

IBM Integration Bus

Creating an Integration Service Accessing Microsoft[®] SQL Server

Featuring:

Basic configuration of SQL Server Defining a new SQL Server data and table Setting SQL Server permissions for IIB Configuring IIB node for SQL Server Testing IIB application with the Flow Exerciser

April 2016 Hands-on lab built at product Version 10.0.0.4

1.	INTR	ODUCTION	3
	1.1	SCENARIO OVERVIEW	3
	1.2		1
			•
2.	SQL	SERVER CONFIGURATION	5
	•		
	2.1	CONFIGURE SQL SERVER NETWORK	5
	2.2	CREATE HRDB DATABASE	7
	2.3	CONFIGURE SECURITY	2
	2.3.1	Create login credentials for iibuser12	2
	2.3.2	Define database authorisation rules for HRDB14	4
	2.3.3	Validate database schema and table permissions10	3
	2.3.4	Check access from iibuser	3
3.	CRE	ATE AND CONFIGURE THE IIB NODE AND APPLICATION)
	3.1	CREATE IIB NODE)
	3.2	SET WINDOWS PATH FOR SQL SERVER AUTHORISATION FOR IIB	ŧ
	3.3	CREATE DATABASE DEFINITION FILE FOR HRDB	5
	3.4	IMPORT THE PREBUILT INTEGRATION SERVICE	2
4.	TEST	THE INTEGRATION SERVICE WITH THE FLOW EXERCISER	3
E	ND OF	LAB GUIDE	ŧ

Microsoft, Windows and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

1. Introduction

An integration service is a specialized application with a defined interface and structure that acts as a container for a Web Services solution.

This lab is based on Lab 1 in this series, Creating an Integration Service. Lab 1 uses a DB2 database, and the scenario creates an integration service that retrieves rows from the EMPLOYEE table in the HRDB database. This is done by using a map to extract the request from the incoming web service request. The map invokes a submap which accesses the database. For more details, please review the lab guide.

1.1 Scenario Overview

In this lab, you will make the necessary configurations in SQL Server and IIB to enable an IIB application (integration service) to connect to SQL Server, and retrieve some rows from a specific SQL Server table.

The integration service is provided for you, and is the solution of the integration service that was developed in Lab 1 (Create Integration Service) in this series of labs.

1.2 Outline of tasks

The tasks to complete in this lab are the following:

- 1. Configure port in SQL Server for TCPIP clients
- 2. Create a new database in SQL Server.
- 3. Configure the appropriate permissions for the database.
- 4. Configure the appropriate Windows configuration for the libuser userID.
- 5. Create a new IIB node
 - a. Specify the required JDBC configurable service
- 6. Use the IIB Toolkit to create a new database definition
- 7. Import the existing solution for the integration service, EmployeeService
- 8. Test the integration service using SQL Server

SQL Server Configuration Configure SQL Server Network

The application that you will use in this lab will use a database named HRDB. This is based on the same HRDB database that was created for DB2 in Lab 1.

In this lab, the HRDB database will be created by the **iibadmin** user, and permissions will be given to the user **iibuser**, so that the IIB node TESTNODE_iibuser will be able to access the database.

- 1. Login to Windows with the user **iibadmin**, password = **passw0rd**.
- 2. From the Windows Start menu, open :

Microsoft SQL Server 2014	
Configuration Tools	
SQL Server 2014 Configuration Ma	anager



3. Expand SQL Server Network Configuration, and select Protocols for SQLExpress.

Right-click the "TCP/IP" protocol, and select Properties.

👼 Sql Server Configuration Manager							
File Action View Help							
🗇 🔿 🖄 🗒 📑 👔							
SQL Server Configuration Manager (Local)	Protocol Name	Status					
SQL Server Services	🐺 Shared Memory	Enabled					
. SQL Server Network Configuration (32bit)	🕷 Named Pipes	Disabled					
⊞	TCP/IP	Fredeland					
🖃 🖳 SQL Server Network Configuration		Enable	-				
Protocols for SQLEXPRESS		Disable					
		Properties					
		Help					

4. Select the "IP Addresses" tab.

TCP/IP Properties	<u>?</u> ×
Protocol IP Addresses	
IP1	
Active	Yes 💌
Enabled	No
IP Address	fe80::51d4:4fb1:9cc3:4a13%10
TCP Dynamic Ports	0
TCP Port	
IP2	
Active	Yes

5. Go to the bottom of the page.

Set TCP Dynamic Ports" to 49688. This port can be varied - we have just picked this at random. It will be referenced by the SQL Server Client, and by the IIB node, so please use this value in the workshop scenario.

Click OK.

	IF AUULESS	1000.100.71.1110 /012
	TCP Dynamic Ports	0
	TCP Port	
	IPAll	
	TCP Dynamic Ports	49688
	TCP Port	
A	ctive	
In	dicates whether the selected IP a	Address is active.
	ОК	Cancel Apply Help
		Пер

6. Close SQL Server Configuration Manager

Integration Service using Microsoft[®] SQL Server

2.2 Create HRDB database

1. Again with the **iibadmin** user, from the Start menu, expand Microsoft SQL Server 2016 CTP 3.2, and select Microsoft SQL Server Management Studio.



2. Ensure Authentication is set to Windows Authentication.

Click Connect.

J Connect to Serve	er X
	SQL Server
Server type:	Database Engine
Server name:	BETAWORKS-ESB10\SQLEXPRESS
Authentication:	Windows Authentication
User name:	BETAWORKS-ESB10\iibadmin
Password:	
	Remember password
	Connect Cancel Help Options >>

3. In the SQL Server Management Studio, expand Databases, and click "New Query" (New Query is used for all interactive commands, not just SQL Query statements).



4. Copy / paste the entire contents of the file

```
C:\student10\integration_service_SQLServer\DBSetup\HRDB.sql
```

into the Query pane.

This script creates the HRDB database, and the EMPLOYEE table, and inserts 42 rows into the EMPLOYEE table.

```
GQLQuery2.sql - BETA...ESB10\iibadmin (54))* 🗙
    USE [master]
                                                                                                  ÷
    GO
    /****** Object: Database [HRDB]
                                        Script Date: 07/01/2016 09:14:39 ******/
  CREATE DATABASE [HRDB]
     CONTAINMENT = NONE
     ON PRIMARY
    ( NAME = N'HRDB', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSS
     LOG ON
    ( NAME = N'HRDB_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS
    GO
   ALTER DATABASE [HRDB] SET COMPATIBILITY LEVEL = 120
   GO
  IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
  ⊟begin
    EXEC [HRDB].[dbo].[sp_fulltext_database] @action = 'enable'
   end
    GO
    ALTER DATABASE [HRDB] SET ANSI_NULL_DEFAULT OFF
    GO
    ALTER DATABASE [HRDB] SET ANSI_NULLS OFF
    GO
    ALTER DATABASE [HRDB] SET ANSI_PADDING OFF
    GO
    ALTER DATABASE [HRDB] SET ANSI_WARNINGS OFF
    GO
    ALTER DATABASE [HRDB] SET ARITHABORT OFF
    GO
    ALTER DATABASE [HRDB] SET AUTO_CLOSE ON
    60
```

5. Click Execute.



6. The database create and insert statements will be executed, hopefully successfully.

100 % -
Query executed successfully.

7. Click Databases, and click the Refresh button.



The HRDB database will be shown in the list of databases.



8. Clear the Query pane.

In the Query pane, type

Select * from IIBADMIN.EMPLOYEE;

Click Execute.

The rows from the EMPLOYEE table should be returned as shown below.

100 %	100 % - (
🛄 Results 📑 Messages											
	EMPNO	FIRSTNME	MIDINIT	LASTNAME	WORKDEPT	PHONENO	HIREDATE	JOB	EDLEVEL	SEX	
1	000010	CHRISTINE	1	HAAS	A00	3978	1995-01-01	PRES	18	F	
2	000020	MICHAEL	L	THOMPSON	B01	3476	2003-10-10	MANAGER	18	М	
3	000030	SALLY	А	KWAN	C01	4738	2005-04-05	MANAGER	20	F	
4	000050	JOHN	В	GEYER	E01	6789	1979-08-17	MANAGER	16	М	
5	000060	IRVING	F	STERN	D11	6423	2003-09-14	MANAGER	16	М	
6	000070	EVA	D	PULASKI	D21	7831	2005-09-30	MANAGER	16	F	
7	000090	EILEEN	W	HENDERSON	E11	5498	2000-08-15	MANAGER	16	F	
8	000100	THEODORE	Q	SPENSER	E21	0972	2000-06-19	MANAGER	14	М	
9	000110	VINCENZO	G	LUCCHESSI	A00	3490	1988-05-16	SALESREP	19	М	
10	000120	SEAN		CONNELL	A00	2167	1993-12-05	CLERK	14	М	
11	000130	DELORES	М	QUINTANA	C01	4578	2001-07-28	ANALYST	16	F	
12	000140	HEATHER	А	NICHOLLS	C01	1793	2006-12-15	ANALYST	18	F	
13	000150	BRUCE		ADAMSON	D11	4510	2002-02-12	DESIGNER	16	М	
14	000160	ELIZABETH	R	PIANKA	D11	3782	2006-10-11	DESIGNER	17	F	
15	000170	MASATOSHI	J	YOSHIMURA	D11	2890	1999-09-15	DESIGNER	16	М	
16	000180	MARILYN	S	SCOUTTEN	D11	1682	2003-07-07	DESIGNER	17	F	
17	000190	JAMES	н	WALKER	D11	2986	2004-07-26	DESIGNER	16	М	
18	000200	DAVID		BROWN	D11	4501	2002-03-03	DESIGNER	16	М	-
Query executed successfully. BETAWORKS-ESB 10\SQLEXPRESS BETAWORKS-ESB 10\jibadm HRDB 00:00:00 42 rows							s				

2.3 Configure Security

2.3.1 Create login credentials for iibuser

1. Again with the **iibadmin** user, in the Server Management Studio, expand Security, Logins.

Right-click Logins, and select New Login.



- 2. Select the General tab, and set the following properties:
 - Login name = BETAWORKS-ESB10\iibuser (use the Search button to search for "iibuser" to use the correct format of the name).
 - Windows authentication = selected
 - Default database = HRDB

Click OK; the user will be added to the SQL Server Logins.

📕 Login - New					<u>- 🗆 ×</u>
Select a page	🔄 Script 👻 📑 Help				
General					
Server Notes	Login name.	BETAWORKS-ESB10	ibuser		Search
Securables	Windows authentication				
🚰 Status	O SQL Server authentication				
	Password:				
	Confirm password:				
	Specify old password				
	Old password:				
	Enforce password policy				
	Enforce password expira	tion			
	🔽 User must change passv	vord at next login			
	C Mapped to certificate			7	
	C Mapped to asymmetric key			7	
	Map to Credential			v	Add
Connection	Mapped Credentials	Credential	Provider		
Server: BETAWORKS-ESB10\SQLEXPRI					
Connection: BETAWORKS-ESB10\iibadmin					
View connection properties					
Progress			_		Remove
Ready	Default database:	HRDB			
The ask	Default language:	cdefault>		•	
				OK	Cancel



2.3.2 Define database authorisation rules for HRDB

1. Expand Databases, HRDB, Security, Users.

Right-click Users and click "New User".



- 2. Select the General tab, and set the following properties. Use the "..." button on each property to search for the correct format of the property value.
 - User type = Windows user

•

•

- User name = BETAWORKS-ESB10\iibuser
- Login name = BETAWORKS-ESB10\iibuser
- Default schema = IIBADMIN

🧵 Database User - New		
Select a page General Owned Schemas Membership Securables Extended Properties	Script - Bername: User name: BETAWORKS-ESB10\vibuser Login name: BETAWORKS-ESB10\vibuser Default language: Default schema: IIBADMIN	

3. Select the Securables tab. Click the Search button.

Select "All objects belonging to the schema", and set the Schema name = IIBADMIN.

Click OK.

<mark>위 Add Objects</mark>	×
What objects do you wish to add?	
O Specific objects	
O All objects of the types	
 A objects belonging to the schema: 	
Schema name: IIBADMIN	
OK Cancel He	lp

4. In the Permissions area, tick all the Grant boxes for all types of access.

Click OK to add these database permissions to the BETAWORKS-ESB10\iibuser user.

🧊 Database User - New						
Select a page	Script - 🕞 Help					
🚰 General	<u>a</u> . «a .					
Cwned Schemas	User name: BETAWORKS-ESB1	0\iibuser				
Membership						
	Securables:			Search		
	Schema	Name	Туре			
	IIBADMIN	EMPLOYEE	Table			
	I					
	Permissions for IIBADMIN.EMPLOYE	E:	Colo	mn Permissions		
Connection	Evelicit 1	-				
Server:	Explicit					
BETAWORKS-ESB10\SQLEXPR	Permission	Grantor	Grant With	Grant Deny 🔺		
Connection:	Alter					
BETAWORKS-ESB10\iibadmin	Control					
View connection properties	Delete					
_	Insert					
Progress	References					
All Parts	Select					
Ready	Take ownership					
~4p*	↓	·				
				OK Cancel		

2.3.3 Validate database schema and table permissions

In the previous section, you set the required database permissions for the user iibuser. You will now check these permissions using the database views.

1. Expand Databases, HRDB, Tables.

Right-click IIBADMIN.EMPLOYEE, and select Properties.

 ■ BETAWORKS-ESB 10\SQLEXPRESS (■ Databases ■ System Databases ■ HRDB ■ Database Diagrams ■ Tables ■ System Tables ■ FileTables 	(SQL Server 12.0.2000 - BETAWORKS-
Image: Security Image: Security	Table Design Select Top 1000 Rows Edit Top 200 Rows Script Table as View Dependencies Full-Text index Full-Text index Stretch Policies Facets Start PowerShell Reports Rename Delete Refresh
	Properties

2. Select the Permissions tab.

You will see that the EMPLOYEE table shows BETAWORKS-ESB10\iibuser as a user who has full access rights.

Table Properties - EMPLOYEE						
Select a page	🕻 Script 👻 📑 Help					
General						
Permissions	Schema: IIBADM	lin				
Storage	View ophome pormit we					
Extended Properties	view schema per asion					
	Table name: [EMPLC	YEE				
	Users or roles:				Search	
	Name			Type		
	A BETAWORKS-E	SB10\iibuser		User		
				000.		
	Permissions for BETAW	ORKS-ESB10\iibuser:		Column Pe	ermissions	
	Explicit Effective					
Connection	Permission	Grantor	Grant	With Grant	Deny	
Server:	Alter					
BETAWORKS-ESB10\SQLEXPR	Alter	dbo				
Connection:	Control					
BETAWORKS-ESB10\iibadmin	Control	dbo				
■ View connection properties	Delete					
	Delete	dbo				
Progress	Insert					
Beach	Insert	dbo	V			
neauy	References					
.db.	References	dbo				~
				ОК	Can	cel

2.3.4 Check access from iibuser

Finally, to validate that the permissions have been specified correctly, use the user **iibuser** to access the HRDB database.

1. Login to Windows with the user **iibuser**, password = **passw0rd**.

(Use Windows, Start, Switch user).

2. From the Start menu, open Microsoft SQL Server 2016 CTP3.2.

Open "Microsoft SQL Server Management Studio".

Microsoft SQL Server 2016 CTP3.2	
Download Microsoft SQL Server Compac	Computer
Kicrosoft SQL Server Management Stud	,
SQL Server 2016 CTP3.2 Import and Ex	Control Panel
🍌 Analysis Services	control and
Configuration Tools	Devices and Printe
Integration Services	
Performance Tools	Default Programs
\mu Mozilla Firefox	
Notepad++	Help and Support
◀ Back	
Search programs and files	Shut down
🏹 Start 🛛 ᇘ 💿 🙂	

3. Click Connect.

Connect to Serve	r X
	SQL Server
Server type:	Database Engine
Server name:	BETAWORKS-ESB10\SQLEXPRESS
Authentication:	Windows Authentication
User name:	BETAWORKS-ESB10\vibuser
Password:	
	Remember password
	Connect Cancel Help Options >>

4. Click New Query.

In the query pane, enter :

```
Use HRDB;
Select * from IIBADMIN.EMPLOYEE;
```

Click Execute.

SQLQuery1.sql - BETAWORKS-ESB10\SQLE	XPRESS.HRDB (BETAWORKS-ESB10\iibuser (53))* - Microsoft SQL Serv
File Edit View Query Project Debug To	ols Window Help
🗄 🛅 🕶 📨 😂 属 🍠 📜 New Query 🗋	😬 🕾 🖏 🌡 🖿 🖭 🄊 🔹 🖓 - 🎘 - 💷 🚳 🍇 🕨
HRDB	Execute Debug 💻 🗸 👯 🗐 🔒 👫 🖓 🍓 🏹 🏹
Object Explorer 🔹 부 🗙	SQLQuery1.sql - BETAESB10\jibuser (53))* ×
Connect 🕶 📑 📑 🖉 🍙 🏹	EUse HRDB;
E KAWORKS-ESB10\SQLEXPRESS (SQL S	Select * from IIBADMIN.EMPLOYEE;
🖃 🧰 Databases	
🛨 🚞 System Databases	
🗄 🗉 🔰 HRDB	
🗄 🔃 Security	
🛨 📄 Server Objects	
🛨 🚞 Replication	
🛨 🚞 Management	

5. You should see the returned rows from the EMPLOYEE table.

This has confirmed that **iibuser** has the correct permissions to access the EMPLOYEE table in the HRDB database.

	Results Messages											
l		EMPNO	FIRSTNME	MIDINIT	LASTNAME	WORKDEPT	PHONENO	HIREDATE	JOB	EDLEVEL	SEX	
	1	000010	CHRISTINE	1	HAAS	A00	3978	1995-01-01	PRES	18	F	
	2	000030	SALLY	Α	KWAN	C01	4738	2005-04-05	MANAGER	20	F	
	3	000050	JOHN	В	GEYER	E01	6789	1979-08-17	MANAGER	16	М	
l	4	000060	IRVING	F	STERN	D11	6423	2003-09-14	MANAGER	16	М	
	5	000070	EVA	D	PULASKI	D21	7831	2005-09-30	MANAGER	16	F	
l	6	000090	EILEEN	W	HENDERSON	E11	5498	2000-08-15	MANAGER	16	F	
l	7	000100	THEODORE	Q	SPENSER	E21	0972	2000-06-19	MANAGER	14	М	
	8	000110	VINCENZO	G	LUCCHESSI	A00	3490	1988-05-16	SALESREP	19	М	

6. No further use will be made of the Server Management Studio, so close it now.

3. Create and configure the IIB node and application

3.1 Create IIB node

In previous labs in this series, you have used the generated TESTNODE_iibuser node to run the IIB applications and services. The TESTNODE node is created automatically by the IIB Toolkit, and uses a SYSTEM user as the owning user (ie. the corresponding "mqsi" command does not include the "-i iibuser" parameter.

However, when using Windows integrated authentication, it is necessary to create a new IIB node which uses a specific user, not the SYSTEM user.

- 1. Login to Windows with the user **iibuser**, password = **passw0rd**. (You are already there, if you are following the steps in this lab guide).
- 2. Start the IIB Toolkit, and create a new workspace, for example with the name "\workspace_SQLServer", as shown.

Click OK.

🌐 Workspace Launcher		×
Select a workspace		
IBM Integration Toolkit stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.		
Workspace: C:\Users\ibuser\IBM\IIBT_0\workspace_SQLServer	•	Browse
Copy Settings		
?	ОК	Cancel

3. Create a new Local Integration Node:

Right-click Integration Nodes, New, Local Integration Node.

[않 In 없 않 In.	🔁 Da 🎬 Da	8		
L					
L		DE_iit		Þ	를 Local Integration Node
L					
L					
L		🖑 Refresh			
					-

- 4. Set the following properties:
 - Name = IB10NODE_SQLS
 - User name = iibuser
 - Password = passw0rd

Click Finish.

The new node will take a few moments to create. When complete, click Close.

local Integrat	tion Node
Create a new local i	integration node 🚔
Integration Node *Integration node nat *User name: *Password:	ne: IB10NODE_SQLS iibuser
Start the integration	on node when Windows starts
Server *Default integration so Default debug port:	erver name: default
Default Queue Mar Configure a local queu Queue manager name	nager ne manager to be used by all the nodes that do not explicitly specify MQ connection :
Note that if you run the iib_creat AggregateReply, TimeoutNotificati	plan to use any of the following nodes in message flows or subflows, you will need to equeues script that creates the necessary system queues: AggregateControl, , AggregateRequest, Collector, Resequence, Sequence, TimeoutControl, on
?	<u>F</u> inish Cancel

5. Create a Configurable Service for the JDBC connection to the HRDB SQL Server database.

In an IIB Command Console, run the following commands:

cd \student10\integration_service_SQLServer\ConfigService

create_JDBC_for_HRDB_SQLSERVER

Accept the default node (IB10NODE_SQLS) for the command.

6. Review the configurable service you just created using a web browser.

In the IIB Toolkit, right-click the node IB10NODE_SQLS to automatically open a web admin browser connected to this node.

🖧 Integrati 🛛 🎖 Inte	egrati 😤 Data Proj 🎁 Data S
🗆 📲 Integration Nodes	
⊡섉 IB10NODE_SQLS ⊡ <mark>ॡ</mark> default	😤 New Integration Server
	Stop
	of Refresh ∰ Change
	💢 Delete
	👩 Start Web User Interface
	Y Open Policy Sets

7. Expand Operational Policy, Configurable Services, JDBC Providers, and select HRDB.

Review the following properties carefully, which have been changed from the default values:

- databaseType Microsoft SQL Server
- jdbcProviderXASupport false (IIB does not support the XA protocol with SQL Server databases. If you leave this value set to true, you will observe JDBC XA errors when the service is executed).
- portNumber set to the SQL Server TCPIP port you set earlier.
- jarsURL this is the directory name that holds the Microsoft JDBC drivers. Note that this is not included in the main SQL Server package, and has to be downloaded and installed (unzipped) separately.
- databaseName HRDB
- securityIdentity This property is not used in this lab scenario, because we have chosen to use Windows integrated security (see next property). If we had configured the lab in the same way as the DB2 version, we would normally set this property to a Security Identity defined by a corresponding mqsisetdbparms command.
- connectionUrlFormat This property has been changed to replace the string "user=[user],password=[password]", with the string "integratedSecurity=true". This means that the user credentials that are used to start the IIB node are used to authenticate and authorise access to the SQL Server system automatically by Windows. (The user is iibuser in this lab).

Overview		
		🧷 E0
 Properties 		
turad		and discuss fit and a sub-state of the COL Sector VADate Sector
type4	Datasourcectassname	com.microsoft.sqlserver.jdbc.SQLServerAdbataSource
datab	aseType	Microsoft SOL Server
idhcP	roviderXA Support	false
portN	umber	49688
conne	ectionUrlFormatAttr5	
conne	ectionUrlFormatAttr4	
serve	rName	localhost
conne	ectionUrlFormatAttr3	
conne	ectionUrlFormatAttr2	
conne	ectionUrlFormatAttr1	
envire	onmentParms	default_none
maxC	onnectionPoolSize	0
descr	iption	default_Description
jarsUl	RL	C:\Program Files\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\enu
datab	aseName	HRDB
datab	aseVersion	default_Database_Version
secur	ityldentity	None required because we are using Windows Integrated Authentication (integratedSecurity=true instead of username-password security in connectionURL format parameter)
conne	ectionUrlFormat	jdbc:sqlserver://[serverName]:[portNumber];DatabaseName= [databaseName];integratedSecurity=true
datab	aseSchemaNames	useProvidedSchemaNames

3.2 Set Windows Path for SQL Server authorisation for IIB

Both the IIB Toolkit and the IIB runtime node require access to the file **sqljdbc_auth.dll**. This file is required when using Windows integrated authentication to connect IIB components to the SQL Server database system, which will be the method used in this lab. (Note - this requirement is not restricted to IIB - see multiple items on many SQL Server forums for more discussion).

This file is shipped in the SQL Server JDBC package. In the IIB workshop lab VMWare image, the Windows Path has already been changed to include the directory:

c:\Program Files\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\enu\auth\X64

You can check this by running the DOS command "Set Path".

If you are running this lab on your own Windows system, you will need to make a similar change, and rebooting your system.

3.3 Create Database Definition file for HRDB

In this part of the lab, you will use the IIB Toolkit to connect to the HRDB database on SQL Server, and use this to create an IIB database definition. This would then typically be used to create an IIB Message Model (see Lab 1 in this series).

We have included these IIB Toolkit tasks for SQL Server here, because they are a little different from the equivalent tasks when connecting to DB2. However, because the existing EmployeeService used in the DB2 version of this lab has been written, this will be used to perform the runtime part of this lab. Hence, later in this lab, you will import the pre-built solution of EmployeeService (demonstrating portability of IIB applications across different database vendors - the original application was built for use with DB2). The database definition that you are about to create will not actually be used when this service is implemented and tested.

Make sure you are logged in as "**<u>iibuser</u>**", and open the IIB Toolkit using the shortcut on the Windows Start menu (if already open from the previous step, you can use this).

🔚 Application Dev 🛛 💐 Patterns Explore	r 🗆 🗆 🗍	
🛎 E	1 € , ▼	
Application Development	New	
New Application		
New Integration Service		
New REST API		
New Library	New	Message Flow
	Сору	∎ [≜] Subflow
	Paste	Message Model
	Delete	🔁 Message Map
	Move	😭 ESQL File
	Rename	🛱 Broker Schema
<u>2</u>	Import	😂 Adapter Connection
<u>1</u> 2	Export	🧵 Database Definition
	Refresh	🚉 Data Lineage Documents
		💳 👔 BAR file
		🕞 Decision Service
		🕼 MQ Service
		🕼 Database Service

1. In the Toolkit navigator, right-click in blank space, select New, Database Definition.

2. Click New to create a data design project.

🌐 New Database D	efinition File					
Create a database definition file Specify the database type, version and the data design project that will contain the database definition file.						
Data design project:	DB2 for Linux, LINIX, and Windows	T				New
Version:	V10.1 •					
?			< Back	Next >	Finish	Cancel

3. Name it HRDB. Click Finish.

🌐 New Data Design Project		
Create a data design project		
Specify a basic definition for the new project. This project stores of	lata design object	ts.
Project name: HRDB		
Use default location		
Location; C:\Users\iibuser\IBM\IIBT10\workspace2\HRDB		Browse,
_ Working sets		
Add project to working sets		
Working sets;	_	Select
(?)	Finish	Cancel

4. Set the Database to SQL Server, and the Version to 2012 (2014 is supported and SQL Server Express Edition 2014 is installed on the workshop VM system).

Click Next.

🌐 New Database D	New Database Definition File				
Create a database definition file Specify the database type, version and the data design project that will contain the database definition file.					
Data design project: Database: Version:	HRDB SQL Server	V			New
			\frown		
?		< Back	Next >	Finish	Cancel

5. Click New to create a new connection.

New Database Definition Select Connection Select an existing connection.	File				
Connections ────────────────────────────────────					New Laicht
Property	Value				
?		< Back	Next >	Finish	Cancel

6. At the top of the window, you will see a message that says the required JDBC driver class, sqljdbc4.jar, was not found.

Wew Connection	
Connection Parameters	
8 The JDBC driver class was not found in the specified JAR or .zip file. Click the ellipsis () button to update the path to the Current file and path location: sqljdbc4.jar	JDBC driver files.
Connection identification Use default naming convention Connection Name: pubs	
Select a database manager: JDBC driver: SQL Server 2005 - Microsoft SQL Server 2005 JDBC Driver: SQL Server 2005 - Microsoft SQL Server 2005 JDBC Driver: Properties	r Default 💌
General Optional Database: pubs	
Host: localhost	
Port number: 1433	

7. Click the JDBC driver dropdown, and select "SQL Server 2012 with system tables".



8. Click on the three dots button to the right of the JDBC driver dropdown. This will show you the name of the JDBC driver file that the Toolkit is looking for.



9. Using the "Edit JAR/.zip" button, specify the location of the sqljdbc41.jar file.

On the IIB workshop system, this is

```
C:\Program Files\Microsoft JDBC Driver 4.2 for SQL
Server\sqljdbc_4.2\enu\Sqljdbc41.jar
```

Click OK.

🌐 Edit JAR List	×
Specify JAR List	
Specify the file path to the JAR or .zip files that are required by the JDBC driver. Some of are not included with this product and must be obtained separately. Typical file names and	drivers re shown
Driver files:	
C:\Program Files\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\enu\sqljdbc41.jar	Add JAR/.zip
	Edit JAR/.zip
	Remove JAR/.zip
	Clear All
?	Cancel

•

- 10. Set the following properties:
 - Database name = HRDB
 - Host = localhost
 - Port number = 49688 (this must match the change you made earlier for the SQL Server TCPIP port)
 - Use integrated authentication = ticked

Click Test Connection, to ensure you have set the right properties.

Assuming the Test Connection is successful, click Finish.

IDBC driver: SQL Se	erver 2012 - Microsoft SQL Server 2012 JDBC Driver with system table▼	
Properties		
General Option	al	<u> </u>
Database:	HRDB	
Host:	localhost	
Port number:	49688	
Use integrate	d authentication	
User name:		
Password:		
🗖 Save passwo	rd	
Connection URL:	jdbc:sqlserver://localhost:49688;databaseName=HRDB;integratedS ecurity=true;	
	Test Conr	nection

11. Highlight the new connection (HRDB1), and note the various connection properties that are shown (confirming the definitions you just made).

Click Next.

New Database Defini	tion File
elect Connection	tion.
Connections	
HRDB1	New
	Edit
	Delete
 Properties 	
Property	Value
Name	HRDB1
Description	
Category	Database Connections
Database	HRDB
JDBC Driver Class	com.microsoft.sqlserver.jdbc.SQLServerDriver
Class Location	C:\Program Files\Microsoft JDBC Driver 4.2 for SQL Server\sqljdbc_4.2\enu\sqljdbc41.jar
Connection URL	jdbc:sqlserver://localhost:49688;databaseName=HRDB;integratedSecurity=true;
User ID	
1	
?	< Back Next > Finish Cancel

Integration Service using Microsoft[®] SQL Server

12. Select the IIBADMIN schema, and click Finish.

(Note that the HRDB database has been defined with the IIBADMIN schema. Recall that you earlier granted permissions for this user on this database/schema).

Select Objects Select schema(s) to reverse engineer. Apply name filter (? = Any character, * = Any string): Select objects:	Clear Filter
Apply name filter (? = Any character, * = Any string): Select objects: db_accessadmin db_backurseerster	Clear Filter
Select objects:	Clear Filter
Select objects:	Sicci Friter
db_accessadmin	
	Select All
	Deselect All
db_datareader	
db_denydatareader	
db_denydatawriter	
db_owner	
guest	
Sack Next > Fir	ish Cancel

13. The database definition for the HRDB database is now present as a project in the Independent Resources part of the navigator.

Note that this HRDB.dbm file can be used by any other developer requiring access to the database. It contains a definition of the database, and the selected tables and schemas, and can be shared as required, to create the related IIB schema definitions.

🔚 Application	x	뭖	Pat	tern	s Exp	ol		
					Ċ	Ē	\$₽¢	\bigtriangledown
Application Development New							ew	
⊡… [_] Independer ⊡…' [] HRDB	nt Re DB.d	sour	ces					

3.4 Import the prebuilt integration service

Although we have shown you how to use the IIB Toolkit to connect to a SQL Server database, and create a database definition, the rest of this lab will use the existing integration service that was built in Lab 1 (Employee Service). This application will be used unchanged, and will therefore demonstrate that IIB applications are independent of the connected database, providing that the database properties such as database name, schema name and table structure are the same across the different databases.

1. In the IIB Toolkit, import the PI file

c:\student10\integration_service\solution\ EmployeeService.10.0.0.3.zip

Import all projects. Because you have earlier created a project named HRDB, this will be overwritten with the project HRDB contained in the PI file. The Toolkit will ask you to confirm the over-write of the HRDB project. Answer Yes to this.

The imported projects will be shown like this.

Expand the EmployeeService, and double-click the Integration Service description. This will open the service definition for EmployeeService.

🕽 Integration Development - IBM Integration Toolkit - C:\Users\iibuser\IBM\IIBT10\workspace_SQLServer						
File Edit Navigate Search Project Run Window Help						
🖆 ▾ 🗐 🖷 ≜ @ 📽 🌣 ▾ 💽 ▾ 💊 ▾ 🖋 ▾ 🖉	Y ₩ Y ← Y → Y Quick Access	Integration Development				
🔚 Application Develop 🙁 👯 Patterns Explorer 💛 🗖	Market EmployeeService					
🛎 🖻 🔄 🏹	S EmployeeService >	• H				
Application Development New Image: Service Image: Service Description Image: Service Interface Location Image: Service Service Interface and maps Image: Service Interface Interface and maps Image: Service Service Interface Interfa	SEmployeeService ImployeeService ImployeeServi	EmployeeService actEmployee Error Handlers Error Handlers Eng Ealure Ga Catch Ga Timeout				

4. Test the Integration Service with the Flow Exerciser

This lab will use the Flow Exercise to perform a simple unit test of the service you have just imported.

First, make sure that the node IB10NODE_SQLS is started. You can check this in the Toolkit Integration Nodes pane. If it isn't started, right-click and select Start.

1. In the service editor, make sure the Service tab is selected. You will see a red button in the top right of the editor pane.

📴 EmployeeService 🛛		
S EmployeeService >		• F =
S EmployeeService	(I) EmployeeService	
SOAP/HTTP Binding	🤯 getEmployee	
	Error Handlers E Failure E Catch E Timeout	-
Service 💿 Interface		j
🔲 Properties 🔀 💦 Problems 👫 Ou	ıtline 🧔 Tasks 🖽 Deployment Log	2 - 0

2. Click the red button. The integration service will be deployed to the IB10NODE_SQLS/default server.

Note that if you have other nodes or servers running, you will be presented with a selection dialogue. Make sure you select the correct node/server.

When the service is deployed, you will see an information message. After reading this, select the box to not see in the future, and click Close.



Integration Service using Microsoft[®] SQL Server

3. The flow editor will turn grey, indicating that the Flow Exerciser is ready for use.

In the top right, the available icons will have changed. Select the "Send message" icon (the middle icon).

EmployeeService 🛛		
S EmployeeService >		🛛 🕅 🕅
S EmployeeService	EmployeeService getEmployee	
	Error Handlers	
	E Timeout	

4. In the Send Message window, click the button to create a new message.

🌐 Send Message		
Send Message Create or select a message to send to the flow. Click the message category header (e.g. Input Messages) for more information.		
Input Messages	Manage your messages: 2 - Click 'New' to create an input message. 2 - Click 'Duplicate' to copy the selected input message. 2 - Click 'Delete' to remove the selected messages. - Click a message to view or edit it.	

5. Name the new message "Employee 000010".

The Flow Exerciser will have automatically populated a template input message, based on the WSDL and associated service operation input message.

In the employeeNumber element, change the value to 000010, and click Send.

🌐 Send Message	
Send Message Create or select a message to send to the	ne flow. Click the message category header (e.g. Input Messages) for more information.
Imput Messages Imput Messages Imployee 000010	Name Employee 000010 Main Input Location: SOAP Input Soap operation getEmployee • Message Details • • Edit, type, or import a message. Import from file • <tns0:header></tns0:header> <tns0:header></tns0:header> <tns0:envelope xmlns:tns0="7http://scher.</td"> • <tns0:header></tns0:header><tns0:header></tns0:header><tns0:envelope xmlns:tns0="7http://scher.</td"> • <tns0:header></tns0:header><tns0:body></tns0:body></tns0:envelope> • <tns0:header></tns0:header><tns0:envelope> • Show in hexadecimal viewer (Read Only) • Export Source Apply Revert</tns0:envelope></tns0:envelope>
?	Send Close

6. The service operation will be executed. Highlighting the line "Sending Message to SOAP Input. This will show the message that was sent to the service.

e	Progress Information	<u>_ ×</u>
	Invoke Message Flow (Employee 000010) Message flows deployment successfully completed Starting Sending Message to "SOAP Input" Received HTTP reply message for "SOAP Input" Stopped	
	<pre><?xml version="1.0" encoding="UTF-8"?><tns0:envelope xmlns:q10="http://www.example.org/employeeNumber" xmlns:tns0="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tns1="http://EmployeeService" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"></tns0:envelope></pre>	-

7. Highlighting the "Received HTTP" line will show the message returned from the service. Note that the employee record for the selected employee number has been retrieved from the EMPLOYEE table.

Also note the UserReturnCode = 0, and RowsRetrieved = 1.

Progress Information	_ 🗆 ×
E Invoke Message Flow (Employee 000010)	
Message flows deployment successfully completed	
🚊 🖓 Karting	
Received HTTP reply message for "SOAP Input"	
Stopped	
xml version="1.0" encoding="UTF-8"? <soapenv:envelope< td=""><td>-</td></soapenv:envelope<>	-
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">	
<soapenv:body></soapenv:body>	
<io2:getemployeeresponse <="" td="" xmlns:io="http://hrdb/iibadmin"><td></td></io2:getemployeeresponse>	
xmlns:io2="http://EmployeeService"	
xmins:xsi= http://www.w3.org/2001/XMLSchema-instance >	
<id:employeeresponse></id:employeeresponse>	
<userpeturocode <="" s0="" s<="" td="" userpeturocode=""><td></td></userpeturocode>	
<bowsretrieved>1</bowsretrieved>	
<rowsadded xsi:nil="true"></rowsadded>	
<rowsupdated xsi:nil="true"></rowsupdated>	
<rowsdeleted xsi:nil="true"></rowsdeleted>	
<sqlcode_errorcode xsi:nil="true"></sqlcode_errorcode>	
<sqlstate_sqlstate xsi:nil="true"></sqlstate_sqlstate>	
<sql_error_message xsi:nil="true"></sql_error_message>	
<out:employee xmlns:out="http://hrdb/iibadmin"></out:employee>	
<empno>000010</empno>	
<pikstnme>CHKISTINE</pikstnme>	
<phoneno>3978</phoneno>	
<hr/>	_
	diam 1
	Close

Close the Progress Information window.

8. In the flow editor, you will now see that some parts of the service definition have turned green. Click the "getEmployee" operation name.

	📝 EmployeeService 🛛	
	S EmployeeService >	
	S EmployeeService	EmployeeService aetEmployee
K Error Handlers		🔀 Error Handlers
□ _□ Failure		🗏 😑 Failure
□ _□ <u>Catch</u>		E Catch
Interview Interv		🗉 🗉 Timeout

9. In the flow editor, you will see the subflow that implements this operation. You will see that the connectors are green, indicating that the message flowed down this path.

💿 EmployeeService 🕨 👹 g	jetEmployee 🕨 🗉 Request_Response
👌 😳 Palette	
💫 🖌 💷 🔺	
🙀 Favorites	
院 WebSphere MQ	Input 🗹 Output
RP MQTT	
Gms JMS	getEmployee_WS
💭 НТТР	
🚱 Web Services	
🔁 SCA	
🐻 WebSphere Adapters	
Consultant	

10. Click the icon on the first connector.

The message tree that was current at the time it passed through this connector will be retrieved and displayed. Note that this is the full message tree, not just the user data.

You can expand various parts of the tree, such as LocalEnvironment, Environment, ExceptionList, as well as the Message.

Note that the domain of the message tree is shown. In this case, it is XMLNSC.

Input Output	
Recorded Message	
Environment	
Local Environment	
Exception List	
▼ Message	
+ <properties></properties>	
I <httpinputheader></httpinputheader>	
□ <xmlnsc></xmlnsc>	
<: XmlDeclaration >	
<version>1.0</version>	
<encoding>UTF-8</encoding>	
NS1:getEmployee>	
<employeenumber>000010</employeenumber>	

11. Click the icon on the final connector. This time, you will see the message tree after the employee record has been retrieved from the database.

Note that you can return to the first connector icon to review the message tree again, at that point. The data is not discarded.

Input	Output
getEmployee_WS	Recorded Message
	Environment
	Local Environment
	Exception List
	▼ Message
	☐ <message></message>
	□ <xmlnsc></xmlnsc>
	<io2:getemployeeresponse></io2:getemployeeresponse>
	io:EmployeeResponse>
	<pre> <out:employee> </out:employee></pre>
	<eirstnme>CHRISTINE</eirstnme>
	<midinity <="" midinity<="" th=""></midinity>
	<lastname>HAAS</lastname>
	<workdept>A00</workdept>
	<phoneno>3978</phoneno>
	<hiredate>1995-01-01</hiredate>
	<job>PRES </job>

12. Click the Send Message icon again.



13. Create a new test as before. Name it "Employees matching 0020".

Set the employeeNumber to 0020, and click Send.

-	
	Name Employees matching 0020 Main Input Location: SOAP Input Soap operation getEmployee • Message Details Edit, type, or import a message. • Import from file xml version = "1.0"encoding= 1/1F-8"? <tns0:envelope "http:="" sche<="" td="" xmlns:tns0=""> <tns0:header></tns0:header><tns0:header></tns0:header><tns0:einvelope< td=""> <tns0:body> <tns0:body> Show in hexadecimal viewer (Read Only) Export Source</tns0:body></tns0:body></tns0:einvelope<></tns0:envelope>

14. The returned message will contain two employees, both containing "0020" in the employee number field.

RowsRetrieved = 2.



15. Create a new test as before. Name it "Employee 000012".

This time, select "Import from file". Using the File system button, navigate to c:\student10\integration_service\data\employeeNumber 000012.xml.

Name Employee 0	00012
Input Location:	SOAP Input
Soap operation	getEmployee
Message Deta	ils
Edit, type, or i	mport a message.
Import from	n file
File name:	Workspace File system

16. Click Send, and use the same tools to review the test outcome.

🌐 Send Message	×
Send Message	
Create or select a message to send to the flo	ow. Click the message category header (e.g. Input Messages) for more information.
Imput Messages Imployee 000010 Imployees matching 0020 Im	Name Employee 000012 Main Input Location: SOAP Input Soap operation getEmployee Message Details Edit, type, or import a message. Import from file File name: C:\student10\Integration_service\data\employeeNumber 000012.xml Workspace Ymo: Price (2, 1, 0, *encoding = 2, 0,
•	Send Close

17. You will see no rows have been returned, and the EMPLOYEE part of the message is not present.

RowsRetrieved = 0.



END OF LAB GUIDE