## 4. Building and deploying the sample EBP project for Heirloom PaaS

You should have already downloaded and installed Eclipse and the Heirloom plugin. To recap:

Download and install Eclipse for J2EE from here : <u>http://www.eclipse.org/downloads/</u> Log into paas.heirloomcomputing.com and download our plugin, following instructions there for installation and license placement

Download Sample Source bundle from here: <u>http://www.elasticcobol.com/downloads/PaaSDemo.zip</u>

## **Build:**

## **Open Eclipse**

Right click the Project Explorer and select New->Project... :



Choose COBOL Project and click Next:

000	New Project		
Select a wizard Create a COBOL Project			\$
Wizards:			
type filter text			
<ul> <li>Java Project from Existing Ant</li> <li>Plug-in Project</li> <li>General</li> <li>Eclipse Modeling Framework</li> <li>EJB</li> <li>Elastic COBOL</li> <li>COBOL Project</li> <li>Elastic Transaction Platform</li> <li>Java</li> <li>Java EE</li> <li>JavaScript</li> <li>JAXB</li> <li>JPA</li> </ul>	Buildfile		
?	Next >	Cancel Finis	ih

Name the project 'ListTable', select 'Default Project' under Project Type and click Finish:

000	New
COBOL Pro	ject
Create a ne	
Project na	me: ListTable
🗹 Use de	fault location
Location:	/Users/ianw/Documents/workspace-azure/ListTable Browse
	Choose file system: default \$
Defer	Project type Project directories
► Sampl ► Tutori Diagn	es als ostic
Details	
This p	roject template represents the default project setup.
?	< Back Next > Cancel Finish

Expand the ListTable project:



From your source bundle drag the ListTable.cbl file into the cobol\_source folder and choose to 'Copy files' and click OK:

000	File Operation
Select how files s	hould be imported into the project:
Copy files	
🔵 Link to file	s
☑ Create	link locations relative to: PROJECT_LOC +
Configure Drag a	nd Drop Settings
?	Cancel OK

Expand the intermediate\_java\_code folder and note that Java has been generated from the COBOL:



Drag the file test.jcl into the workdir folder and postgresql-9.2-1003.jdbc4.jar into the workdir folder, again choosing to 'Copy files' :



Right click the ListTable Project and choose properties, then click 'Java Build Path':



Click 'Add JARS' and expand the ListTable project to the workdir, select postgresql-9.2-1003.jdbc4.jar and click OK:

▼ CistTable		
🕨 🗁 .executable		
cobol_source		20
intermediate_java_code	2	
Verkdir	Didbed in	
stgresqi-9.2-100	13.Jubc4.jar	
X).classpath		
X .project		
▶ → RemoteSystemsTempFiles	i	

Open the cobol\_source folder , double click ListTable.cbl to bring up the source code. Edit the jdbc driver information with the jdbc connection string you copied from the Amazon RDS console properties. Remember to add in the user and password:



Save ListTable.cbl. Right click the source code and choose 'Run As->1 COBOL Application' :



If you have set up the database and table you should see a SQL Code 100 in the console window:



This confirms the program will work correctly after it is exported and run under EBP.

To export this application so it is ready to deploy to EBP later on, right click the ListTable project and choose 'Export':

	Export	
elect		-7-
Package Elastic Co	OBOL project resources for deployment	25
Select an export v	wizard:	
type filter text		
🕨 🧁 General		
🕨 🧁 EJB		
🔻 🗁 Elastic COE	BOL	
🧧 Elastic C	COBOL Deploy Wizard	
😰 Elastic T	Transaction Platform Deploy Wizard	
🕨 🗁 Install		
🕨 🗁 Java		
🕨 🗁 Java EE		
🕨 🗁 Plug-in De	velopment	
Remote Sy	stems	
Run/Debug	9	
Tasks		
Team		
Web		
Web Servic	es.	

Choose Elastic COBOL Deploy Wizard and click Next:

000	COBOL Deploy Wizard
Archive Package Specification Define what resources to package	, ē
Resources to export:	
[ListTable] listtable\$Wrk.class [ListTable] listtable.class [ListTable] listtable.java.smap	Add Remove
Deployment Options	
Select the export destination	: customers/DEMO/LISTTABLE.jar Browse
Specify the deployment licen	se location: Browse
Options: Compress the contents of the ar Overwrite existing files without w	chive warning
?	< Back Next > Cancel Finish

Fill in the name of the file to export (seen here is /Users/ianw/LISTTABLE.jar) NOTE - The filename should be all lower or all upper case.

Click 'Add..' at the top to add the SQL driver to the export.

Open the workspace/project and workdir folder and select the postgresql-9.2-1003.jdbc4.jar, then click OK:

000	COBOL Dep	oloy Wizard
Archive	Please select resources to add to the list	
Define v Resource [ListTak [ListTak [ListTak	<ul> <li>Vorkspace</li> <li>Vorkspace</li> <li>ListTable</li> <li>cobol_source</li> <li>copylib</li> <li>copylib</li> <li>intermediate_java_code</li> <li>java_source</li> <li>listing</li> <li>resources</li> <li>cowrkdir</li> <li>RemoteSystemsTempFiles</li> </ul>	✓  postgresql-9.2-1003.jdbc4.jar ↓ test.jcl
Deploy Tr	?	Cancel OK

This will add the driver to the export list:

Resources to export:	
[ListTable] listtable\$Wrk.class	Add
[ListTable] listtable.class	
[ListTable] listtable.java.smap	Remove
[ListTable] workdir/postgresql-	9.2-1003.jdbc4.jar
	▲
	<b>T</b>

Click Next and select ListTable.cbl as the Run Configuration:

000		CC	BOL Deploy Wiza	ard		
Archive Pac	kaging Options					8
Define requi	red runtime eleme	nts and ot	ner options			
Soloct runtin	no elemente requir	od by ann	ication			•
Flasti	c COBOL Runtime I	ed by app Flements	Core Runtime F	lements (Requi	red 💼	
C) Liast		crements	Printing API	alementa (kequi		
			File Protocol Su	pport	ort	
			JCurses Suppor	t (Platform Dep	enc	
			XML Support			
			Phylronment Si	innort	_	
🗹 Inclu	de the selected rur	ntime elem	ents within the exp	ort archive		,
Colored allow and						
Select the set	tings to use for lai	unching th	e application:			
Run Configu	ration: ListTable	e.cbl		÷		
Colored alternation						
Select the cla	ss of the applicatio	on entry p	oint:			
Main Class:	listtable				Browse	)
Other Option	s (JAR only):					
Copy	JAR dependencies	from the E	astic COBOL directo	ory to the expo	rt directory	
Gener	ate script-based a	pplication	launcher stub			
	Generate WAR pag	kage fron	IAR and HTML stub			
	Use Java plugin		,, u	-		
	ose java plugin					
?		< B	ck Next >	Car	ncel	Finish

Click Finish, the jar file will be created.

Minimize the ListTable project and close the ListTable.cbl source file:

000	workspa
🗗 - 🛛 🕼 🗳 🔪 🕩 🛛 🗉 🕅 3. 3. 12 🗮 👼 😿	ि 🚳 🐼 🍣 🎇 🐄 🖸 र 🂁 🖏 😵
Project Explorer 🕱 📄 🔽 🗖 🗖	
▶ 🖄 ListTable	

Go to Eclipse - > Preferences, Elastic COBOL/EBP Server settings:

000	Preferences	
type filter text	EBP Server Settings	(> + ⊂> + ▼
Code Recommenders Code Recommenders Data Management Elastic COBOL COBOL Editor COBOL IDE Fonts Debug EEP Server Settings	EBP Server Url: http://cicsdemo.apps.heirloomcomputing.com:8080/eb Refresh: 5001 ms (EBP Server Refresh rate should be > 9 Restore Defaults	p 5000ms) Apply
▶ Heln		

Enter your server URL, set the Refresh rate to be 5001 and click Apply. From the tools menu bring up the EBP server:



## Deploy:

In order to deploy the jar file to the instance we will use the file explorer built into the platform.

Open a webpage to the home page of the application instance:



Heirloom PaaS deployment instance: cicsdemo.apps.heirloomcomputing.com Instance: Secure | File Explorer | DB Explorer | Tomcat | EBP JES/JCL Console | EBP JES/JCL Configuration External: Heirloom Computing | Heirloom PaaS | Support Copyright (c) 2017 Heirloom Computing, Inc.

Click the File Explorer link and sign in with your PaaS Portal credentials:

e	Xtploi	rer
Login		
Username:	demo	
Password:	••••••	
Language:	English 👻	
	Login	Reset

Click the 'Data' directory:

Directory							
🛧 Home 🍰 Reload 🔍 Search   🛃 🕎 🗋	in 🔀 🕏	0 🛯 🗉	🖬 🖄 🗐 🏦 🛯	Show Dire	ctories		
Name	Size	Туре	Modified	Perms	Owner		
🛅 data	4 KB	Directory	2015/05/07 11:42	777 (rwxrwxrwx)	n/a		
🛅 geronimo	4 KB	Directory	2012/02/09 07:31	777 (rwxrwxrwx)	n/a		
🛅 h2	4 KB	Directory	2015/05/07 11:59	755 (rwxr-xr-x)	n/a		

Click the upload icon to bring up the file upload dialog:



Upload file(s)		3
flashupload standa	ardupload Transfer from another Server	
	Maximum File Size = 200 MB Maximum Upload Limit = 8 MB	
File 1:		Browse
File 2:		Browse
File 3:		Browse
File 4:		Browse
File 5:		Browse
File 6:		Browse
File 7:		Browse
Overwrite existing file(s)?:		
		Save Cancel

Click the 'Browse' button next to the 'File 1:' field and select your LISTTABLE.jar file, then click Save:

Directory						
👚 Home 🏂 Reload 🔍 Search 🛛 📴 🖺 🗅 🗶 📲 🎦 💷 🚍 🖄 🏦 🧐 🛞 😵 🧕 Show Directories						
Name	Size	Туре	Modified		Perms	Owner
🛅 spool	4 KB	Directory	2014/10/19 19:25		777 (rwxrwxrwx)	n/a
🛅 tape	4 KB	Directory	2014/10/19 19:25		777 (rwxrwxrwx)	n/a
HCI00.TESTALL.LOG	28 B	File	2015/05/07 11:42		644 (rw-rr)	n/a
🗐 LISTTABLE.jar	4.15 MB	Jar	2017/08/11 15:22		644 (rw-rr)	n/a

You are now ready to test the jar file using JCL.

Switch back to Eclipse and expand the ListTable project and its workdir folder:



Right click the 'test.jcl' and click 'Submit JCL':

🚁 postgresq।– ५.४	2-1003.jdbc4.jar		
G test.jcl ▶ 🚰 onlinedemo	New	•	
▶ 🐼 onlinedemo-deploy	Show In Open Open With	C₩W ► F3 ►	
	<ul> <li>Copy</li> <li>Copy Qualified Name</li> <li>Paste</li> <li>Delete</li> <li>Remove from Context</li> <li>Mark as Landmark</li> <li>Build Path</li> <li>Move</li> <li>Rename</li> </ul>	第C 第V 区 て 合 第↓ て 合 第 ↑ ト	
	≥ Import ≥ Export		
	Refresh	F5	
	Validate Show in Remote Systems Run As Debug As Profile As Database	view	Problems 📃 JOB ID
	Build Program	r	
	Submit JCL		Submit JCL
	Team Compare With Replace With	* * *	
	Properties	жі	

This will send the JCL to EBP, click the EBP Server tab:

Rroblems	i 📃 Console 🚹 EE	BP Server 😫								
谷 JOB ID	JOB NAME	CLASS	TYPE	USER	STATE	SUBMITTED	INITIATED	COMPLETED	TIME	MESSAGE
▶31756	TEST01	A	JCL	EBP	TERMINATED	2017-08-13T0	2017-08-13T0	2017-08-13T0	0	WITH CC 0

The tab will refresh until you see the job listed and you will see a Message of 'WITH CC 0' which means the job completed correctly.

Click the left hand arrow to drop down the outputs of the job:

R Problems	📃 Console 🚹	EBP Server 😫		
谷 JOB ID	JOB NAME	CLASS		
▼31756	TEST01	A		
OUT317	56			

Double click the OUTnnnn file and it will open up in Eclipse:

```
■ JOB:31756::MSG SS
  1 2017-08-13 03:12:32 HEIRLOOM COMPUTING INC - ELASTIC BATCH PLATFORM - JES/JCL (v17.6.26)
 3 2017-08-13 03:12:32 $HASP1111 JOB 31756 TEST01 STARTED
                                                                                            ON CLASS A
 4 2017-08-13 03:12:32 $HASP117I JCL SCAN START
                                        5 2017-08-13 03:12:32 $HASP118I
  6 2017-08-13 03:12:32 $HASP118I
                                        //TEST01 JOB (HCIACCT), 'EBP Table Test', CLASS=A, MSGCLASS=A,
                                                        MSGLEVEL=(2,2),NOTIFY=USERID
 7 2017-08-13 03:12:32 $HASP118I
                                        11
                                        //STEPID01 EXEC PGM=ListTable
 8 2017-08-13 03:12:32 $HASP118I
 9 2017-08-13 03:12:32 $HASP118I
                                         //SYSIN DD DUMMY
 10 2017-08-13 03:12:32 $HASP118I
                                        //SYSPRINT DD
                                                        SYSOUT=*
 11 2017-08-13 03:12:32 $HASP118I
                                         //SYSOUT DD SYSOUT=*
 12 2017-08-13 03:12:32 $HASP118I
                                         //SYSERR DD SYSOUT=*
 13 2017-08-13 03:12:32 $HASP117I JCL SCAN END
 14
 15 2017-08-13 03:12:32 $HASP125I JOB ANALYSIS START
 16 2017-08-13 03:12:32 $HASP126I
                                            JOB ATTRIBUTE NAME: TEST01
 17 2017-08-13 03:12:32 $HASP126I
                                            JOB ATTRIBUTE CLASS: A
 18 2017-08-13 03:12:32 $HASP126I
                                            JOB ATTRIBUTE ACCOUNT: HCIACCT
                                           JOB ATTRIBUTE PROGRAMMER: EBP Table Test
 19 2017-08-13 03:12:32 $HASP126I
 20
 21 2017-08-13 03:12:32 $HASP126I
                                             STEP ATTRIBUTE NAME: STEPID01
 22 2017-08-13 03:12:32 $HASP126I
                                            STEP ATTRIBUTE PGM: ListTable
 23 2017-08-13 03:12:32 $HASP126I
                                            DD ATTRIBUTE NAME: SYSIN
 24 2017-08-13 03:12:32 $HASP126I
                                            DD ATTRIBUTE NAME: SYSPRINT
 25 2017-08-13 03:12:32 $HASP126I
                                            DD ATTRIBUTE NAME: SYSOUT
 26 2017-08-13 03:12:32 $HASP126I
                                            DD ATTRIBUTE NAME: SYSERR
 27
 28 2017-08-13 03:12:32 $HASP125I JOB ANALYSIS STOP
 29
 30
 31 2017-08-13 03:12:32 $HASP135I JOB 31756 TEST01 STEP STEPID01
                                                                         STARTED
 32 2017-08-13 03:12:34 $HASP136I JOB 31756 TEST01 STEP STEPID01
                                                                         TERMINATED
                                                                                            WITH CC 0
 33
                                                                                            WITH CC 0
 34 2017-08-13 03:12:34 $HASP112I JOB 31756 TEST01 TERMINATED
 35 2017-08-13 03:12:34 =
 36 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSPRINT'
 37 2017-08-13 03:12:34 ==
 38 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSOUT'
 39 name: ian
                        : 123456
 40 name: IAN
                         : 512-555-1234
 41 name: BOB
                         : 408-333-2222
 42 sqlcode: 000000100+
 43 End of Program.
 44 2017-08-13 03:12:34 ===
 45 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSERR'
 46
```

Congratulations!

You just configured, built, deployed, and tested a CICS application, and a JCL/JES batch application to the Heirloom PaaS