

IBM Integration Bus

IBM Integration Bus on Cloud Connecting to local MQ and DB2 Systems using Endpoint Connectors

Featuring:

- REST API
- Using a REST API Description in a Shared Library
- Creating a generic map using JSON Models
- MQ and DB2 Endpoint Connectivity
- Using IIB on Could with Secure Connector Agents
- Testing using IIB Flow Exerciser and Swagger UI

September 2016 Hands-on lab built at product Version 10.0.0.6

1. INTE	RODUCTION	3
1.1	SCENARIO OVERVIEW	3
2. INIT	IAL ENVIRONMENT PREPARATION	5
2.1 2.2 2.3 2.4	OPEN THE WINDOWS LOG MONITOR FOR IIB CONFIGURE TESTNODE_IIBUSER FOR REST APIS CONFIGURE INTEGRATION BUS NODE TO WORK WITH DB2 CREATE REQUEST/RESPONSE QUEUES	
3. REV	IEW HR_SERVICE SOLUTION	9
3.1 3.2	REVIEW /EMPLOYEES/{EMPLOYEENUMBER}/MQENDPOINT REVIEW /EMPLOYEES/{EMPLOYEENUMBER}/CLOUDODBC	
4. CRE	ATE HR_SERVICE_EXECUTABLES SHARED LIBRARY	15
4.1 4.2	IMPORT THE REST API DESCRIPTION CREATE GETEMPLOYEEJSON MAP	
5. CRE	ATE HR_SERVICE_MQPROVIDER APPLICATION	27
5.1 5.2 5.3	CREATE THE APPLICATION TO RESPOND TO HR_SERVICE CONFIGURE THE MAPPING NODE TEST (LOCALLY) HR_SERVICE_MQPROVIDER	
6. RUN	I HR_SERVICE LOCALLY	
6.1 6.2 7 RUN	PREPARE AND DEPLOY A BAR FILE TEST MQ SCENARIO USING SWAGGER UI	
7.1		40
7.1		
73	DEPLOY HR SERVICE TO UR ON CLOUD	43 43
7.4	CONFIGURE IIB ON CLOUD TO CONNECT TO LOCAL MQ AND DB2	
7.4.	1 Define Endpoints	
7.4.2	2 Determine IIB Secure Connector configuration	
7.4.3	3 Option1: Set up a NEW IIB Secure Connector	51
7.4.4	4 Option2: Set up a previously configured IIB Secure Connector	
7.4.3	5 Confirm MQ and DB2 Endpoint configuration	
8. TES	T HR_SERVICE RUNNING ON IIB ON CLOUD	55
8.1	START YOUR IIB ON CLOUD INTEGRATION	
8.2	TEST THE MQENDPOINT INTEGRATION	
8.3	TEST THE DB2 (ODBC) ENDPOINT	
APPEND	Х	59
	LAB GUIDE	

1. Introduction

1.1 Scenario Overview

In this lab guide you will develop a scenario to show how a REST API (created using IBM Integration Bus) can run on IIB on Cloud and access local (on premise) MQ and DB2 systems.

Whilst developing this scenario the lab guide will demonstrate how json models created as part of process of defining the REST API can also be re-used by mapping nodes defined in a Shared Library. When a REST API description is imported into a Shared Library, any graphical data mapping node also created and stored within the Shared Library has access to the json models defined within the imported REST API Description. This enables applications to easily process data produced from (or intended for) a REST API.

1. The MQ Scenario:



You will import and review a REST API solution "HR_Service" into your workspace. The MQ scenario has already been implemented using a resource **/employees/{employeeNumber}//MQEndpoint**. After reviewing this implemented solution, you will then import the REST API Description used by this REST API into a Shared Library and create a map that an MQ Application (created as an IIB Application) can use to process data (requests) from and send data (responses) to the REST API. Local MQ queues will be used to pass requests and responses between the HR_Service REST API and the MQ application. Once working correctly on your local system, the HR_Service REST API will

be deployed to IIB on Cloud where you will configure MQ Endpoints and a Secure connector to enable IIB on Cloud to communicate with your local MQ system.

2. The DB2 (ODBC) scenario:



You will review the implementation of an additional resource in HR_Service,

/employees/{employeeNumber}/cloudODBC. The implementation uses ESQL in a Compute node to access your local DB2 HRDB database directly from IIB on Cloud. A DB2 Endpoint is configured on IIB on Cloud to enable this resource to access the HRDB DB2 database using the same Secure Connector configured for the MQ scenario.

2. Initial Environment Preparation

Important note

This lab will require an 'IIB on Cloud' account. For the lab you can use the Free Trial version, which you can request at *http://www-03.ibm.com/software/products/en/ibm-integration-bus-on-cloud*.

You should use the Windows user "iibuser". This user is a member of mqbrkrs and mqm, but is not a member of Administrators. The user "iibuser" can create new IIB nodes and do all required IIB development work. However, installation of the IIB product requires Administrator privileges (not required in this lab).

2.1 Open the Windows Log Monitor for IIB

A useful tool for IIB development on Windows is the IIB Log Viewer. This tool continuously monitors the Windows Event Log, and all messages from the log are displayed immediately.

From the Start menu, click IIB Event Log Monitor. The Monitor will open; it is useful to have this always open in the background.

💶 IIB Event Log Monitor
<code>BIP3132I: < IB10NODE.server1</code>) The HITP Listener has started listening on port ' \square
'7800'' for ''http'' connections. [10/3/2014 3:17:23 PM]
BIP2154I: (IB10NODE.server1) Execution group finished with Configuration messa
ge. L10/3/2014 3:17:24 PMJ
BIP21521: (IB10NODE.server1) Configuration message received from broker. [10/3
/2014 5:07:36 PM]
BIP2153I: (IB10NODE.server1) About to ''Change'' an execution group. [10/3/201
4 5:07:36 PM]
BIP2155I: < IB10NODE.server1 > About to ''create '' the deployed resource ''Empl
oyeeService_JSONClient'' of type ''.APPZIP''. [10/3/2014 5:07:37 PM]
BIP2155I: < IB10NODE.server1 > About to ''create '' the deployed resource ''gen.
getEmployee_EmployeeService_EmpServClient_JSON1'' of type ⁷ '.SUBFLOW''. [10/3/20]
14 5:07:37 PM]
BIP2155I: < IB10NODE.server1 > About to ''create '' the deployed resource ''EmpS
ervClient_JSON1'' of type ''.MSGFLOW''. [10/3/2014 5:07:37 PM]
BIP2154I: < IB10NODE.server1 > Execution group finished with Configuration messa
ge. [10/3/2014 5:07:43 PM]
BIP3132I: < IB10NODE.HTTPListener > The HTTP Listener has started listening on p
ort ''7080'' for ''http'' connections. [10/3/2014 5:07:47 PM]
BIP21521: (IB10NODE.server1) Configuration message received from broker. [10/3
/2014 5:50:41 PM]
BIP21531: (IB10NODE.server1) About to ''Change'' an execution group. [10/3/201
4 5:50:41 PM]
BIP2155I: < IB10NODE.server1 > About to ''delete '' the deployed resource ''EmpS

This tool is not shipped as part of the IIB product; please contact us directly if you would like a copy.

2.2 Configure TESTNODE_iibuser for REST APIs

If you have already done a previous lab involving the REST APIs in this series of lab guides, you can skip to the next heading.

The IIB support for the REST API requires some special configuration for the IIB node and server.

_1.	Login to Windows as the user "iibuser", password = "passw0rd". (You may already be logged in).
	Start the IIB Toolkit from the Start menu.
_2.	Ensure that TESTNODE_iibuser is started.
_3.	Enable Cross-Origin Resource Scripting for REST. This is required when testing with the SwaggerUI test tool. See http://www.w3.org/TR/cors/?cm_mc_uid=09173639950214518562833&cm_mc_sid_5020000=1 452177651 for further information. (Helpful hint - the VM keyboard is set to UK English. If you cannot find the "\" with your keyboard, use "cd" to move the a higher-level folder in a DOS window), or change the keyboard settings to reflect your locale.)
	In an IIB Command Console (shortcut on the Start menu), run the command: mqsichangeproperties TESTNODE_iibuser -e default -o HTTPConnector -n corsEnabled -v true
_4.	Restart the IIB node

2.3 Configure Integration Bus node to work with DB2

If you have already done a previous lab involving the HRDB database in this series of lab guides, you can skip to the next heading.

To run this lab, the Integration Bus node must be enabled to allow a JDBC connection to the HRDB database.

1. Open an IIB Command Console (from the Start menu), and navigate to

```
c:\student10\Create HR database
```

2. Run the command

3_Create_JDBC_for_HRDB

Accept the defaults presented in the script. This will create the required JDBC configurable service for the HRDB database.

3. Run the command

```
4_Create_HRDB_SecurityID
```

4. Stop and restart the node to enable the above definitions to be activated

mqsistop TESTNODE_iibuser

mqsistart TESTNODE_iibuser

This will create the necessary security credentials enabling TESTNODE_iibuser to connect to the database.

Recreating the HRDB database and tables

The HRDB database, and the EMPLOYEE and DEPARTMENT tables have already been created on the supplied VMWare image. If you wish to recreate your own instance of this database, the command **1_Create_HRDB_database.cmd** and

2_Create_HRDB_Tables.cmd are provided for this. If used in conjunction with the VM image, these commands must be run under the user "iibadmin". Appropriate database permissions are included in the scripts to GRANT access to the user iibuser.

2.4 Create Request/Response Queues

For this lab you need to create two MQ queues MQREQUEST_CLOUD and MQRESPONSE_CLOUD which are used in the MQ Scenario. In this next section you will run a script to create these queues on IB10GMGR (note: *This lab guide will assume that IB10QMGR is the default MQ Queue Manager, you may need to customize this value throughout this lab guide if you are running this in your own environment*).

In an Integration Console navigate to:
$C:\student10\IIBoC\EndpointConnectors\install$
Enter the command:
DefineQs.cmd
Respond to the prompts (accept the default IB10QMGR). Ensure both queues are created successfully:
<pre>C:\student10\IIBoC\EndpointConnectors\install>defineqs Enter Queue manager name (default is IB10QMGR): The queues: MQREQUEST_CLOUD and MQRESPONSE_CLOUD on "IB10QMGR" will be recreated,</pre>
<pre>AMQ8006: WebSphere MQ queue created. 2 : define ql(MQREQUEST_CLOUD) replace AMQ8006: WebSphere MQ queue created. 2 MQSC commands read. No commands have a syntax error. All valid MQSC commands were processed.</pre>

3. Review HR_Service Solution

In this section you will review the REST API implementation in HR_Service for the MQ and DB2 scenarios outlined in the introduction.

1.	In the Integratio Import.	n Toolkit right click on the Application Development background and select
	Navigate to c:\ file:	student10\IIBoC\EndpointConnectors\Resources\ and import the PI
		HR_Service_EndpointConnectors.zip
) Import Project	Interchange Contents
	Import Projects Import Projects from	m a zip file.
	From zip file:	C:\student10\IIBoC\EndpointConnectors\Resources\HR_Service_EndpointConnectors.zp Browse
	Project location root	:: C:\Users\iibuser\IBM\IIBT10\Workspace_EndpointConnectors Browse
	 ✓ ☺ HRDB ✓ ☺ HRDB_project ✓ ☺ HR_Service 	
	Select All Desele	ct All Select Referenced
	?	< Back Next > Finish Cancel
2.	Double click on API:	the REST API Description to see the resources that are defined in this REST
		Application Development [⋈] Pa
		Application Development
		□ · · · □ HR_Service
		REST API Description
		the sources the s
		🗄 🗁 Subflows

3.1 Review /employees/{employeeNumber}/MQEndpoint

In this next section you will review the Resource that will demonstrate the ability for Resources running on IIB on Cloud to connect to an MQ queue defined locally (on premise) using an MQ Endpoint Policy.

1.	Scroll down the list of Resources and expand /employees/{employeeNumber}/MQEndpoint:
	I HR_Service ∞
	→ Header
	- Resources
	/departments/{departmentKey}
	/departments/{departmentKey}/employees
	/departments/{departmentKey}/manager
	/employees/{EDLEVEL}/predictSalary
	/employees/{employeeNumber}
	/employees/{employeeNumber}/MQEndpoint
	/employees/{employeeNumber}/cloudODBC
	/employees/{employeeNumber}/department
	and the ball to be the more concerned
۷.	path of the resource (so will be passed in the LocalEnvironment tree to sub flows that implement this Resource):
	GET getMQEndpoint
	Name Parameter typData type Format Required Descrip
	employeeNumberbath string
	Response stat Description
	200 The operation was successful.
3.	Click the following icon to open the sub flow that implements this resource:
	Open the subflow for the operation

	Image: HR_Service → getMQEndpoint.subflow → Image: HR_Service_IIBonC.bar
	N Flow Evercicer: 🔍 🖼 🔟 🔍 🤤
	(a) Input getEmployeeNumber (b) GetEmployeeNumber (b) Remove MQ Header Retrieve Response Submit MQ Request to Provider
	(a) A map is executed that takes the REST API input Parameter "employeeNumber" and puts the value in the message tree as a JSON object, for example the data output from this map could be "{ "employeeNumber": "000010"}
	(b) Output from the map is written to an MQ queue – this forms the request input for the MQ Application that will obtain the data from the database.
	(c) The Response from the MQ Application is obtained via an MQ Get node (the format of the response message is in JSON (of type EmployeeResponse – as defined in the REST API Definition for HR_Service).
	(d) MQ Headers are removed from the message payload, NB at the time of writing this lab guide MQ Header nodes were not supported on IIB on Cloud so a small piece of ESQL node was written to remove the MQMD from the message.
5. C ta co	Click on the MQ Output node "Submit MQ Request to Provider" and click the MQ Connection ab (in properties). Notice the connection is configured using an MQ Client connection (no onnection details are specified.

6.	Click the Policy tab and same policy configured) exist locally (on-premise were created locally (ie this policy on IIB on Clo	note that the node has a Policy configured (<i>the MQ Get node has the</i> . To run this REST API on IIB on Cloud this policy does not need to e), however the policy URL does need to have the same format as if it the name prefixed by " /apiv1/policy/MQEndpoint ". You will configure ud later in this lab guide:
	HR_Service	*getMQEndpoint.subflow 🛛 🖪 HR_Service_IIBonC.bar
	Flow Exercise	er: 🖻 🖩 🖄 🗍 🔍 🔍
	getEmployeeN Submi	Number Remove MQ Header Retrieve Response t MQ Request to Provider
	Properties 🛛 🕄 F	Problems 🗄 Outline 🖉 Tasks 🎟 Deployment Log
	MQ Output	Node Properties - Submit MQ Request to Provider
	Description Basic	Use a policy to control the operational behavior of the node at run time, the policy override any properties that are set on the MQ Connection tak
	Advanced	Policy URL /apiv1/policy/MQEndpoint/CLOUDPOLICY
	Request	
	Validation	
	Policy	
	Monitoring	

3.2 Review /employees/{employeeNumber}/cloudODBC

In this next section you will review the Resource that will demonstrate the ability for Resources running on IIB on Cloud to connect to a DB2 database running locally (on premise) using ODBC.

1.	In the HR_Service REST API Description (In the Integration Toolkit) locate and open /employees/{employeeNumber}/cloudODBC, (note that the GET for this resource has the same employeeNumber parameter):
	/ employees/{employeeNumber}/MQEndpoint
	/employees/{employeeNumber}/cloudODBC
	GET getCloudODBC
	Name Parameter typData type Format Required De
	employeeNumberbath string v
	Response stat Description
	200 The operation was successful.
	/employees/{employeeNumber}/department
2.	". Open the subflow that implements this operation
3.	The resource is implemented using an ESQL Compute node. Double click on getEmployeeODBC to open the ESQL that implements the operation:
	Flow Exerciser: Flow Exerciser: Input GetEmployeeODBC

🖻 getCloudODBC_getEmployeeODBC.esql 🕮 🔚 getCloudODBC.subflow 🖉 🖽 Service
CREATE COMPUTE MODULE getCloudODBC_getEmployeeODBC CREATE FUNCTION Main() RETURNS BOOLEAN BEGIN
SET OutputRoot.JSON.Data.EmployeeResponse = NULL;
<pre>SET OutputRoot.JSON.Data.DBResp.UserReturnCode = 0; SET OutputRoot.JSON.Data.DBResp.RowsRetrieved = 0;</pre>
CREATE FIELD OutputRoot.JSON.Data; CREATE FIELD OutputRoot.JSON.Data.Employee IDENTITY (JSON.Array)Employee;
 populate Employee on the output message tree with employee info from the database Use of Item ensures that the Name-value pairs are included - if you omit Item, <onl< li=""> item values (without their corresponding names) are written to the message tree. </onl<>
SET OutputRoot.JSON.Data.Employee.Item = THE(SELECT T.* FROM Database.HRDB.IIBADMIN.EMPLOYEE AS T
if Data has been retrieved set the value of DBResp.RowsRetrieved to 1 (else set to
<pre>IF EXISTS(OutputRoot.JSON.Data.Employee.Item[]) THEN SET OutputRoot.JSON.Data.DBResp.RowsRetrieved = 1;</pre>
SET OutputRoot.JSON.Data.DBResp.RowsRetrieved = 0;
END IF;
RETURN TRUE;
Source

4. Create HR_Service_Executables Shared Library

The map that you will create as part of this lab guide will be stored in a Shared Library ready for re-use by other applications in this series of lab guides. In this next section you will create this Shared Library.

1. In the Integration Toolkit, create a new Shared Library. Right click on the Application Development window and select New > Library:	
Call the Shared Library HR_Service_Executables, Click Next:	
🛞 New Library	
Create a new library	
Create a library to group reusable resources so that they can be shared and managed together. Enter a name for the new library.	
Library name HR_Service_Executables	
Type Shared Library	
O Static Library	
Select Shared Library when you require one copy of the library to be deployed and referenced by one or more applications. When a shared library is undated, changes are available immediately to all applications or shared	
libraries that refer to it.	
Image: Second	
2. Select HRDB from the list of existing Libraries and click Finish:	
New Library	
Create a new library	
Specify dependencies on existing libraries.	
Select existing libraries to reference from the new library:	
₩ HRDB	
Image: Second	
Note: this is required in order for the map to have access to the HRDB database definition (it stored in the HRDB Shared Library).	is

4.1 Import the REST API description

The map that you will create as part of this lab guide will require access to this JSON models that were created as part of the Lab guide that created HR_Service (Lab 01 and Lab 02). In this next section you will import the swagger file associated with HR_Service into the Shared Library. The result of doing this will mean that the map (that will also be created in the Shared Library) will have access to these JSON models.

1.	Right click on the Shared Library HR_Service_Executables and select Import:
	Application Development 🛛 🥵 Patterns Explorer
	Application Development New
	HR_Service_Executables
	(New) Manage Library references
	e-Co Other Resources Procus on Library
	HRDB Export XSDs from Library
	el- [™] Other Resources
	A HRDB.dbm
	BARs
	GeneratedBarFiles Paste
	E
	Rename
	월 Integra ≅ 월 Integra
	Export
	L
2.	In the input source window, Select REST APIs > REST API Definition, then click next :
	Select
	Import a REST API definition into the REST API Catalog.
	Soloct an import cource:
	type filter text
	Pre> Modeling
	REST APIS
	Run/Dobug
	e leam e leam ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

3.	Click Select a Swagger Document from the workspace, then click next:
	Import a REST API definition
	Import a REST API definition
	Import a REST API definition into the REST API Catalog.
	Import a REST API definition into the REST API Catalog for use with a REST request node by importing a Swagger document into the selected project that describes the REST API you wish to invoke. Select one of the following options to locate a Swagger document:
	Project: HR_Service_Executables
	Select a Swagger document from the workspace or from a file system
	C Retrieve a Swagger document from a URL using HTTP or HTTPS
	< Back
4.	Choose to "select from the all projects in the workspace" and click Finish:
	If the Swagger document is not in a referenced project, then that Swagger document will be imported into the current project.
	Import a Swagger 2.0 document from one of the following locations:
	C Select from a file system
	Location: Browse
	Select from all projects in the workspace
	□-□ HR_Service □-□ Resources □-□ Other Resources □-□ HR_Employee_and_Department_Services.json
	? < Back



4.2 Create getEmployeeJSON map

You will now create the getEmployeeJSON map and store it in the HR_Service_Executables Shared Library. The Map will process input from a JSON message with the following format:

```
{"employeeNumber":"nnnnn"}
```

Where "nnnnn" is the key of the database user to find.

The output of the map will be in the format of the json object EmployeeResponse (defined as a json object in the REST API description – the map will have access to the definition of the json object because the swagger file has been imported into the Shared Library.

∣ ₩a Ar	oplication Development ¤	R Patterns Explorer □□	Getting Started - IBM Integration
App	lication Development	New	
÷	HR Service Executables	New	■ Subflow
	API Catalog	Manage Library references	🖹 Message Model
	HR Employee and [Manage included projects	🕒 Message Map
E	Referenced Libraries	a nanage mended projecto	ESQL File
		😹 Focus on Library	🛱 Broker Schema
	Generation Other Resources	Convert to Application	Independent Resources
	🗄 📾 HRDB_project	Evport VSDc from Library	
Ē	• Ther Resources	Export ASDS from Elbrary	📑 Example
		Migrate	=° Othor
- hone		~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	more with the second and the second
	New Message Map Specify a new mess Colort map have control	sage map file	
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message map 	sage map file iner, name, and broker schema for want to create:	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message map Submen called by an annual set of the se	sage map file iner, name, and broker schema for want to create: ap called by a message flow node	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message mat Submap called by an 	sage map file iner, name, and broker schema fo want to create: ap called by a message flow node nother map	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message main Submap called by an Container: HR_Service 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message main Submap called by an Container: HR_Service 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message mail Submap called by an Container: HR_Service Map name: getEmployee 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you v Simple message mail Submap called by an Container: HR_Service Map name: getEmployee Map organization 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables eeJSON	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message main Submap called by an Container: HR_Service Map name: getEmployee Map organization Vse default broker set 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables eeJSON	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message main Submap called by an Container: HR_Service Map name: getEmployee Map organization Use default broker set Schema: (default broker 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables eeJSON	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you w Simple message main Submap called by an Container: HR_Service Map name: getEmployee Map organization Vse default broker set Schema: (default broker 	sage map file iner, name, and broker schema fe want to create: ap called by a message flow node nother map _Executables eeJSON schema ser schema)	e New
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message mail Submap called by an Container: HR_Service Map name: getEmployee Map organization Use default broker set Schema: (default broker 	sage map file iner, name, and broker schema fe want to create: ap called by a message flow node nother map _Executables eeJSON schema ter schema)	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message mail Submap called by an Container: HR_Service Map name: getEmployee Map organization Use default broker essist Schema: (default broker) 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables eeJSON schema ter schema)	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you vont Simple message main Submap called by an Container: HR_Service Map name: getEmployee Map organization Use default broker sets Schema: (default broker) 	sage map file iner, name, and broker schema for want to create: ap called by a message flow node nother map _Executables eeJSON schema ter schema)	or the new map.
	 New Message Map Specify a new mess Select map type, contain Type of map that you voil Simple message mail Submap called by an Container: HR_Service Map name: getEmployee Map organization Use default broker set Schema: (default broker 	sage map file iner, name, and broker schema fe want to create: ap called by a message flow node nother map _Executables eeJSON schema ter schema)	or the new map.

3.	On the "Select Map inputs", expand IBM supplied message models and select JSON Object Model.
	On " Select map outputs ", expand HR_SERVICE_Executables > JSON Types and select EmployeeResponse, (note these JSON message models appear here because earlier you imported the REST API Definition file into the Shared Library which describes the HR_Service). Click Finish.
	New Message Map
	Select map inputs and outputs
	Creates a map that can contain message inputs and outputs with the Properties folder. Optionally, database operations, message headers, and Loca Environment can be added to the map after creation.
	Filter map input names (? = any character, * = any Filter map output names (? = any character, * = any String):
	Select man inputs Select man outputs
	Image: Select map duputs Image: Select map duputs Image: Hr_Service_Executables Image: Hr_Service_Executables Image: I
	BLOB (BLOB domain nessage model sur SOAP_Domain_Msg (SOAP domain mess SOAP_Domain_Msg (SOAP domain mess SOAP_DOMAIN_SOAP domain mess SOAP domain mess SOAP_DOMAIN_SOAP domain mess
	Son (JSON object message model supple
	← ← ← EmployeeResponse - HR-Employee-and-Departme
	Physical Location:
	Library: * HR_Service_Executables Path: HR-Employee-and-Department-Services.json Namespace: # (http://www.ihm.com/iih/msl/ison)
	? < Back

getEmployeeJSON					
-getEmployeeJSON	🎶 🦊 🗯 🐊	🛛 🗶 🛛 🐗 rat	🖙 🛱 🔞 🗞 🎝	🎟 🏪 🛱	8
□ ➡ Message Assembly ➡ <click filter="" to=""></click>	y JSC	ON			□ ➡ Message A ➡ <click filt<="" p="" to=""></click>
🗉 📌 Properties	[01] Pro	pertiesType		love -	🔹 🐙 Dronertie
🗉 🞜 JSON	[11] JSC) NMsgType	Undo		
Padding	[01] stri	ng	Revert		
□ ♣ choice of cast it	ems [11]	-			
🖳 Data	[11] any	Туре	Cut	(Ctrl+X
🗉 🖻 Data	[11] JSC	NObject	Daste		Ctrl+C Ctrl+V
ny 🐔 📶	[0*]		- Lu		50111
		-	Delete		
		4	Add output Add environmen	(t mapping	Ctrl+Shift+N, O
		-	Open Informatio	n Popup	Ctrl+Shift+I
			Open Declaratio	n l	F3
			🛛 Cast		
			🗳 Add User-Define	ed (Ctrl+Shift+C
		4	 Add Transform Add Connection Auto Map Ordel inlete Ordel 	(Ctrl+J
Rename the element	to employeeN	Number:			
-getEm	ployeeJSON	/ P1 P 🎘		₽ ₩ ₩ ¥) U) III }
	sage assembly		13010		
		[0 1]	DropertiesTupe		Move
	opercies	[U1]		_	
	NUN Dadding	[11]	string		
		[U1]	suing		}
	Data	ms [11]	2PV/Tv/Po		
-		[11]	any rype	_	
	Ddld	[11]	JSONODJECT		
		H [0 ¥]			4
	choice of cast	it [0*]			5

6.	Click Select Rows from a Database:					
7.	 In the "New Database Select" window, a) Select EMPLOYEE in "Choose the columns to include" b) Delete the 1=1 in the SQL where clause and leave your cursor in this box. c) Double click on the IIBADMIN>EMPLOYEE>EMPNO (leave the cursor in the box once more) d) Double click on the equals sign "=" under the list of operators (this will add an equal sign to the "SQL where clause" box. e) In "available inputs for column values" expand JSON until you find employeeNumber (this should be the last element). f) Double click on employeeNumber, this will add a question mark "?" into the where clause. The meaning of this question mark appears in the XPATH expression – this is the value of employeeNumber passed into the map. g) Click OK when complete. 					
	 We valabase values is used to extract only those rows that fulfill a specified condition, which is often to value of a key column in the database table. The value can come from other inputs in the map. The where clause is used to extract only those rows that fulfill a specified condition, which is often the value of a key column in the database table. The value can come from other inputs in the map. The wales can a come from other inputs in the map. The wales of a key column in the database table. The value can come from other inputs in the map. The expression must evaluate to a boolean. Choose the columns to include You must choose at least one column. Image: MPLOYEE <p< th=""></p<>					
	OK Cancel					

-getEmployeeJSON	목 X 에 데 ⑥ ⑥ 역	🏷 🖏 🐌 🎟 🏭 📾 🏭		
Bessage Assembly Click to filter>	JSON		□ ➡ Message Assemble ➡ <click filter="" to=""></click>	y JSON
🗉 🖈 Properties	[01] PropertiesType	Move -	🗉 🕖 Properties	[01] PropertiesType
⊟ 🖉 JSON	[11] JSONMsgType		🗉 🞜 JSON	[11] JSONMsgType
Padding	[01] string		Padding	[01] string
🗆 🖧 choice of cast items	[11]		🗉 🕼 Data	[11] EmployeeResponse
📲 Data	[11] anyType		DBResp	[01] DBRESP
🗉 🖥 Data	[11] JSONObject		E Employee	[01] JSONArray_Employe
ା କି choice of cast items	[0*]	C	u ∎ Item	
器 any	[11]			
employeeivumber	[11] string			
Select from HRDB				
Click to filter>			J	
🗉 🕫 ResultSet	[0*] Result Set Row	\ <u>≣ Select</u> +		
Click the Select (text	to enter the nes	sted map that wil	I process the da	ata passed back fro
the Select statemen	t:	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		E Item



14.	In the Custom XP	ATH properties	type fn:count()				
	Place your cursor between the two brackets and type R then press Ctrl and the Space bar. Select the ResultSet pertinent to your environment – in the example here \$ResultSet1 :							
	for many many many many many many many many							
	Select from HRDB							
	Section 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	>			8			
	}				Select -			
	🗉 🗟 ResultSet		[0*] Result Set Ro	W SC	ustom XPath 🗸			
	}			_ /	1			
	{•							
	} ₽ Properties ¤ 🔝	Problems 🗄 O	outline 🖉 Tasks 🔳	Deployment Log	}			
	Transform - (Custom XPa	ath	1 9 5	5			
	1		Incort Simple	/Dath				
	General	fn:count(R	w \$ResultSet1	Maul				
	Namespaces		y fn:abs (numer	ic?): numeric?				
	Cardinality		× fn:adjust-date	Time-to-timezone (x	s:dateTime?			
	~	'	v fn:adjust-date	Time-to-timezone (x	s:dateTime?			
	h	<u> </u>	-tron		manninda			
15.	The map will look	like this when a	complete:					
			•					
	GetEmployeeJSON			ram1				
	•getEmployeeJSON	100N	[유] 'OF 'OF 'OF 표 258 259 259 2 	S-	1001			
	Message Assembly * <click filter="" to=""></click>	JSON		Click to filter>	JSON			
	🗉 률 Properties	[01] PropertiesType	Move -	Properties	[01] PropertiesType			
	■ # JSON	[11] JSONMsgType		Padding	[11] JSONMISGType [01] string			
	■ Adding	[11]		🗉 🕼 Data	[11] EmployeeResponse			
	🖁 Data	[11] anyType		DBResp	[01] DBRESP			
	🗆 🕼 Data	[11] JSONObject	🛍 Assign 👻	UserReturnCode	[01] <integer></integer>			
	🗉 🖧 choice of cast items	[0*]		RowsRetrieved	[01] <integer></integer>			
	any .	[11]		RowsAdded	[0.1] <integer></integer>			
	Contraction of the second seco	[11] string			[0.1] <integer></integer>			
	Select from HRDB			SOLCODE Errorcode	[01] <integer></integer>			
	Click to filter>		The last of the la	SQLSTATE_SQLState	e [01] <string></string>			
	🗉 🕫 ResultSet	[0*] Result Set Row	© <u>Select</u>	SQL_Error_Message	[01] <string></string>			
				Employee	[01] JSONArray_Employee			
				∎ ∎ Item	[0*] EMPLOYEE			
	hannennen		-low-manana-antic	manter and marked				
16.	Save (Ctrl S) and	close the map.	Check to ensure the	ne map has been sa	aved in the			
	HR_Service_Exec	utables Shared	d Library:					
		HR Servi	ce Executables	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	1			
		🖻 🕾 Mans			1			
			mployee ISON man		ş			
					3			
		KEST /			Į			
		}	mployee and Departm	ent Services 3.0.0	{			
		🖻 🗟 Refere	nced Libraries		}			
		👌 🗄 💒 HRD	ЪВ		{			
		🗄 🖻 🕆 Other I	Resources					
		1	Employee-and-Departn	nent-Services.json				
			~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	3			

# 5. Create HR_Service_MQProvider Application

This application will be used as a provider of details from the EMPLOYEE table. The implementation (subflow in HR_Service) that is associated with Resource

**/employees/{employeeNumber}//MQEndpoint** will write request data (an MQ message) to MQREQUEST_CLOUD (an MQ queue which exists locally). The request data will be processed by this application and its response will be written to MQRESPONSE_CLOUD (also a local queue). The HR_Service REST API will obtain this response data from MQRESPONSE_CLOUD and provide the details to the requestor.

## 5.1 Create the Application to respond to HR_Service



2.	Right-click on the Applica References window, sele	ation, select Manage Libra ect HR_Service_Executab	ary References. In the Manage Library oles then click OK:	
		The following static libraries will also are referenced by the selected static l	be included because they libraries.	
3.	In the Integration Toolkit Message Flow.	, right click on HR_Service	ee_MQProvider and select New > Getting Started - IBM Integration Toolkit Qui Start Message Flow Message Model Message Map BESQL File	
4.	Call the message flow g	etEmployeeMQ		

5.	When the message flow ed	itor opens, add the following:						
	a) An MQ Input node Configure the queu Configure the (loca	(call it Get Request). The name field ("Basic" tab) to be: MQREQUEST_CLOUD. I) queue manager name (" <i>MQ Connection"</i> tab) to be: IB10QMGR						
	b) An MQ Output node (call it Write Response) Configure the queue name field to be: MQRESPONSE_CLOUD. Configure the (local) queue manager name to be: IB10QMGR							
	Image: Employee MQ      Image: Palette      I	rmsgflow ¤ Flow Exerciser: ■ ■ 1   • • • Get Request Write Response						
6.	The data being passed to the node needs to be configure.	me MQGet node (Get Request) is in JSON format so the MQGet ad to handle JSON input. main on the <i>"Input Message Parsing" tab</i> ) to: JSON						
	Properties      Problem	s 🗄 Outline 🖉 Tasks 🖩 Deployment Log						
	MQ Input Node P	roperties - Get Request						
	Input Message Parsing							
	Parser Options	Message domain JSON : For JavaScript Object Notation messages						
	Advanced	Message model						
	Security	Message						
	Lonimmon	a durical freeze to me many a summer was a summer of						

## 5.2 Configure the Mapping node

	Add a mapping hade between the two MO hades					
1.	ad a mapping node between the two MQ hodes.					
	live the mapping node a label of getEmployeeJSON.					
	Connect the manning node to the Get Request (MOInput) and Write Response (MOOutput)					
	odes as shown:					
	■ *getEmployeeMQ.msgflow 🕸					
	GRouting					
	NET					
	Cet Request					
	Mr.NETCompute					
	Security getEmployeeJSON					
	Compute					
2	he man has already been created and stored in the HR. Service. Executables Shared					
<i>-</i> .	ibrary, you will now configure this mapping node to reference that map.					
	elect the aetEmployee ISON map (single click). In the properties tab (Basic) click the					
	Browse button:					
	[™] *getEmployeeMQ.msgflow ⊠					
	▷ Flow Exerciser:					
	Get Request					
	Write Response					
	getEmployeeJSON					
	Graph User Defined Properties					
	<u> </u>					
	🗏 Properties 🏽 🖫 Problems 🗄 Outline 🖉 Tasks 🖩 Deployment Log 👘 🎽 👘 🗖					
	Mapping Node Properties - getEmployeeJSON					
	Description					
	Basic Mapping routine* {default}:getEmployeeMO_getEmployeeJSON Browse					
	Validation Transaction* Automatic					

3.	Select the getEmployeeJSON map from	n the Shared Library and	click OK:				
	Data Transformation Map Selection						
	Filter names (2 = any character $*$ = any String):						
	Filter names (? = any character, * = any String):						
	Select a Data Transformation Map						
	<pre> default}:getEmployeeJSON in Shared Library HR_Service_Executables </pre>						
	Location:						
	HR_Service_Executables						
	(?)	ОК	Cancel				
4.	Double click on the map to make sure y	ou have performed the c	correct configuration.	The			
	map will look like this:						
	etEmployee ISON						
	raetEmployee ISON		<u>8-</u>				
	Solution ⇒		Set to filter>	1			
	■ Properties [01] PropertiesType	Move -	🗉 🞜 Properties	[01]			
	■ JSON [11] JSONMsgType		B P JSON	[11]			
	□ 🗃 Select from HRDB		Padding				
	⇒ <click filter="" to=""></click>	- · · · · · · · · · · · · · · · · · · ·	□ ■ Data	[01]			
	■      ResultSet     [0*] Result Set Row	Select -	UserReturnCode	[01]			
			RowsRetrieved	[01]			
	muser where and the here we are the second of the second o		RowsAdded	[01]			
	Close the man editor						
	Close the map editor.						
5.	Save the getEmployeeMQ message flow	w, it will now look like thi	S:				
	🕫 getEmployeeMQ.ms	sgflow ¤	1				
	Flow Exerciser:	◙∰ ¼ (€, e,	1				
			1				
	Cot Poquet	×	o 🗗 B 👌				
	Get Request	Writ	e Posponse				
			e Response				
			5				
		aetEmployee1SON	3				
		getemployeesoon	1				
	Leave the message flow editor open.						

## 5.3 Test (locally) HR_Service_MQProvider

You will now test the application using the Flow Exerciser to ensure that HR_Service_MQProvider works correctly before using it with the HR_Service REST API.

1.	In the message flow editor start the Flow Exerciser (select the Record button):
	I getEmployeeMQ.msgflow ≅
	Flow Exerciser
	Start flow exerciser to record the path a message takes through the flow.
	Get Request Write Response
	aetEmployeeJSON
	This will automatically deploy the application. If you see messages that need responses respond to them appropriately.
2.	When the Flow Exerciser has started the message flow will have a grey background.
	Click the icon to send a message through the flow:
	In getEmployeeMQ.msgflow №
	Flow Exerciser: 🏼 🖳 🖓 🔍 🔍
	getEmployeeMQ
	Send a message to the flow
	Write Response
	actEmployee ICON
	getEmployeeJSON
	have a contraction of the contra

3.	Select New Message:	
	) Send Message	
	Send Message Create or select a message	to send to the flow. Click the message category header (e.g. Ir
	Input Messages New message	<ul> <li>Manage your messages:</li> <li>➡ - Click 'New' to create an input message.</li> <li>➡ - Click 'Duplicate' to copy the selected input message.</li> <li>➡ - Click 'Delete' to remove the selected messages.</li> <li>- Click a message to view or edit it.</li> </ul>
4.	In the message details window { "en Name the message Employee	v, type the following mployeeNumber'': "000010" } 000010 and click Apply, then Click Send:
	Send Message Create or select a message to send to the f	Now. Click the message category header (e.g. Input Messages) for more information.
	0	Send Close





## 6. Run HR_Service Locally

You will now test that the implementation for **/employees/{employeeNumber}/MQEndpoint** is configured correctly when running **locally** on the same machine as **HR_Service_MQProvider**.

### 6.1 Prepare and deploy a bar file

When you test the REST API on IIB on Cloud you will need to upload a Broker Archive (bar) file to the cloud environment. In this next section you will create a Broker Archive file that you will use to:

- a) test the REST API locally and
- b) (after a rebuild) test the REST API running on IIB on Cloud.



In the bar file editor, on the Prepare tab, select the HR_Service REST API and the two Shared 3. Libraries: HR_Service_Executables and HRDB. You will use this bar file to test the REST API locally and on IIB on Cloud. When deploying the bar file on IIB on Cloud, if you do not include the Shared Libraries in the bar file, the deploy process will fail. Click the "Build and Save" button to save the bar file: HR Service [™] getEmployeeMQ.msgflow [™] *HR_Service_CloudMQandODBC.bar [∞] Prepare Select deployable resources to include in the BAR Deployable Resources 🗄 Build and Save. Select an application to package all its contained resources. Resources within an application are isolated from other applications. O Applications, shared libraries, services, and REST APIs O Message flows, static libraries and other message flow dependencies Text filter: type filter text 🗉 🧾 🐸 Applications 🗄 🔲 🖪 HR_Service_MQProvider REST APIs HR_Service 🖌 🏷 Shared Libraries HR_Service_Executables 🔽 🛋 HRDB (*)-Resource types marked with * will be automatically added to the BAR if referenced by another selected artifact. The bar file will now be saved in the HR_Service REST API. Leave the Bar file editor open you will rebuild the bar file again later after you configure the MQ nodes to use an MQ Endpoint Policy. 4. Drag the bar file to the default Integration Server to deploy the bar file: 🖥 Application Develo... 🛛 🤻 Patterns Explorer 🛎 🖪 🔹  $\nabla$ Application Development New... □ I HR Service REST API Description Resources 🗄 😬 Flows 🗄 🗁 Subflows 🗄 🖉 Maps ESQLs 🖻 🖻 BARs HR_Service_CloudMQandODBC.bar 🖻 🖄 REST API Catalog 🗄 🤹 HR Employee and Department Services 3. 🕶 . ► - -🖧 Integration Nodes 🛛 🖧 Integration Registr... 🔹 🔋 🗉 📲 Integration Nodes E Jibuser 🗄 💤 default MR_Service_MQProvider HR_Service_Executables 💕 HRDB

## 6.2 Test MQ Scenario using Swagger UI





In the next section of this Lab guide you will run the REST API HR_Service on IIB on Cloud and access the same local MQ and DB resources.

## 7. Running HR_Service on IIB on Cloud

In this part of the lab you will use the IIB on Cloud user interface to deploy HR_Service to IIB on Cloud. You will then run the resources

# /employees/{employeeNumber}/MQEndpoint and /employees/{employeeNumber}/CloudODBC

to demonstrate how MQ and DB2 resources installed locally on site can be accessed from the IIB on Cloud Managed Service.

If you require access to IIB on Cloud follow this link to find details on how to register:

https://www.ibm.com/ibmid/basic_register/register_generic.html?a=IBMSystems&ctx=IIB&cc=us &lc=en&catalogName=Master&partNumber=IIB_Trial&quantity=1&trial=yes&S_TACT=C4310CS W

The remaining part of this lab guide requires access to an active IIB on Cloud account.

## 7.1 Configure MQ Endpoint Policy

In the previous sections you configured and tested the HR_Service locally using the Flow Exerciser. The names of the MQ queues and queue manager were explicitly configured on the MQ nodes.

In this next section you will reconfigure the MQ nodes to use an MQ Endpoint Policy. MQ Endpoint policies must be used when running MQ nodes on IIB on Cloud (they cannot be used when testing locally using the Flow Exerciser).

1.	In getEmployeeMQ message flow (in HR_Service_MQProvider).
2.	Select the Get Request (MQInput node) and click Generate new policy (Properties/Policy):
	Properties Problems & Outline @Tasks III Deployment Log Progress  MQ Input Node Properties - Get Request  Description Basic MQ Connection Input Message Parsing Parser Options Advanced Validation Security Instances  Policy Monitoring

2	Configure the MO Delieu as follows, then elick Save:
J.	Configure the MQ Folicy as follows, then click Save.
	HR Service attemployee Re*Policy Editor 2 "
	MQ Policy
	Connection*: Local queue manager
	Queue Manager Name: IB10QMGR
	Queue Manager Host Name:
	Listener Port Number:
	Channel Name:
	Security Identity:
	Use SSL:
	SSL Peer Name:
	SSL Cipher Specification:
	Save
4.	Call the policy "CLOUDPOLICY" and attach it to TESTNODE iibuser. Click Finish:
	I Save Policy
	Save Policy
	Save Policy Save Policy to Integration Registry Specify the policy name and select the integration pode where the
	Save Policy       Save Policy to Integration Registry       Specify the policy name and select the integration node where the policy will be saved
	Save Policy Save Policy Save Policy to Integration Registry Specify the policy name and select the integration node where the policy will be saved Policy name: CLOUDPOLICY
	Save Policy  Save Policy Save Policy to Integration Registry Specify the policy name and select the integration node where the policy will be saved Policy name: CLOUDPOLICY Policy URL: /apiy1/policy/MOEndpoint/CLOUDPOLICY
	Save Policy Save Policy Save Policy to Integration Registry Specify the policy name and select the integration node where the policy will be saved Policy name: CLOUDPOLICY Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY ✓ Attach the generated policy to the node
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>TESTNODE_libuser</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>IESTNODE_iibuser</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>TESTNODE_iibuser</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>TESTNODE_iibuser</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>TESTNODE_iibuser</li> </ul>
	<ul> <li>Save Policy</li> <li>Save Policy to Integration Registry</li> <li>Specify the policy name and select the integration node where the policy will be saved</li> <li>Policy name: CLOUDPOLICY</li> <li>Policy URL: /apiv1/policy/MQEndpoint/CLOUDPOLICY</li> <li>Attach the generated policy to the node</li> <li>Integration Nodes</li> <li>TESTNODE_libuser</li> </ul>
	Save Policy          Save Policy to Integration Registry         Specify the policy name and select the integration node where the policy will be saved         Policy name:       CLOUDPOLICY         Policy URL:       /apiv1/policy/MQEndpoint/CLOUDPOLICY         ✓ Attach the generated policy to the node         ✓ Integration Nodes         ✓ TESTNODE_iibuser
5.	Save Policy         Specify the policy to Integration Registry         Specify the policy name and select the integration node where the policy will be saved         Policy name:       CLOUDPOLICY         Policy URL:       /apiv1/policy/MQEndpoint/CLOUDPOLICY         Policy URL:       /apiv1/policy/MQEndpoint/CLOUDPOLICY         Policy URL:       /apiv1/policy/MQEndpoint/CLOUDPOLICY         Attach the generated policy to the node       Integration Nodes         TESTNODE_iibuser       Testnode_iibuser         Click OK to dismiss the Policy Saved message.       Elick OK to dismiss the Policy Saved message.

6.	The Policy is now say saved there, IIB on C Cloud needs to be co	ved in the Integration Registry on TESTNODE_iibuser. Although it is cloud will not use it locally, however the bar file that you deploy to IIB on onfigured to use and MQ Endpoint Policy.
	In the Integration Reg	gistry window, right click on the policy and select Copy URL:
7.	In the <b>getEmployee!</b> the MQ Output Node	Integration Nod       Integration Registries         Integration Registry on TESTNODE_iibuser         Policies         MQEndpoint/CLOUDPOLICY         Services         Delete
	🔲 Properties 🛛 🕄 Prot	olems 🗄 Outline 🖉 Tasks 🎟 Deployment Log 🤜 Progress 🛛 📑 🗖 📄
	MQ Output No	de Properties - Write Response
	Description	Ise a policy to control the operational behavior of the node at run time. If a More
	Basic a	re set on the MQ Connection tab in the Integration Toolkit at run time.
	MQ Connection	
	Advanced Po	licy URL [®] /apiv1/policy/MQEndpoint/CLOUDPOLICY Generate new policy
	Request	
	Validation	
	Policy	
	Lass ward	
8.	Save the Message flo	ow (ctrl s)
<u>l</u>	<u> </u>	

## 7.2 Rebuild the bar file

The bar file that you will deploy to IIB on Cloud needs to have the MQ Endpoint Policy configuration. In this next section you will rebuild the bar file ready to deploy on IIB on Cloud.

1.	In the bar file editor (you should have the bar file editor open already from the previous bar file prepare), make sure the following are selected:
	(REST APIs) <b>HR_Service</b> (Shared Libraries) <b>HR_Service_Executables</b> and <b>HRDB</b>
	Click the "Build and Save" button to rebuild and save the bar file:
	I HR_Service III getEmployeeMQ.msgflow III *HR_Service_CloudMQandODBC.bar ⊠
	Prepare
	Select deployable resources to include in the BAR
	Deployable Resources Select an application to package all its contained resources. Resources within an application are isolated from other
	Applications, shared libraries, services, and REST APIs O Message flows, static libraries and other message flow dependencies     Text filter: type filter text
	□       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □       □
	Image: Strength Libraries         Image: Strength Libraries
	(*)-Resource types marked with * will be automatically added to the BAR if referenced by another selected artifact.
	The bar file with the MQ Endpoint configuration will now be saved in the HR_Service REST API.

## 7.3 Deploy HR_Service to IIB on Cloud

In this next section you will deploy this updated bar file to IIB on Cloud.

1. If you are using the BetaWorks IIB Workshop VMware image, open the IIB on Cloud link in the Firefox window (IIB on Cloud in the Cloud folder).

Enter the details of your 'IBM Id'/password authorised for IIB on Cloud and click 'Sign In'.

(If you are using your own system sign on to IIB on Cloud using this URL: <u>https://ibm-cloud-ui.ibmintegrationbus.ibmcloud.com/</u>)

2.	You now have your IIB on Cloud default integration space opened.			
	If you have done the introduction lab for IIB on Cloud from this series you will have the sample integration CustomerDatabaseV1 shown in this view.			
	The trial version for IIB on Cloud allows only one integration to run at one time. Therefore, if you have previous Integrations running, please make sure that they are stopped.			
	You will now upload the REST API integration that you created in earlier in this series. Click 'Add Integration'.			
	IBM Integration Bus on Cloud			
	default   CustomerDatabaseV1_fromProduction Stopped			
	Integrations			
	Callable Flows			
	Endpoint Connectivity			
3.	Click 'Upload your BAR file'.			
	IBM Integration Bus on Cloud < Back ? ~ 💽 ~			
	Integrations			
	Callable Flows			
	Endpoint Connectivity Let's upload your BAR file			
	Upload your BAR file			
	A BAR file contains the resources that are required by your integration, such as applications, libraries, and message flows.			
4.	In the Integration Toolkit, make a note of the file system path of your workspace (it will appear in the header of your toolkit window:			
	Tintegration Development - HR_Service_MQProvider/getEmployeeMQ.msgflow - IBM Integration Toolkit C:\Users\iibuser\IBM\IIBT10\Workspace_EndpointConnectory			
	Hie Lait How View Palette Navigate Search Project Kun Window Help [이 ★ 이 집 집 집 같은 않 ★ 이 ★ 및 ★ 1 ★ 3 ★ 5 ☆ ★ → ★   ★   100% ★   ↓   目 1 왕 중 영 월 즉 과   1 F / Quick Access   即   風 Integration Q			
	Application Develop X Retterns Explorer HRDs.abm @ getEmployeeMQ.msgflow X @ HR_Service_CloudMQandODBC.bar @ HRDs.dbm @ getEmployeeJSO			
	Application Development New.			

5.	In the File upload (browser) window, navigate to the HR_Service folder in your workspace (where you saved the bar file that you tested in the previous section), select the bar file and click Open:
	File Upload     C:\Users\iibuser\IBM\IIBT10\Workspace_EndpointConnectors\HR_Service     Search HR_Service
	Organize ▼ New folder
	Favorites  Name  Da
	Libri IBT10
	Iliya     IR_Service_CloudMQandODBC.bar     18       workspace
	Workspace_EndpointConnectors
	L BARfiles
	GeneratedBarFiles
	gen     HR Service Executables
	HR_Service_MQProvider
	HRDB project
	File name: HR_Service_CloudMQandODBC.bar
	Open Cancel
6.	IIB on Cloud will validate the bar file and then show details about the REST API.
7.	As the REST API is not running, you can prepare a limited set of features. By default Basic Authentication is set to ON. Switch this to OFF (click OFF), and click Save:
	> I HR_Service
	> M HR_Service_Executables
	> a HRDB
	Public Endpoints         ^           View how to invoke this integration         ^
	Basic Authentication OFF ON
	If basic authentication is enabled, any request to an HTTP endpoint in this integration must include an HTTP BasicAuth header.
	Find out more about basic authentication
	Cancel Save
Q	The Integration will then show as ' <b>Prenaring</b> ' This may take a few minutes. Click the
о.	"Refresh Listing" at the bottom of the screen to update the page with the latest status of the Integration. The status will change to Stopped:
	HR_Service_CloudMQandODBC     Stopped
	C Refresh Listing
	Whilst the Integration is preparing, you can proceed with the rest of the lab.

## 7.4 Configure IIB on Cloud to connect to local MQ and DB2

In this part of the lab, you will configure how HR_Service will access your local MQ and DB2 environment using Endpoint Connectivity. The resources you will test in HR_Service will access local MQ queues and access your local HRDB DB2 database directly from IIB on Cloud. It will access these local resources using the IIB Secure Connector.

### 7.4.1 Define Endpoints



3.	Click to 'Select endpoint type', then click 'MQ Client Connection'.
	Configuration About your endpoint Select endpoint type MQ Client Connection DB2 ODBC
4.	Configure the MQ Endpoint as follows, click Save when complete:
	<ul> <li>MQ policy name – CLOUDPOLICY (this is the MQ Endpoint policy that is specified on the MQ nodes in the getMQEndpoints subflow in HR_Service)</li> <li>Host - localhost</li> <li>Port - 1441</li> <li>Queue manager – IB10QMGR</li> <li>Channel name – SYSTEM.DEF.SVRCONN</li> </ul>
	CLOUDPOLICY
	Configuration About your endpoint
	MQ policy name* CLOUDPOLICY Host* Port* localhost Queue manager* Channel name* IB10QMGR SYSTEM.DEF.SVRCONN
	Cancel Save
5.	Once the configuration has been saved you will see a screen, listing the Endpoint you just created. If you see a message saying "Okay, who pulled the plug" the IIB on Cloud system thinks that you already have an IIB Switch agent configured on your machine. For the purposes of the next section you can ignore this message.
	Okay, who pulled the plug? We can't connect to your on-premises agent. Please ensure your network is connected. Connect agent

6.	If you have previously had an IIB switch Agent configured on using your IIB on Cloud ID, you will see the following screen. Ignore the message to "Enable network communication with 1 configured endpoint (s) ".
	For this lab guide scenario, you will need to create one more configuration – for DB2.
	Enable network communication with 1 configured endpoint(s)     Enable
	▲ CLOUDPOLICY MQ Client Connection localhost:1441
	Click 'Create endpoint configuration'.
7.	From the drop-down menu select 'DB2 ODBC'.
	Configuration About your endpoint Select endpoint type Mo Client Connection DB2 ODBC
8.	Similar to the MQ endpoint set up, the 'DB2 ODBC' requires information about the connection to



### 7.4.2 Determine IIB Secure Connector configuration

1. If you have not previously configured an agent on IIB on Cloud, click on '**Enable**' and follow the rest of this chapter:

		-	
🖄 Enable net	work communication with 2 config	ured endpoint(s)	Enable
	ICY MQ Client Connection	localhost:1441	$\odot$ $\otimes$
MRDB	DB2 ODBC	localhost:50000	○ ⊗
you have previously co ulled the plug?" messag connector":	nfigured an IIB Secure Connected an IIB Secure Connected an IIB Secure Connected at the tasks in " <b>Set up</b> at tasks in " <b>Set up at tasks in "Set up at tasks in "Set up at tasks in "<b>Set up at tasks in "Set up at </b></b>	tor, you will see the fo	ollowing "Okay, w red IIB Secure
you have previously co ulled the plug?" messag connector":	nfigured an IIB Secure Connect je. Follow the tasks in " <b>Set up a</b>	tor, you will see the fo	ollowing "Okay, w red IIB Secure
you have previously co ulled the plug?" messag onnector":	nfigured an IIB Secure Connect le. Follow the tasks in "Set up a Okay, who pulled t We can't connect to your on-p Please ensure your network	tor, you will see the for a previously configu he plug? premises agent. is connected.	ollowing "Okay, w red IIB Secure
you have previously co ulled the plug?" messag connector":	nfigured an IIB Secure Connect le. Follow the tasks in "Set up a Okay, who pulled t We can't connect to your on- Please ensure your network Connect agent	tor, you will see the for a previously configu he plug? premises agent. is connected.	ollowing "Okay, w red IIB Secure

### 7.4.3 Option1: Set up a NEW IIB Secure Connector

# Follow this section <ONLY> if you are configuring an IIB Secure Connector for the first time in your IIB on Cloud ID.

Once you click on 'Enable' (previous step) you will see a pop up, which shows the steps required for setting up the local agent that will act as the local (on premise) end of the Secure Connector.

The Secure Connector is shipped with IIB v10.0.0.2 (and higher). Configuration of the agent is now required.





### 7.4.4 Option2: Set up a previously configured IIB Secure Connector

Follow this section <ONLY> if you have a previously configured IIB Secure Connector in your IIB on Cloud ID. If you have configured a NEW IIB Secure Connector (Option1 above), go to the Next section.

	Okay, who pulled t	he plug?	
	We can't connect to your on-p Please ensure your network Connect agent	oremises agent. is connected.	-
A HRDB	DB2 ODBC	localhost:50000	
2			



box, Clicl	
3	Check that your agent is connected and has the correct endpoint configuration.
	Finish

### 7.4.5 Confirm MQ and DB2 Endpoint configuration

At this point you will have either created (or updated) the IIB Secure Connector that IIB on Cloud will use to communicate with your local system. In the next section you will test connectivity from IIB on Cloud to your local MQ and DB2 Systems.

1.	In the lis "Test Co	et of Endpoints you will onnection" icon for each	of Endpoints you will see the two Endpoints – MQ and DB2. On the right, click on the nection" icon for each Endpoint ( <i>the small 'plug'-like icon</i> ).					
		Create end	point configuration	<u>⊬</u> ? `	Test connection			
		CLOUDPOLIC	Y MQ Client Connection	n localhost:1441	×			
		HRDB	DB2 ODBC	localhost:50000	€			
2	After a f	ew seconds the Plug	icon will turn to a green ti	ck mark (confirming	that IIBoC can			
2.	communicate with MQ and DB2 On-Premise). The green tick mark will revert back to a plug sign after a few seconds:							
	(+	Create endpoir	nt configuration	± ?				
		CLOUDPOLICY	MQ Client Connection	localhost:1441	$\bigcirc \otimes$			
		HRDB	DB2 ODBC	localhost:50000	• •			
	Lunn		~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	~. <i>^.</i> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~				

# 8. Test HR_Service running on IIB on Cloud

In this part of the lab you will test your IIB on Cloud integration. HR_Service will connect to your local MQ and DB2 systems using the Endpoint configurations you have just defined. Two HR_Service resource URLs will be used to obtain information from the local HRDB database running on your machine.

### 1. /employees/{employeeNumber}/MQEndpoint

Testing the MQ Endpoint connection from IIB on Cloud to your local system, the request will be written to your local MQ environment where the HR_Service_MQProvider application will process it and write a response to a local queue. When written HR_Service will receive the response and pass back to the application calling the REST API.

### 2. /employees/{employeeNumber}/cloudODBC

This HR_Service resource will use the ODBC endpoint definition to access HRDB directly from IIB on Cloud (without accessing any local MQ environments).

## 8.1 Start your IIB on Cloud integration





## 8.2 Test the MQEndpoint integration

default     EmployeeService MQ Cloud       • Running   Actions ~       Integrations       Callable Flows       Endpoint Connectivity	
Integrations       Callable Flows       Endpoint Connectivity         Output         Output         Output	
	Ý
Vew how to invoke this integration	×
Vew the runtime log Download Runtime L	ne Log 🛛 👻





## 8.3 Test the DB2 (ODBC) Endpoint

	Public Endpoints View how to invoke this integration	^ <b></b>
Basi	c Authentication OFF ON	
Serv	ice URLs	
Host: h	ittps://kpnso0ma.ibmintegrationbus.ibmcloud.com	
} ~ 🔳	HR_Service	
	/HR_Services/resources/employees/{employeeNumber}/MQEndpoint	
~	https://kpnso0ma.ibmintegrationbus.ibmcloud.com/HR_Services/resources/emp	ployees/temployeeNumber//MQEndpoint
	/HR_Services/resources/employees/{EDLEVEL}/predictSalary	
}	/HR_Services/resources/employees/{employeeNumber}/department	Show full URL
	/HR_Services/resources/employees	Show full URL
{	(HR_Services/resources/employees/{employeeNumber}/cloudODBC)	Show full URL
} (	https://kpnso0ma.ibmintegrationbus.ibmcloud.com/HR_Services/resources/emp	Hide full UBL ployees/{employeeNumber}/cloudODBC
{	An and the second and an and a second s	
As befo with the	re, copy the url into a separate browser window. Rep e text 000010 for example:	place <b>{employeeNumber}</b> in the



## Appendix

### Recreating the HRDB database and tables

The HRDB database, and the EMPLOYEE and DEPARTMENT tables have already been created on the supplied VMWare image. If you wish to recreate your own instance of this database, the commands

#### 1_Create_HRDB_database.cmd and

2_Create_HRDB_Tables.cmd are provided for this. If used in conjunction with the VM image, these commands must be run under the user "iibadmin". Appropriate database permissions are included in the scripts to GRANT access to the user iibuser.

## END OF LAB GUIDE