PROTEXPRESS 1.0

Silver Compatibility Review Steps

NATIONAL® CANCER INSTITUTE

Center for Biomedical Informatics
and Information Technology

March 2, 2009

protExpress Development and Management Teams			
Development	***Quality Assurance***	***Documentation***	***Project and Product Management***
Krishna Kanchinadam[2]	Tom Boal[5]	Carolyn Kelley Klinger[4]	Carl Schaefer[1]
Scott Miller[2]	Nonna Rabinovich[5]	Krishna Kanchinadam[2]	Anand Basu[1]
Steve Matyas[2]		Mahidhar Narra[6]	Xiaopeng Bian[1]
			Bill Mason[2]
			Brent Gendleman[2]
Systems & Application Support			
Bob Wysong[3]	Andrea Johnson[3]	Ralph Rutherford[3]	Sriram Kalyansundaram[3]
Nimish Shah[3]			
1. National Cancer Institute Center for Biomedical Informatics and Information Technology (NCI CBIIT)		2. 5AM Solutions Inc.	3. Terrapin Systems
4. Lockheed Martin	5. NARTech	6. Stelligent	

Contacts and Support	
NCICB Application Support	http://ncicb.nci.nih.gov/NCICB/support Telephone: 301-451-4384 Toll free: 888-478-4423

Contents

Purpose

The purpose of this document is to list the artifacts, and mechanisms for generating such artifacts, required for a Silver Compatibility Review for protExpress v1.0.

List of Required Artifacts

The list of artifacts required for the Silver Compatibility Review is given below. The current set of artifacts is located in SVN at:

https://gforge.nci.nih.gov/svnroot/gpsxar/trunk/docs/silver_compatibility_review/v1.0/artifacts

1. Powerpoint presentation providing an overview of the functionality and basic architecture of protExpress 1.0.
2. UML model – Enterprise Architect model file (.eap).
3. Semantically Annotated XMI file – This is a copy of the most recent XMI file submitted to caDSR for loading.
4. SIW error log – Can be copied from SIW once the annotated XMI file has been loaded successfully into SIW.
5. UML Loader Submission Form – Use the most recently submitted caDSR submission form.
6. Value Domain Report – Refer to the section "**Generating the Value Domain Report**" below.
7. Vocabulary Report
8. Standards Report – The first part of the report should be generated by using the CDE Browser (http://cdebrowser.nci.nih.gov/CDEBrowser/). Open the current version of the protExpress project in the tree browser and choose to Download Data Elements to Excel. Save a copy of the report for editing as described below and save a separate copy for submission (refer to #9 below).

   After downloading the data elements, format the output as below:

   1. Open the file in MS Excel.
   2. Delete all the columns except:
      a. Data Element Long Name
      b. Data Element Version, and,
      c. Data Element Public ID
   3. Order the above three remaining columns in the order specified below:
      a. Data Element Public ID,
      b. Data Element Version,
      c. Data Element Long Name.
   4. Copy the content into the Standards Report Document.

9. CDE Search Results Excel file – This is an unedited copy of the "**Download Data Elements to Excel**" from the CDE browser.
10. Grid Client API Docs – This is the Javadocs for the classes, attributes and associations exposed via the published grid client model. Refer to the section "**Generating Class Documentation (Javadocs)**" below.
11. Grid API Test Client and Logs – This is a set of JUnit classes that attempt to exercise all API methods in the grid client. Refer to the section "**Grid API Test Client and Logs**" below.
12. Use Case Summary Document – This is a MS Word or PDF of the most recent Use Case Summary for protExpress v1.0.
13. Vision Document
14. Technical Guide
15. Pointers to Source Code Control – This is a text document with links (URL's) pointing to the API source code in SVN.

Chapter 1 Reports, Javadocs and API Test Client

Purpose

The purpose of this section is multi-fold, as below:

1. Lists the steps for generating the Value Domain Report
2. Lists the steps for generating the Javadocs for the published model.
3. Describes the API Test Client.

Running Scripts with WScript.exe

WScript.exe is an MS Windows Script Host that enables execution of scripts from Windows. It is usually found in **C:/windows/system32** folder. To run a script, do the following:

1. Open a new command window and navigate to the folder where the script file (to be executed) is located.
2. At the command prompt, execute the following command:

   WScript.exe XYZ

   where, XYZ is the name of the script file to be executed.
3. Example: **WScript.exe protExpress_create_value_domain_report.js**

Generating the Value Domain Report

Steps are:

1. Download the **protexpress_create_value_domain_report.js** file from SVN. The script is located at the url below:

   https://gforge.nci.nih.gov/svnroot/gpsxar/trunk/docs/silver_compatibility_review/v1.0/scripts/protexpress_create_value_domain_report.js
2. Open the script in a text editor, and modify the following variables:
   a. **eaFilePath**: The path to the EA Model file (eg: protExpressGrid.eap)
   b. **csvFilePath**: Name and location of the report file to be created (eg: 06. protexpress-value-domain-report.xls)
3. Execute the script as described in the section "**Running Scripts with WScript.exe**".

Generating Class Documentation (Javadocs)

Steps are:

1. Download the **protexpress_create_pojos_from_model.js** file from SVN. The script is located at the url below:

   https://gforge.nci.nih.gov/svnroot/gpsxar/trunk/docs/silver_compatibility_review/v1.0/scripts/protexpress_create_pojos_from_model.js

2. Open the script in a text editor, and modify the following variables:

   a. **eaFilePath**: The path to the EA Model file (eg: protExpressGrid.eap)

   b. **GENERATED_JAVA_DIR**: Folder location where the java classes will be generated.

3. Execute the script as described in the section "**Running Scripts with WScript.exe**". This will generate the java source files in the folder specified by the variable **GENERATED_JAVA_DIR**.

4. Browse to the folder where the java classes are generated from the previous step (**GENERATED_JAVA_DIR**), open a new command prompt and execute the following command:

   javadoc -d javadocs -public -sourcepath . -subpackages gov

5. If the above command executes successfully, a new subfolder named "**javadocs**" will be created under the folder specified by the variable **GENERATED_JAVA_DIR.**

Grid API Test Client and Logs

This is a test client (ANT script) for the Grid API. It attempts to invoke all methods in the published model and log them in a text file.

The source for the test client is in SVN at:

https://gforge.nci.nih.gov/svnroot/gpsxar/trunk/grid_service_implementation/grid_api_test_client

Download the source code and save to a local folder. This location will be referred to as: **<TEST_CLIENT_SRC_LOCATION>**.

Modify **build.properties** file

Open the build.properties file in a text editor, and specify values for the following properties:

1. server.hostname – Hostname for the application.
2. server.port – Port number on which the application is listening.
3. grid.server.hostname – Hostname where the grid application is deployed.
4. grid.server.port – Port for the grid application.
5. experiment.id – The id of the experiment to be retrieved for the tests.
6. experimentrun.id – The id of the experiment run to be retrieved for the tests.

7. protocol.id – The id of the protocol to be retrieved for the tests.
8. protocolapplication.id – The id of the protocol application to be retrieved for the tests.
9. inputoutputobject.id – The id of the Input/Output to be retrieved for the tests.
10. contactperson.id – The id of the Contact Person to be retrieved for the tests.

Run the Tests and Log the Results

Open a command prompt and execute the ant task **runGridTests**:

ant runGridTests

The tests are executed and the output is logged in appropriate text files in the folder **<TEST_CLIENT_SRC_LOCATION>/out.**