# 3. Building the sample CICS project for Azure

This assumes you have built the sample EBP project for Azure and have an open, working Eclipse that looks similar to this:



## Building the ETP project

Right click the Project Explorer and choose 'New->Project':

000 Ne	ew Project
Select a wizard Create a COBOL Project	
Wizards:	
type filter text	
<ul> <li>Java Project</li> <li>Java Project from Existing Ant Builds</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>Java Script</li> <li>JAXB</li> <li>JPA</li> </ul>	file
?	Next > Cancel Finish

Choose 'COBOL Project' and click Next:

000	Now
000	
COBOL Project	
Create a new CO	BOL project.
Project name:	onlinedemo
-	
🗹 Use default	location
Location: /Use	ers/ianw/Documents/workspace-azure/onlinedemo Browse
Choo	ose file system: default ‡
	Project type Project directories
IBM EBC	DIC
MICTO FO	
Flastic T	iransaction Platform Template
Elastic T	ransaction Platform WAR Template
►Samples	
►Tutorials	
Diagnostic	
Details	
This project Platform pro Tomcat, Jet template fo Java EE jars no source f	t template contains compiler settings for compiling Elastic Transaction ojects and packaging them into .war files suitable for deploying to ty or other servlet container systems. It is identical to the standard ETP r deploying .ear files into Java EE servers except it references additional that are not included in Tomcat required by ETP. This project includes iles.
?	< Back Next > Cancel Finish

Name the project online demo, expand the 'Compatibility' option under 'Project type' and choose 'Elastic Transaction Platform WAR Template'.

Click Finish and click Yes to open the Elastic COBOL perspective:

Project name	e: onlinedemo
🗹 Use defa	ult location
Location: /	Users/ianw/Documents/workspace-azure/onlinedemo Browse
000	Open Associated Perspective?
Remembe	er my decision
	No Yes
IBM E	BCDIC
Micro	> Focus
Details	

Expand the onlinedemo project:



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



You'll note at this stage there are errors in the compile, we'll fix this with additional source files and steps:



Drag the dclgenf.cpy file into the copy lib folder, choosing 'Copy files' again. Right click the resources folder and 'New->Other' :

<ul> <li>Project Explorer ☆ 📴 Outling</li> <li>▶ 29 ListTable</li> <li>♥ 30 onlinedemo</li> <li>▶ 28 java_source</li> <li>▶ 28 intermediate_java_code</li> </ul>	e □ □			
➢ resources ► ▲ Referenced Libraries	New			📸 Project
▶ 🛋 JRE System Library [Java	Go Into			🗳 COBOL Project
<ul> <li>cobol_source</li> <li>copylib</li> <li>listing</li> </ul>	Open Type Hierarch Show In	y ∖⊂≋w	F4 ▶	S ETP Project
≥ workdir	<ul> <li>Copy</li> <li>Copy Qualified N</li> <li>Paste</li> <li>Delete</li> </ul>	ame	жс жv	<ul> <li>COBOL Copyfile</li> <li>COBOL Program</li> <li>ETP Deploy Settings</li> <li>JCL</li> </ul>
	Remove from Co Build Path Source Refactor	ntext TA T#S T#T	↓ ₩1 ↓	<ul> <li>Annotation</li> <li>Class</li> <li>Enum</li> <li>Interface</li> </ul>
	≥ Import ≥ Export			₽ Package ₽ Source Folder
	Refresh		F5	📸 Example
	Run As Debug As Profile As Validate Restore from Local I Maven PyDev Team Compare With	History	* * * * * * *	<mark>≌ Other </mark>
	Properties	2. Problems	жı	

Select General->File and click Next:

🎦 Project Explorer 🔀 📴 Outlin	e 🛛 🗖			
<ul> <li>ListTable</li> <li>Onlinedemo</li> <li>Java_source</li> <li>Image: intermediate_java_code</li> </ul>	<b>□</b> ∰ ♥ ▼			
Referenced Libraries	New		•	📸 Project
🕨 🛋 JRE System Library [Java	Go Into			🖄 COBOL Project
<ul> <li>cobol_source</li> <li>copylib</li> <li>listing</li> </ul>	Open Type Hierarch Show In	y ∖⊂≋w	F4 ▶	SETP Project
🦢 iisting	<ul> <li>Copy</li> <li>Copy Qualified N</li> <li>Paste</li> <li>Delete</li> </ul>	ame	жс жv	<ul> <li>COBOL Copyfile</li> <li>COBOL Program</li> <li>ETP Deploy Settings</li> <li>JCL</li> </ul>
	Remove from Co Build Path Source Refactor	ntext TA T#S T#T	↓#{       	<ul> <li>Annotation</li> <li>Class</li> <li>Enum</li> <li>Interface</li> </ul>
	≥ Import ≥ Export			₽ackage Source Folder
	Refresh		F5	📸 Example
	Run As Debug As Profile As Validate Restore from Local I Maven PyDev Team Compare With	History	*** ***	➡ Other 第N
_	Compare with		•	
	Properties		<b>%</b> I	

Name the file ecobol.dir and click Finish.

Drag the ecobol.dir from your source bundle into the resources folder. Choose to 'Copy files' and confirm 'yes' for the file replacement:

	works	space-azure - Elastic COBOL - onlinedemo/resources/ecobol.dir - Eclipse
9		onlinedemo/resources/ecobol.dir exists. Do you wish to overwrite?
di	-1	
		Yes To All     No     Cancel     Yes

Click Yes to perform a project rebuild. Close the ecobol.dir window.

Drag the onldemo.bms file into the resources folder and choose to 'Copy files'.

Right click the onldemo.bms file and select 'Elastic COBOL BMS->Generate Copyfile' :



Right click the onlinedemo Project and choose properties, then click 'Java Compiler'.

Check 'Enable project specific settings' and choose 1.6 and the Compiler compliance level:

000	Properties for onlinedemo	
type filter text	Java Compiler	⟨⊐ • □⟩ • ▼
▶Resource Builders	✓ Enable project specific settings	Configure Workspace Settings
Builders COBOL Compiler Settings COBOL Runtime Settings Java Build Path Java Code Style Java Compiler Javadoc Location Project Facets Project References Run/Debug Settings Task Repository Task Tags Validation WikiText	<ul> <li>Enable project specific settings</li> <li>JDK Compliance</li> <li>Use compliance from execution environment of Compiler compliance level:</li> <li>Use default compliance settings</li> <li>Generated .class files compatibility:</li> <li>Source compatibility:</li> <li>Disallow identifiers called 'assert':</li> <li>Disallow identifiers called 'enum':</li> <li>Classfile Generation</li> <li>Add variable attributes to generated class files</li> <li>Add source file name to generated class file (assert)</li> <li>Preserve unused (never read) local variables</li> <li>Inline finally blocks (larger class files, but imp</li> <li>Store information about method parameters (block)</li> </ul>	Configure Workspace Settings on the 'Java Build Path' 1.6 ‡ 1.6 ‡ 1.6 ‡ Error ‡ Error ‡ Error ‡ s (used by the debugger) s files (used by the debugger) used by the debugger) used by the debugger) used by the debugger)
	When selecting 1.6 compliance, make sure to a activated (currently 1.8). Configure the <u>'Installe</u> Environments', or change the JRE on the <u>'Java B</u>	have a compatible JRE installed and ed JREs' and ' <u>Execution</u> Build Path'. Restore Defaults Apply
?		Cancel OK

Click OK, then click Yes to perform a project rebuild.

### Building the deployment project

Right click the Project Explorer and select New->Project...: Choose Elastic Transaction Platform/ETP Project and click Next:

COBY BEIIBUCCI			
000	New Project		
Select a wizard Create an Elastic Transaction Platfor	m Project		
Wizards:			
type filter text			
Java Project Sava Project from Existing Ant Plug-in Project General Celipse Modeling Framework Elipse Modeling Framework Elipse Modeling Framework COBOL Project COBOL Project Elastic Transaction Platform ETP Project Dava Java EE Java EE Java Script DavaScript DavaScript	Buildfile		
?	Next >	Cancel	Finish

Name the project 'onlinedemo-deploy' and click Finish.

000	New
Elastic Tra Create a n	ew ETP project.
Project na	me: onlinedemo-deploy
🗹 Use de	fault location
Location:	/Users/ianw/Documents/workspace-azure/onlinedemo-deplo Browse
	Choose file system: default ‡
?	< Back Next > Cancel Finish

Expand the onlinedemo-deploy project:



Drag the lib folder from your source bundle onto the WEB-INF directory and choose to 'Copy files and folders':

	ould be imported in	nto the project:	
$\odot$ Copy files and folders			
O Link to files and folder	;		
Link to files and recrea	te folder structure w	vith virtual folde	rs
Create link location	s relative to: PRO	JECT_LOC	÷
onfigure Drag and Drop Sett	ings		
oningure brag and brop set			

Expand the lib folder and your project structure should look like this:



Drag your license file into the WEB-INF folder, choosing to 'Copy files':



In the Elastic Transaction Platform Deploy settings Editor set the SYSID to ONLD and check the onlinedemo project:

Click Apply. Click the Transactions category and add two transactions, ONLD and INIT both pointing to the program onlinedemo:

General	Transactions			
Programs				
Transactions	Transaction ID	Program ID	UCTRAN	Add
SQL	onld	onldemo	false	
HTML Templates	init	onldemo	false	Edit
Journals				
Queues				Remo
Spools				
JMS				
Program Control				T
File Control				
IMS DB				
Date/Time				
Counter				
Operator				

Click the SQL category and click Add...:

[	000	Edit dialog	
	Connection name:	default	ŀ
	Datasource:		ŀ
	Username:	ianw	L
	Password:	Onlinedemodb1	
	Connection URL*:	jdbc:sqlserver://onldemosr	P
	Driver classname:	com.microsoft.sqlserver.jdb	Ŀ
on	Auto Commit:	False ‡	on I
	Isolation Level:	Repeatable Read \$	dat
	Catalog:		
	Read Only:	False ‡	
	*Only re	quired if no datasource is given	
		Cancel OK	

Fill in username,password, the connection URL (the same URL you used for the ListTable.cbl code and the driver classname as "com.microsoft.sqlserver.jdbc.SQLServerDriver" (no quotes). Click OK.

Click Apply.

You are now ready to export the converted CICS application. Right click the onlinedemo-deploy Project and select Export...:

n	O O O Export
L m rr eu	Select Package ETP resources for deployment
ig (i te ur erci A S S io	Select an export wizard: type filter text Ceneral Cene
Г	Cancel Finish

Select the 'Elastic Transaction Platform Deploy Wizard' and click Next:

gra	ams
M	Archive Package Specification
eu	Define what resources to package
g	Select the ETP deploy project to use:
l e	
ur -	Deployment Options
ci A	• Traditional O Cloud
S S	Select the export destination:
0	FAR/WAR file: //Isers/ianw/onlinedemo.war Rrowse
L	EAR/WAR ne. / Users/nanw/onintedento.wai
l	
L	
l	
l	
I	Specify the deployment license location:
L	Browse
L	biowse
I	Ontions:
	Compress the contents of the archive
L	Overwrite existing files without warning
H.	
1	Include Project .smap hies for debugging
n	
W	
1	
"	(?) < Back Next > Cancel Finish

Fill in the name of the war file (in this case it is /Users/ianw/onlinedemo.war) and click Finish:

Calact the own	Archive Export Problems
Export fir	shed with warnings or info messages:
	Details >> OK

# Click OK:

000	O Export Successful
1 🕦	The export was successful.
	The application when deployed will be available as:
	http:// <host>:<port>/<sysid (default:="" dflt)="">/<servlet (default:="" <br="" pattern="" url="">servlet)&gt;</servlet></sysid></port></host>
	For example:
	http://127.0.0.1:7001/dflt/servlet
	ОК
-	Browse

Click OK. The war file is now ready for deployment