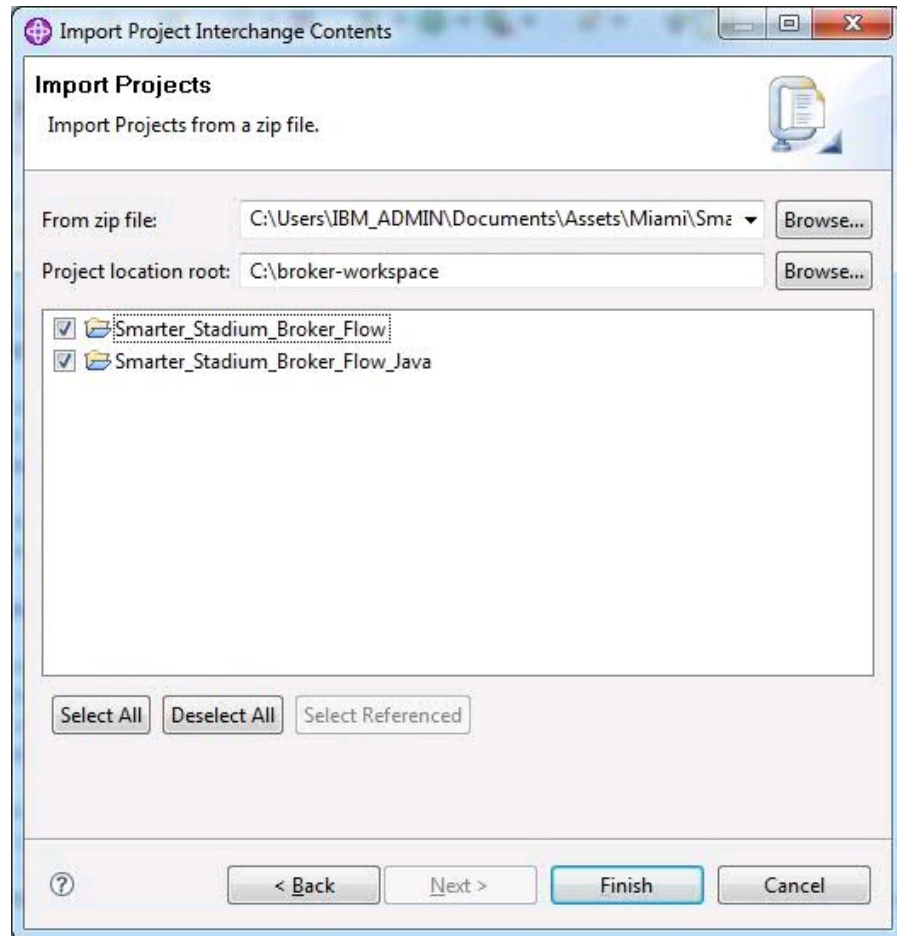
Step	Action

5 Click **Finish**.

Note that this project references a Message Broker runtime, which may need to be altered based on the target deployment. Also, if an object exists, click "yes" to overwrite it.
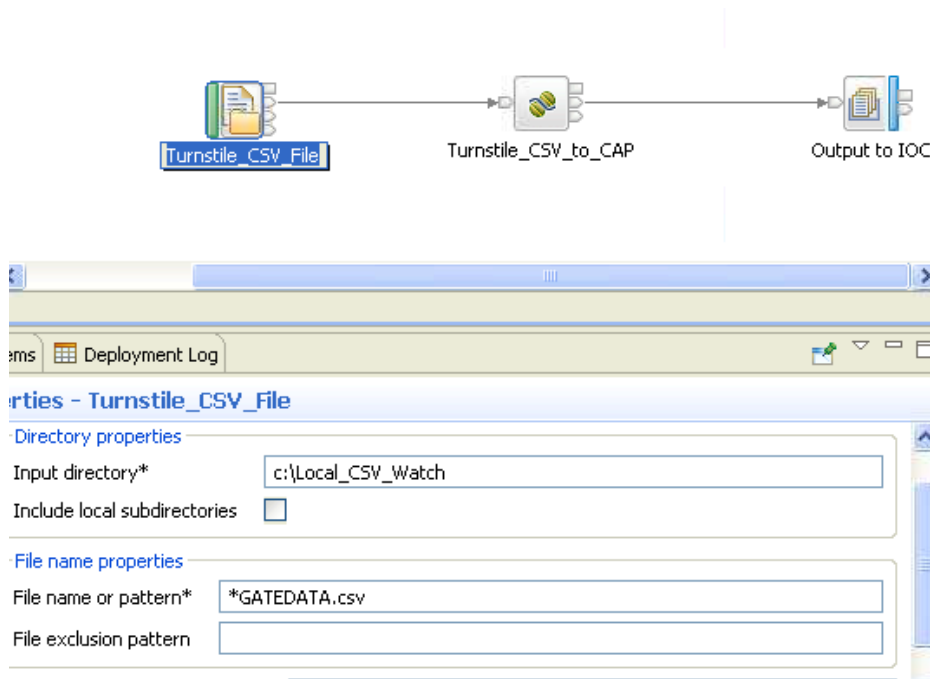
In the next step we will use the imported message flow since it already has the custom code in it.

Double click on the **Turnstile_CSV_Flow_msgflow** and then click on the csv input icon. Make sure the input directory matches the directory of the msgflow that was created earlier... C:\Local_CSV_Watch
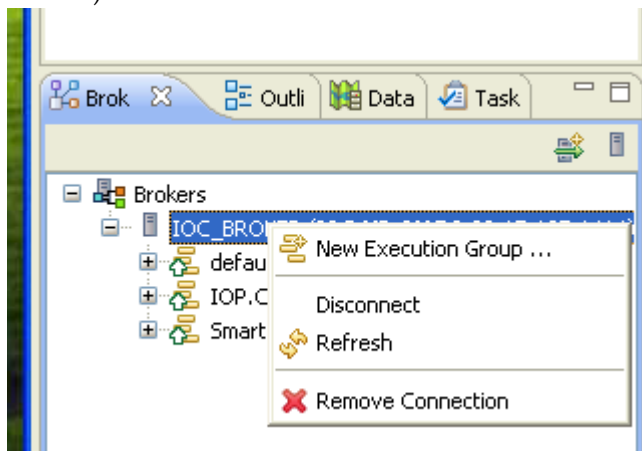
(Images below)

Step	Action

---

6   Right Click on the IOC_BROKER in the bottom left part of the screen and select **New Execution Group** and then provide a name for the Execution Group. (eg. **Smarter Stadium**).

---

7   Right Click on the New Execution group and select **Deploy**.

Step	Action
8	Select the **Turnstile_CSV_Flow_Msgflow** for the execution group and click **Finish**.
9	Right Click the Execution group you just created and select **Deploy**.

Summary

In this section you:

- Created a new Broker Flow

- Created an Execution Group

- Deployed the Broker Flow

Exercise 3 – Create Sample Stadium Data

The Smart Stadium pack includes a data generation utility that will generate sample data to test the solution. The tool allows for creating events like the start of a quarter, half, break or other event. The actual flow data continues into the system but can be reported on based on the events.

Step	Action
1	Copy the utility files to the ioc-event server. *(Note: This step may have been completed, or see lab prep guide)*
2	Modify the configuration file to support the number of event messages to generate. The following steps outline this process. First change directories: cd /root/datageneration
3	The configuration file, *stadiumdata.properties*, has the following parameters: • total messages: this is the number of messages sent per stadium turnstile. • Messagedelay: this is the delay between messages sent. • Turnstileinterval: this is the wait time before sending the next group of messages. Use the **vi** editor or similar editor to modify the configuration file's values. The values of totalmessages=8, messagedelay=1, and turnstileinterval=60000 are recommended.
4	To start the utility issue the following command from the datageneration directory: **/opt/ABC/HTTPServer/java/jre/bin/java -jar GateDataGenerator.jar /root/**
5	The utility will start generating data. Press [CTRL] + [C] to stop the utility.
6	The CSV files can be found in **/root/Local_CSV_Watch**. These files need to be copied to the c:\Local_CSV_Watch directory on the ioc-dev server.
7	**Note:** If the probe has not started the messages will not be in the report and will be in the CAP.OUT.Q instead of in the DB2 database where the following report will check. Instructor will inform you where the messages should be for this class.

Step	Action
	To view the data using the reporting tools open a browser to the following cognos address. **http://ioc-app.ABC.com/cognos/ServletGateway/servlet/Gateway** userid/password: **wpsadmin/aust1ni0c**
8	Click on **ioc_model > reports > CAP events by type status and date**.
9	Change the end date to tomorrows date as the dates will use 12:01am for the time.
10	Click **View the Report**.

Summary

In this section you:

- Installed / Configured Data Generator Utility
- Generated sample data
- Viewed the generated data in a Cognos Report