1 Building, Deploying and Testing DPES application

This chapter provides updated instructions for accessing the sources code, developing, building and deploying the DPES application in the user environment. DPES development environment is split into two part 1. Backend DPES webserver – Maven Java Project and 2. DPES Frontend Adobe Flash UI project using flex 3 SDK. The following instruction covers both development environments.

1.1 Development Environment

This section is useful for developers who wants to develop the DPES tools by extending some features which are already exists in the project or create some additional new features based users need. Source codes are freely available to the developer community at TIMBUS open source repositoryⁱ. It is assumed that the developer would have sufficient knowledge on Java, Maven, Eclipse, Version controls such as Git and eGit and flex development. Hence the scope of this document is restricted to detailing DPES application only rather than providing information for setting up the above said technologies.

1.1.1 DPES Webserver System Requirements

- Java Development Kit (JDK) (version 1.7_17 or above)ⁱⁱ
- Apache Maven plugin for Eclipse (version 3.2.3)ⁱⁱⁱ
- Eclipse IDE for Java EE Developers (version Juno or later)^{iv}
- Tomcat Web Servlet Container (version 7 or above)^v
- EGit plugin for eclipse^{vi}
- H2 data base for DPES registry (version 1.3.173)^{vii}

1.1.2 DPES UI application System Requirements

Adobe Flex Development Tools

- Flex builder 3 standalone or plugin to eclipse
- Flex SDK 3.2

Supported Operating Systemsviii

- Microsoft[®] Windows[®] 7/8 (32/64 bit), Windows Vista[®] (32 bit), or Windows XP Professional with Service Pack 2 or 3
- Mac OS X v10.5.6–10.6 (Intel[®] only)

Java Virtual Machine

• Java[™] Virtual Machine (32 bit), IBM[®] JRE 1.5, IBM JRE 1.6, Sun[™] JRE 1.5, Sun JRE 1.6

Supported Runtimes

- Adobe[®] Flash[®] Player 11 Web browser plugin
- Adobe AIR[®] 3 standalone UI application

1.2 Requirement for Deploying DPES application

The following software packages, services and resources are needed for successfully deploying and testing of DPES application. DPES is integrated with other TIMBUS tools such as PI Preservation Identifier, iERM/Risk Management tools, Ontologies and Context Model Instances, In order for DPES to function properly the following resources and services must be available on the service end point described in the API implementation section (reference-add all the ontology lo curls, PI service url).

1.2.1 TIMBUS Services requirements

- iERM Service or Risk Management Tools to Invoke preservation action by sending Risk docs
- PI Preservation Identifier Service to provide alternate preservation recommendation.
- Ontologies for business process, DIO, DSO, CUDF, etc.
- Context Model Instances for the specific business process that are to be preserved.

1.2.2 DPES Runtime Environment Requirement

- Java Run Time JRE-JVM (version 1.7 or above)
- Apache Tomcat Server (version 7 or above)

1.2.3 DPES Application Bundles

- DPES Webserver War bundles file
- DPES Flash UI Deployment package

1.2.4 DPES Infrastructures Services

- H2 Database embedded server
- Amazon AWS S3 account to configure storage credentials
- Network connectivity to access to TIMBUS Services and Ontology
- KVM+QEMU Virtualisation Server (PROXMOX)

1.3 Building of DPES Application, Installation and Testing

Follow the below instruction to download DPES source code, compile and deploy them into you own test environment.

1.3.1 Building of DPES UI Flex 3 Project

Check out project source using following git command from your preferred command line interface

git clone https://@opensourceprojects.eu/git/p/timbus/dpes/preservation-console DPES_Flex_UI

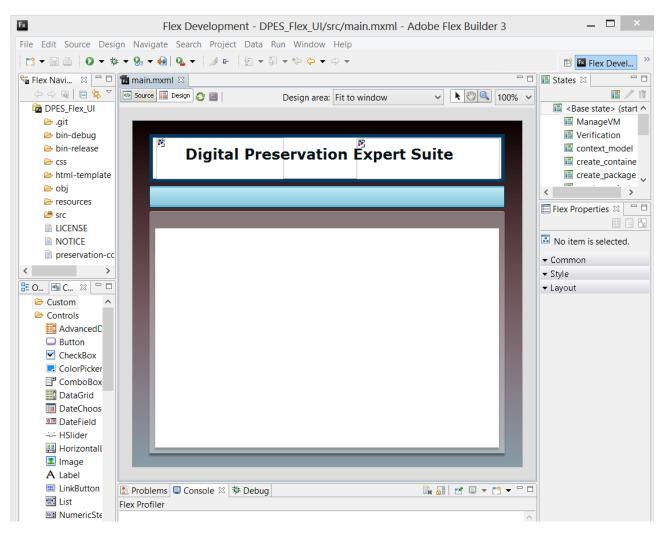


Figure 1 : Flex Builder with DPES UI Project

Launch the flex builder or the Eclipse IDE with Flex builder plug-in installed. Import the project into the IDE from the folder (DPES_Flex_UI) location where project was cloned. Once the project is imported into the IDE, developers can be able to browse the code, make modification and do testing within the IDE. Flex development environment shown in **Error! Reference source not found.** displaying the main.mxml file in design view, flex builders provides graphical design environment to drag and drop the widget and tools to

build the rich user interface and the source code will be generated automatically that can be viewed by switching to source tab. Developers needs to make additional code changes at the source code to make new UI widgets to work. This particular UI development need programming skills in Action Script 3 and MXML development. The scope of the document is limited to provide detailed tutorials on above said technologies. However, Developers can learn more about flex development respective official website.

Once the project is loaded into the IDE, we needs to build the source code and export as release build for deploying at DPES server.

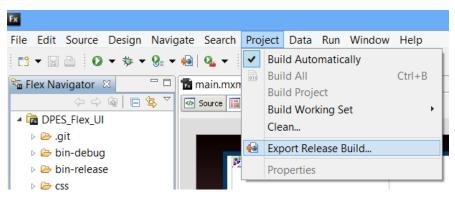


Figure 2 : DPES UI Project Export

In order export the release build of this UI project navigate to Project menu, enable build automatically check box and click Export Release Build option as shown in the screen capture **Error! Reference source not found.**. Follow the export release build dialog box as shown in the Figure 3 and specify main.mxml as main application and folder location where release build will be saved.

Fx		Flex Development - DPES_	Flex_UI/
File Edit Source Design Navi	gate Search	Project Data Run Window Help	
□ - 🗟 📄 💿 - 🎄 - 🚱 -	😔 🛛 🖌 🗍	∥ ⊨ ½ ▼ % ▼ \$ ↓ ↓ ▼	
😼 Flex Navigator 🛛 🗖 🗖	💼 main.mxml	×	
수 수 @ 🗖 🔩 🏹	Source 🛅 🕻	Design 📀 📾 State: <base state="" td="" 🗸<=""/> <td></td>	
DPES_Flex_UI			
🗁 .git	Fx	Export Release Build — 🗖	×
🗁 bin-debug	-		
🗁 bin-release	Export Rele	ase Build	v
🗁 css	Files alread	ly in the export destination folder may be overwritten.	V
🗁 html-template			
🗁 obj			
🗁 resources	Project:	DPES_Flex_UI	~
i interest in the second seco	Application:	main.mxml	~
NOTICE	10		
preservation-console.as	View source		
< >	Enable vi	ew source	
🗄 Outline 🛃 Compo 🙁 🗖 🗖	Choose S	ource Files	
Custom ^			
AdvancedDataGrid	Export to fol	der: bin-release Browse	
Button		(in /DPES_Flex_UI)	
CheckBox			
DataGrid			_
DateChooser	?	< Back Next > Finish Cancel	
3 DateField			
No LICEARS			

Figure 3 : Export DPES UI Release Build

1.3.2 Building of DPES Webserver application project

Check out the maven java project from TIMBUS open source repository into eclipse either by using git command line interface or using eGit plugin with Eclipse. The following git command shows cloning of project using command line interface.

Git clone <u>https://@opensourceprojects.eu/git/p/timbus/dpes/preservation-manager</u> preservation-manager

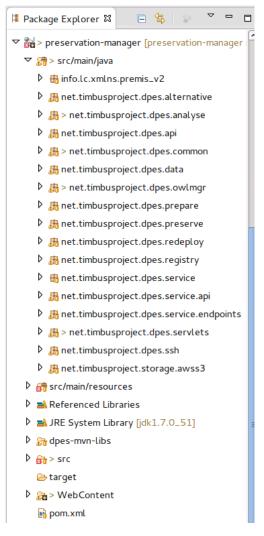


Figure 4 : DPES Server Package lists from Eclipse

After cloning the complete source code developers can import the code into eclipse using project -> import option and selecting project type as maven. Eclipse will scan and automatically pick up the pom.xml file which is located at the root of the project as shown in the Figure 4, this figure also shows existing the package list in the eclipse package explorer window. There is a config.properties file located at **preservation-manager/src/main/resources/config.properties** location that needs to be updated to reflect your existing deployment environment. A screen capture shown in the Figure 5 details the list of parameters to be updated in the config.properties file. Almost all the values highlighted in the blue text have to be replaced by current live information to successfully run all the features of the DPES application.

```
📄 *config.properties 🛛
```

```
#H2 database Server details
driverName:org.h2.jdbcx.JdbcDataSource
connectString:idbc:h2:WebContent/database/dbtimbuslocalReg
userName:sa
userPass:sa
#DPES Server details for Email Notification
mailServer:smtp.gmail.com
mailPort:587
mailUser:dpes.timbus@gmail.com
mailPwd:mail-password
#PROXMOX Server IP address
redeploy server ip address:proxmox-ip-address
redeploy server userid: root
redeploy server password:proxmox-root-pw
redeploy server authentication:pam
redeploy server node:type-node-name
# Configure proxy settings
http proxyHost:specify-proxy-server
http proxyPort:proxy-port
socks proxyHost:specify-socks-server
socks proxyPort:type-socks-port
proxy_enable:specify-yes/no
#admin credentials to perform verification
net admin user:type-admin-username
net admin pwd:type-your-password-here
```

Figure 5 : DPES Configuration Properties

Go to the command line interface, go to the project root directory and type the *mvn package* command to download all third party libraries and dependencies jars from the maven repositories and perform packaging. Finally from Eclipse, choose export and select type as *WAR* file and then type file name folder location to save the DPES web application file as shown in **Error! Reference source not found.**.

NOTE: Several alternate methods are available for all or some of the installation steps. For example any command executed through a GUI interface may also be possible to be executed through a command line interface. We have outlined above the methods popularly utilised by TIMBUS developers.

	c/main/java/net/timbusproject/dpes/preserve/Packagerjava actor Navigate Search Project Run Window Help	- Éclipse	Disport
 ○ • ○ ○ ○ ○ • ○ ○ ○ • ○	ect	Image: State	WAR Export Export Web project to the local file system.
desWebapn preservation Preservation Preservation Servers	OKIL Named Individu Go Into - Open in New Window - Open Type Hierarchy F4 Show In All+-Shift+W + Copy Qualified Name - Paste Cbi+V	<pre>el newIndividual = factory.getONLUmmedIndividua masertims sesterims = factory.getONLUmealForgenYII(RL.create = factory.getONLUmealForgenYIII); sertionAxiom nameAssertionAx = factory.getONLD x = new AddAxiom(ontology, nameAssertionAx); ge(addImaex); ship property ertion axioms</pre>	Web groject: dpesWebapp Destination: C\dpes\dpeswar Iarget runtime Optimize for a specific server runtime Apache Tomcat v7.0
۵. کت	Delete Delete Remove from Context Ctrf+Ab+Shift+Down Build Path - Source Abt+Shift+Shift+Remove Refactor Abt+Shift+Shift+The Import Eport Chift Patient -	sserion to individual document ent format = manager.getOntology/format(ontology); ogy(ontology, format, IRI.create(new File("Web n	Export source files Overgrite existing file
	Build Project Refresh F5	rFactoryConfigurationError rException	

Figure 6 : Export DPES WAR file

1.3.3 Deploying Web application War file

To install the DPES web application into the Tomcat server, go the Tomcat admin URL (TIMBUS DPES Test Bed URL: <u>http://134.191.240.68:16080</u>) and click *manager app* button as shown by red arrow in the **Error! Reference source not found.** and then click choose file under Deploy ->WAR file to deploy.

Use the application tab from the Tomcat admin console to provision (stop, start, reload and undeploy) the DPES app and monitor the status. Use the following TIMBUS Test Bed URL to browse the application.

• DPES Development site: <u>http://134.191.240.68:16080/dpesui/</u>

Home	Documentation	Configuration Examples Wil	ki Mailing Lists	i.		Find Help
Apac	he Tomcat/7.0).39		*	Apache	Software Foundation //www.apache.org/
X	Reco Secu Mana	re seeing this, you've succ ommended Reading: urity.Considerations HOW-TO ager Application HOW-TO tering/Session Replication HOW		illed Tomca	t. Congrati	Server Status
Iomcat	oper Quick Start Setup th Application	Realms & AAA JDBC Data Sources	Examples			et Specifications at Versions
ost-manager	None specified	Tomcat Host Manager Application	n	true	Q	Start Stop Reload Undept Expire sessions with idle 2 30
sanager	None specified	Tomcat Manager Application		true	1	Start Stop Reload Undeploy Expire sessions with idle 2 30
eploy						
ploy directory o	r WAR file located on	server				
		Context Path (required):				
		XML Configuration file URL:				
		WAR or Directory URL:				
			Deploy			
AR file to deploy	í.		П			
		Select WAR file to uploa	d Choose File	io file chosen		
			Deploy			

Figure 7 : Deploy DPES WAR file into Tomcat Server

1.3.4 Deploying DPES Flash UI to the Server

To deploy the DPES flash UI project copy the contents under bin-release folder and contents under resources from flex build IDE to tomcat directory .../webapps/dpesui/ and .../webapps/resources/ respectively. Directory locations are shown in the figure 8 and 8. Once the files are copies stop the tomcat server and restart, then goto the web browser anf type http://host-ip:8080/dpesui you will be served with dpes startup page.

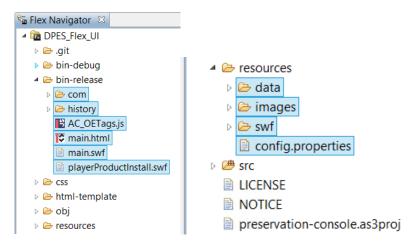


Figure 8 : Deploy DPES WAR file into Tomcat Server

Server directory locations for flash UI deployment

	/nome/timbi	us/WebServers/	dev-apache/	/webapps/dpes	sui	
		gs mp ebapps dpes dpesui 2 com				
		2 history				
Filename	n		Filetype	Last modifi	Permissi	Owner/G
Filename	n		Filetype	Last modifi	Permissi	Owner/G
. .	uctinstall.swf	Filesize		Last modifi 10/27/2014		
. .	uctinstall.swf	Filesize	Shockwa		-rw-rw-r	timbus ti
📕 📄 playerProd	uctinstall.swf	Filesize 657 838,847	Shockwa	10/27/2014 10/27/2014	-rw-rw-r -rw-rw-r	timbus ti timbus ti
). playerProd main.swf		Filesize 657 838,847	Shockwa Shockwa HTML D	10/27/2014 10/27/2014	-rw-rw-r -rw-rw-r -rw-rw-r	timbus ti timbus ti timbus ti
) playerProd main.swf main.html		Filesize 657 838,847 4,215	Shockwa Shockwa HTML D	10/27/2014 10/27/2014 10/27/2014 10/27/2014	-rw-rw-r -rw-rw-r -rw-rw-r	timbus ti timbus ti timbus ti timbus ti

Figure 9 : DPES UI Binary Deployment

ⁱ TIMBUS open source project repository <u>https://opensourceprojects.eu/p/timbus/wiki/Home/</u>

ⁱⁱ Java Development Kit - <u>http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-</u> <u>1880260.html</u>

- Apache Maven Project <u>http://maven.apache.org/</u>
- ^{iv} Eclipse IDE download <u>https://www.eclipse.org/downloads/</u>
- ^v Tomcat Java Serverlet <u>http://tomcat.apache.org/</u>
- vi EGit Eclipse Team Provider <u>http://www.eclipse.org/egit/</u>
- ^{vii} H2 Relational database engine <u>http://www.h2database.com/html/main.html</u>
- viii Flex builder requirement <u>http://www.adobe.com/products/flex/tech-specs.html</u>