Banner Financial Aid Release Guide

Release 8.8.1
September 2010

SUNGARD HIGHER EDUCATION

Trademark, Publishing Statement and Copyright Notice

Prepared by: SunGard Higher Education

4 Country View Road
Malvern, Pennsylvania 19355
United States of America

Customer Support Center Website

http://connect.sungardhe.com

Documentation Feedback

http://education.sungardhe.com/survey/documentation.html

Distribution Services E-mail Address

distserv@sungardhe.com

Revision History Log

Publication Date	Summary
September 2010	New version that supports Banner Financial Aid 8.8.1 software.

Banner Financial Aid 8.8.1 Release Guide

Contents

Introduction

This guide documents the Banner® Financial Aid 8.8.1 release and describes specific changes to Federal Methodology (FM) Need Analysis processing.

Beginning with the 2011-2012 aid year, SunGard Higher Education will no longer support the INAS integration to Banner. In its place, the US Federal Methodology (FM) and the College Board will provide the Institutional Methodology (IM), integrated with Banner Financial Aid.

These changes are being implemented through either of two Need Analysis options:

- Off-Premise - This Software-as-a-Service (SaaS) component will be managed by SunGard. For this option, there are no specific hardware or software requirements, and you will no longer need to implement patches and upgrades specifically for the FM Need Analysis calculation.
- On-Premise - This option requires local management and is similar to your current implementation. For this option, additional server-class hardware and software is required.

Problem Resolutions

Problem resolutions are usually summarized, by object, at the end of the release guide. However, there are no problem resolutions associated with Release 8.8.1.The `finaid80801resolutions.txt` file that accompanies this release is used only to indicate that there are "No Defects" associated with Release 8.8.1.

Note

Any release hints, suggestions, or corrections to this release guide that arise, subsequent to the posting of this release, will be documented in the *Are there any helpful hints, suggestions and/or corrections to release documentation for Banner Financial Aid release 8.8.1?*, Solution/FAQ # 1-DJZFNP, and made available through the Customer Support Center (https://connect.sungardhe.com/customer_support). It is recommended that you refer to this document periodically under Solutions/FAQs (# 1-DJZFNP) as you use the new release functionality.

Reporting Issues to ActionLine

Should it become necessary to report an issue to the ActionLine for this optional 8.8.1 release of the 2010-2011 FM Need Analysis as a SaaS component, select the following new product name:

- Financial Aid FM Need Analysis

Choose this new name from the Product drop-down list when submitting a Service Request via the Customer Support Center. Please follow this new process for reporting issues to ActionLine. These steps will ensure that your issue is routed and tracked appropriately.

1 Federal Methodology (FM) Need Analysis - Functional

With the Banner Financial Aid 8.6.1 release (January 2010), SunGard® Higher Education provided a pilot program with an alternative for the Federal Methodology (FM) Need Analysis component of INAS for the 2010-2011 aid year. This alternative offers FM Need Analysis calculations in an Off-Premise, software-as-a-service (SaaS) environment, supported by the SunGard Data System Infinity platform.

Note

SunGard Higher Education continues to use College Board INAS for all IM calculations. An institution can also continue using College Board INAS for FM Need Analysis calculations for the 2010-2011 aid year, rather than this new alternative, through options provided on the Global Institution Financial Aid Options Form (ROAINST). ■

This Banner General Availability release document provides an overview of the benefits, architecture, security, prerequisites, processing flow, and object changes of this Off-Premise solution.

For the 2011-2012 aid year, the SunGard Higher Education FM Need Analysis Calculation will be the only FM solution delivered. Schools have been offered the option of electing to supply this solution locally or access the functionality remotely.

Enhancements found in Release 8.8.1 address:

- FM Need Analysis Calculation for 2010-2011 - Enhancements for this release have been made to streamline FM Need Analysis Calculation operations and provide overall processing efficiency.

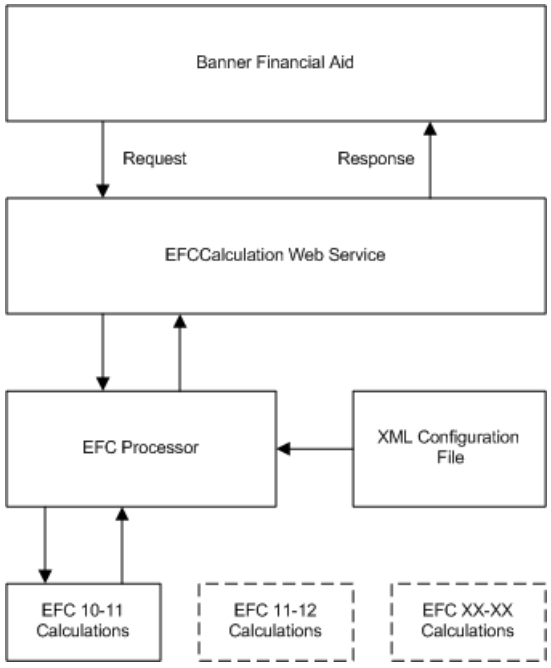
Benefits

This new FM Need Analysis calculation and deployment model provides the following benefits:

- SunGard Higher Education can work directly with the Department of Education to deliver calculation logic to customers more quickly.
- One regulatory posting (in February) can be eliminated annually.
- Updates and defect resolutions can be delivered more quickly.
- Clients can adopt the changes as they occur without having to make upgrades.
- The burden on customer IT staff is reduced because upgrades and defect patches related to FM need analysis are no longer applied on site.

Architecture

The following graphic illustrates the components of the Off-Premise solution:



The components are used as follows:

- The **EFC Calculation Web service** exposes the processing logic used to calculate the EFC.

- The **XML configuration file** identifies the award year to be used for calculating the EFC.
- The **EFC Processor** loads the EFC Calculations component for the award year being calculated.
- The **EFC Calculations component** evaluates eligibility criteria, selects the appropriate calculation formula, and calculates the federal EFC for a given student record. Different EFC Calculations components are used for different award years.

Security

The EFC Calculation Web service is protected via HTTP basic authentication. All information is encrypted via Secure Socket Layer (SSL) between the client and the server.

Only data that is required by the calculation is transmitted through this process. Personally identifiable information (PII) such as names, addresses, and Social Security numbers are not transmitted. The only transmitted data that might be considered PII are the student and parent birth dates, which are required in the calculation.

All data resides on the client database. No information is stored at the Off-Premise site.

Prerequisite

The Off-Premise solution requires outbound network connectivity (TCP 443) from the Banner Job Submission server to the Off-Premise, service fm1.sungardhe.com, via the Internet. The new agent can be configured to use a proxy server. This connectivity allows Banner Financial Aid to communicate to the EFC Calculation Web service.

Setup

Use the following fields on the Options window of the Global Institution Financial Aid Options Form (ROAINST) to set up the options for EFC calculations:

Field	Description
Need Analysis Calculation Option	Code that determines which methodology is used to calculate EFC: *F* Federal Methodology *I* Institutional Methodology *B* Both Federal Methodology and Institutional Methodology (default)
Use Hosted FM Need Analysis Calculation	Check box that indicates whether the Off-Premise solution is used for Federal Methodology calculations: checked — Use Off-Premise solution. This check box can be selected only if **Need Analysis Calculation Option** is set to *F* or *B*. unchecked — Do not use Off-Premise solution (default).

Note

These options will change for the 2011-2012 aid year, as the only FM option will be from SunGard.

Processing flow

The following sections describe the processing flow for EFC calculations. You can calculate EFC online through certain forms or with batch processes, just as you have in the past.

Online processing

You can calculate EFC on the following forms:

RNANA11	Need Analysis
RNAOV11	Applicant Override
RNAPR11	Need Analysis Processing
RNASU11	Supplemental Need Analysis
RNAVR11	Need Analysis Document Verification
ROAIMMP	Applicant Immediate Process

SunGard FM Need Analysis Solution

If your institution uses the new FM Need Analysis solution, the following processing occurs when you calculate EFC on a form.

Note

Even if your institution uses the new FM Need Analysis solution for FM calculations, you can also use the INAS algorithms for IM calculations if the ROAINST option is set appropriately. ■

1. The user calls the need calculation from a Banner Financial Aid form.
2. (Optional) If the **Need Analysis Calculation Option** on ROAINST indicates that your institution uses both IM and FM calculations, the IM calculations are performed as follows:
   - 2.1. The form calls the 2010-2011 Need Analysis Process (RNEIN11).
   - 2.2. RNEIN11 calculates the institutional EFC for the student.
   - 2.3. Banner updates the student records.
3. The FM calculations are performed as follows:
   - 3.1. The form calls the 2010-2011 FM Need Analysis (RNPFM11).
   - 3.2. RNPFM11 calls the EFC Calculation Web service.
   - 3.3. The Web service calculates the federal EFC for the student.
   - 3.4. The Web service returns the results to Banner.
   - 3.5. Banner updates the student records.

Batch processing

The following batch processes calculate EFC:

- 2010-2011 Need Analysis Process (RNEIN11) uses INAS algorithms for IM calculations.
- 2010-2011 FM Need Analysis (RNPFM11) uses the new FM Need Analysis solution for FM calculations.

Processing flow depends on whether your institution uses FM, IM, or both FM and IM. This is determined by the **Need Analysis Calculation Option** on ROAINST.

SunGard FM Need Analysis Solution

If your institution uses the new FM Need Analysis solution, the following processing occurs when batch EFC calculations are performed.

Note

Even if your institution uses the new FM Need Analysis solution for FM calculations, you can also use the INAS algorithms for IM calculations if the ROAINST option is set appropriately. ■

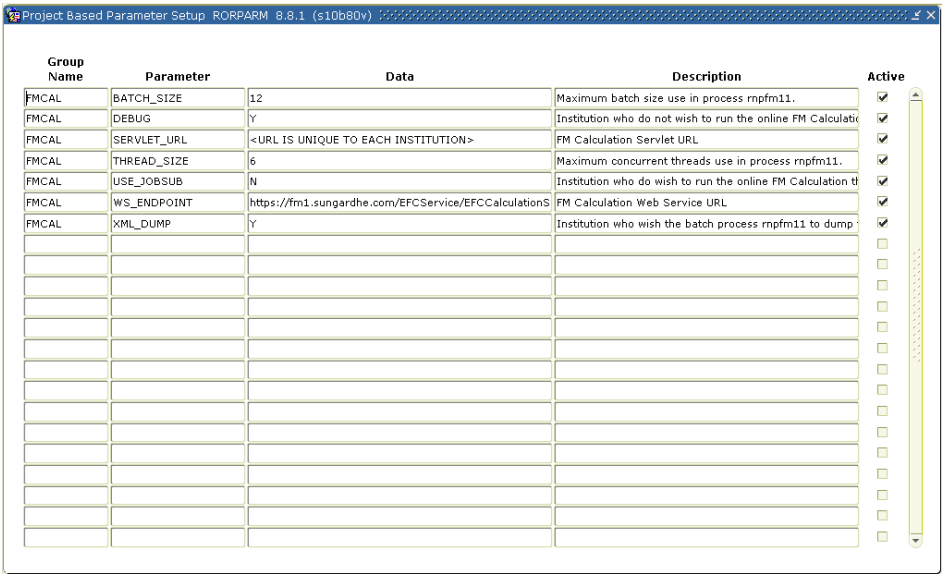
1. The user executes the 2010-2011 FM Need Analysis (RNPFM11).
2. RNPFM11 calls the EFC Calculation Web service.
3. The Web service calculates the federal EFC for the students.
4. The Web service returns the results to Banner.
5. Banner updates the student records.
6. (Optional) If the **Need Analysis Calculation Option** on ROAINST indicates that your institution uses both IM and FM calculations, the IM calculations are performed as follows:
   - 6.1. RNPFM11 automatically calls the 2010-2011 Need Analysis Process (RNEIN11).
   - 6.2. RNEIN11 calculates the institutional EFC for the same population of students.
   - 6.3. Banner updates the student records.

New Form

The following form has been added for this release.

Project Based Parameter Setup Form (RORPARM)

This new form has been included to allow you to configure institution specific data to perform Federal Methodology Need Analysis Calculations.



The RORPARM form was designed to accept entry of group-based parameter data.

Field	Description
Group Name	Group name.
Parameter	The group parameter.
Data	The data associated with the group parameter.
Description	Description of the parameter.
Active	Indicates the parameter is active.

Parameter Configuration

BATCH_SIZE

The BATCH_SIZE parameter, working in conjunction with THREAD_SIZE parameter, are used to optimize performance of the FM Calculation batch process, running from Process Submission Controls (GJAPCTL) form. It is recommended that these initial settings not be modified unless instructed to do so by SunGard Higher Education's Financial Aid ActionLine.

DEBUG

When running the FM Calculation process online, any errors that occur will be displayed as a pop-up window on the form. By default, these error messages will be shown in full text and unedited. This parameter can be set to 'N' to minimize the number of error messages.

SERVLET_URL

By default, the online FM Calculation process is configured to run using a servlet running on the Oracle Application Server (OAS). This servlet must be configured and installed at each institution. Once installed, the servlet's URL must be entered and saved (in place of the default string, "<*URL IS UNIQUE TO EACH INSTITUTION*>"). The value entered should use the following format, "http://<hosting_computer>:<port_number>/rnpfmxx/FmCalxx". A valid URL could be:

```
http://mal0500374.corp.sct.com:8888/rnpfmxx/FmCalxx
```

Note

To perform correctly, the value of the SERVLET_URL parameter must be provided by your institution for all aspects of the Federal Methodology Need Analysis Calculation. ■

Tip

The FmCalxx portion of the above URL is case sensitive. ■

THREAD_SIZE

Refer to parameter BATCH_SIZE for details.

USE_JOBSUB

By default, the online FM Calculation is configured to run using a servlet. As an option, your institution can run the FM Calculation via the Banner job submission server. To run the online FM Calculation through Banner job submission, set this parameter to 'Y'.

WS_ENDPOINT

By default, the FM Calculation Web Service is supplied by Sungard Higher Education. For institutions that have opted to provide the Web Service locally (On-Premise), the institution URL to the On-Premise Web Service must be updated here.

XML_DUMP

The information transmitted between the FM Calculation Client and the Web Service is passed using XML. By default, the XML information is not persistent. Data only exists within the context of the data transmitted between the Client and Web Service and cannot be viewed. If your institution wishes to see the actual data that was transmitted, update this parameter to 'Y'. For an FM Calculation that was processed through job submission, the XML dump can be found in the log file on the job submission server. For online processing that uses the servlet, the XML dump will be available in the log file located on the OAS.

Warning

There is a significant system performance trade-off in setting the XML_DUMP parameter to 'Y' when running through job submission. SunGard Higher Education highly recommends using this option only for a very small population. Once processing is complete (on a job-by-job basis), the parameter should be returned to the default value of 'N'.

Changed Forms

The following form changes support this enhancement.

Financial Aid PL/SQL Library (ROQRPLS)

Code used to make calls to the Java servlet have been centralized in this Banner forms library. The finaidutils.jar, is imported to make the HTTP Post request to the FM Calculation servlet.

Note

For institutions who do not have OAS with OC4J, version 10.1.3 (or higher), a work around will allow FM Calculator operations to be run from Job Submission using rnpfm11.jar (instead of the servlet). This work around code will execute when the data for parameter USE_JOBSUB = Y in RORPARM.

Tip

For additional information on the finaidutils.jar object, refer to *Appendix A - Additional Technical Details, Form Related Objects*.

Common Modifications

The following forms have all been changed to use the modified Financial Aid PL/SQL Library (ROQRPLS) form to make an HTTP Post to the `RNPFMXX.war` Web Archive.

- Need Analysis (RNANA11)
- Applicant Override (RNAOV11)
- Need Analysis Processing (RNAPR11)
- Supplemental Need Analysis (RNASU11)
- Need Analysis Document Verification (RNAVR11)
- Applicant Immediate Process (ROAIMMP)

New Process

2010-2011 FM Need Analysis (RNPFM11)

When this new process is submitted, the previously established web service makes the call to the EFC service and sends the batch through for calculation. The FM Need Analysis calculation is performed off-premise and the results are returned via the established web service. Banner updates to the student records, accordingly.

Note

No Personally Identifiable Information is included in Off-Premise transactions.

The following parameters have been included with the new RNPFM11 process:

Parm	Definitions	Validation/LOV	Description
01	Aid Year Code Required Single Length = *4* Type = *Character*	`ROBINST_EQUAL_AIDY_ACTIVE` Default = 1011	The Aid Year Code for the 2010-2011 aid year must be entered.
02	Process Indicator Required Single Length = *1* Type = *Character*		Indicates which students are to be processed. B (Batch-All Students for Aid Yr), R (Recalc-In waiting status)

Parm	Definitions	Validation/LOV	Description
03	Applicant ID Optional Single Length = *9* Type = *Character*		Must be left blank when running in batch.
04	Application Code Optional Single Length = *30* Type = *Character*		General area for which the selection ID has been defined.
05	Selection ID Optional Single Length = *30* Type = *Character*		Code that identifies the sub-population to work with.
06	Creator ID Optional Single Length = *30* Type = *Character*		ID of the person creating the sub-population rules.
07	User ID Optional Single Length = *30* Type = *Character*		The ID of the person using the sub-population rule.
08	Report Options Required Single Length = *1* Type = *Character*	Default = X	Report print options; (N) Sort by Name, (I) Sort by ID, [X] No Report.

An associated warning message will be located at the top of the RNPFM11 `output.log` file. This message may be repeated multiple times during the run.

WARNING: Exception extracting jars into temporary directory: java.io.IOException: Permission denied : switching to alternate class loading mechanism.

Note

The warning message may be ignored. ■

2 Miscellaneous Enhancements - Functional

This section describes an updated header object (sis.h) that has been changed to meet Internationalization standards.

Internationalization

sis.h (FINAID C Header File)

This header object was updated to meet Internationalization standards.

3 Federal Methodology (FM) Need Analysis - Technical

This chapter describes the technical changes that support the FM Need Analysis enhancement.

New Tables

Project Specific Parameter Configuration Table (RORPARM)

This new table has been added to allow you to define your specific configuration. This new table stores technical parameters for the new FM Need Analysis process.

Note

If your institution uses an On-Premise FM Need Analysis Calculation Web Service, you must replace the delivered URL with your local address.

Column	Null?	Type	Description
RORPARM_GROUP_NAME	No	VARCHAR(30)	The overall group name which owns the set of parameters (the group name for the FM Need Analysis calculation process is "FMCAL".).
RORPARM_PARAMETER	No	VARCHAR(30)	The name of the parameter.
RORPARM_ACTIVE_IND	No	VARCHAR(1)	The parameter is active or not.
RORPARM_DATA	No	VARCHAR(120)	The data associated with the parameter.
RORPARM_DESCRIPTION	Yes	VARCHAR(120)	The description of the parameter.
RORPARM_USER_ID	Yes	VARCHAR(30)	The user ID of the person who inserted or last updated this record.
RORPARM_ACTIVITY_DATE	Yes	DATE	The date that information in this record was entered or last updated.
RORPARM_DATA_ORIGIN	Yes	VARCHAR(30)	Source system that created or updated the data.

Temporary Table to Hold Unique Token to Validate Access to FM CAL Servlet (RPRTKEN)

This new table has been added to store a unique token to validate and secure access to the FM Calculation Java servlet. This additional security is required because the Java servlet is stored and operates from the Oracle Application Server (OAS). The OAS for many clients also acts as the server for their Banner Self-Service product. Without this additional security validation, anyone can see and make an HTTP request of the Java servlet.

This unique token will be used at each request to stop unauthorized execution of the FM Calculation's Java servlet. A token is created via Banner forms and stored in this new table. When a Banner form makes an HTTP request, the same unique token is passed as one of the expected parameters. When the Java servlet receives the request, that request's token will be checked before moving forward with the calculation. If the token is missing, the request will be rejected.

Between calculations, the unique token will be deleted by the Java servlet and the calling forms. The Java servlet is designed to delete the token as soon as possible, after validation. The calling forms also delete the token, if by chance an error should occur and the request to the Java servlet was not successful.

Column	Null?	Type	Description
RPRTKEN_TOKEN	No	VARCHAR(30)	The unique token that is used to validate a user of the database.
RPRTKEN_USER_ID	Yes	VARCHAR(30)	The user ID of the person who inserted or last updated this record.
RPRTKEN_ACTIVITY_DATE	Yes	DATE	The date that information in this record was entered or last updated.
RPRTKEN_DATA_ORIGIN	Yes	VARCHAR(30)	Source system that created or updated the data.

New Database Triggers

RT_RORPARM_LOG_INSUPDDTL (rotparm9.sql)

This new trigger has been included to perform logging for the RORPARM table.

Note

This new trigger is owned by FAISMGR.

RT_RORPARM_USERACTIVIT_INSUPD (rotparm0.sql)

This new trigger has been included to maintain the User ID and activity date for the RORPARM table.

Note

This new trigger is owned by FAISMGR.

Required Data

RPRPARMI_071801.SQL

The `rprparmi_071801.sql` script has been included to make entries to the Project Specific Parameter Configuration Table (RORPARM) table.

This script will be executed as part of the install process to ensure correct table values exist prior to user processing.

The following data has been added to the RORPARM table for FMCAL Group:

RORPARM_GROUP_NAME	RORPARM_PARAMETER	RORPARM_ACTIVE_IND	RORPARM_DATA	RORPARM_DESCRIPTION
FMCAL	WS_ENDPOINT	Y	https://fm1.sungardhe.com/EFCService/EFCCalculationService.asmx	FM Calculation Web Service URL
FMCAL	SERVLET_URL	Y	<URL IS UNIQUE TO EACH INSTITUTION>	FM Calculation Servlet URL
FMCAL	BATCH_SIZE	Y	12	Maximum batch size used in process RNPFM11.
FMCAL	THREAD_SIZE	Y	3	Maximum concurrent threads used in process RNPFM11.
FMCAL	USE_JOBSUB	Y	N	Institutions who wish to run the online FM Calculation through the Job Submission server should set this value to: *Y*.

RORPARM_GROUP_NAME	RORPARM_PARAMETER	RORPARM_ACTIVE_IND	RORPARM_DATA	RORPARM_DESCRIPTION
FMCAL	DEBUG	Y	Y	Institutions who do not wish to run the online FM Calculation with verbose debug messages.
FMCAL	XML_DUMP	Y	N	Institutions who wish the batch process RNPFM11 to dump the XML data to the log files.

Logging Control Base Table (ROBLOGC)

ROBLOGCI_080801.sql

The roblogci_080801.sql script has been included to make entries to the Logging Control Base Table (ROBLOGC) for the RORPARM table.

This script will be executed as part of the install process to ensure correct table values exist prior to user processing.

Logging Control Repeating Table (RORLOGC)

RORLOGCI_080801.SQL Script

The rorlogci_080801.sql script has been included to make entries to the Logging Control Repeating Table (ROBLOGC) for the following columns:

RORLOGC_TABLE_NAME	RORLOGC_COLUMN_NAME	RORLOGC_COLUMN_DESC	RORLOGC_LOG_IND
RORPRAM	RORPARM_GROUP_NAME	GROUP NAME	N
RORPRAM	RORPARM_PARAMETER	PARAMETER	N
RORPRAM	RORPARM_ACTIVE_IND	ACTIVE INDICATOR	N
RORPRAM	RORPARM_DATA	DATA	N
RORPRAM	RORPARM_DESCRIPTION	DESCRIPTION	N
RORPRAM	RORPARM_USER_ID	USER ID	N
RORPRAM	RORPARM_ACTIVITY_DATE	ACTIVITY DATE	N
RORPRAM	RORPARM_DATA_ORIGIN	DATA ORIGIN	N

The `rorlogci_080801.sql` script will be executed as part of the install process to ensure correct table values exist prior to user processing.

A Additional Technical Details

This release, along with its description of Banner Financial Aid Need Analysis and the Off-Premise (SaaS) FM Calculation, includes behind the scenes technical enhancements. This appendix describes additional technical information for the following topics:

- Form Related Object
- Process Related Object
- Software Installation and Configuration
- FM Need Analysis Schema

Form Related Object

finaidutils.jar

Because Oracle forms do not directly support the HTTP Post command, this new Java object has been provided to implement the HTTP Post command.

- This Java language based support library will be used to extend the basic functionality of Oracle forms. This library allows the Oracle form the ability to make an HTTP post to a URL.

Note

This object is located on the Oracle Application Server (OAS).

Process Related Object

RNPFMXX.war

This new web archive is associated with the RNPFM11 process. This new functionality allows selected Banner Financial Aid processing to behave as if it were a Java servlet.

- A JAVA language Web Archive, this servlet is wrapped around the FM Need Analysis process (rnpfm11.jar). This allows the FM Need Analysis process to run on the OAS instead of the Banner job submission server.
- Implementation of this web archive greatly enhances performance for the online FM Need calculation.

Software Installation and Configuration

Installation Options

There are several options available when installing the FM Need Analysis Calculation. This section acts as a guide to determine which steps are required for your system.

Servlet vs. Process

The online FM Need Analysis Calculation can be configured to execute as a:

- Servlet (`rnpfmxx.war`) running on Oracle Application Server. For performance enhancement, this is the default and preferred installation. All steps of the *Online FM Calculations using servlet* must be fully implemented.

  OR

- Process (`rnpfm11.jar`) running on the job submission server. All steps of *Online FM Calculations using job submission* must be fully implemented.

Off-Premise vs. On-Premise

Institutions may choose to host their own local version of the FM Need Analysis Calculation Web Service:

- Off-Premise is the default configuration. This will use the software-as-a-service (SaaS) supported by SunGard Data System Infinity platform. No additional installation is required.

  OR

- On-Premise installations are provided for those institutions that wish to maintain their own Web Services. A guide, *On-Premise (Local) FM Calculations*, is included to assist in configuring your system. Two separates installation procedures are provided, one each for Windows 2003 and Windows 2008. Only one of the two Web Service options needs to be implemented.

Installation Requirement (all installations)

Updating security nas.properties file of rnpfm11.jar

The nas.properties file is embedded with the process `rnpfm11.jar`.

Updating rnpfm11.jar

1. The batch process requires a user name and password to connect to the FM Calculation Web Service, EFCService. The following steps outline the process of updating `rnpfm11.jar` with the institution unique authentication.

2. To extract the `nas.properties` file from `rnpfm11.jar`, use the following command:

   ```
   jar xf rnpfm11.jar nas.properties
   ```

3. Open the `nas.properties` file in a text editor.

4. Change the user name and password values, the text to the right of the equal sign, for these properties

   - `httpusername=`
   - `httppassword=`

Note

For Off-Premise applications, use the authentication code provided when the institution applied for the Off-Premise status. For On-Premise applications, use the authentication setup during the install of the Windows IIS, if basic authentication was configured (otherwise, make no changes).

5. Update the `rnpfm11.jar` with the edited `nas.properties` file with the following command:

   ```
   jar uf rnpfm11.jar nas.properties
   ```

Online FM Calculations using servlet

Overview

These instructions are used to configure your system to allow online (Off-Premise) FM Calculations in a software-as-a-service (SaaS) environment, supported by the SunGard Data System Infinity platform.

Step 1 Minimum Requirements

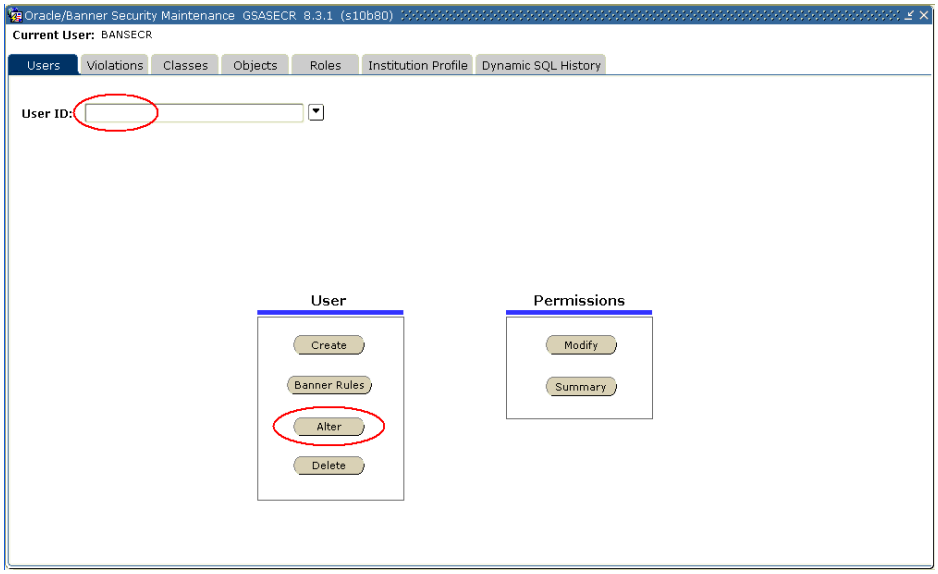
The servlet `rnpfmxx.war` can only be installed on an Oracle Containers for J2EE (OC4J) of version 10.1.3.

Step 2 Configure banproxy

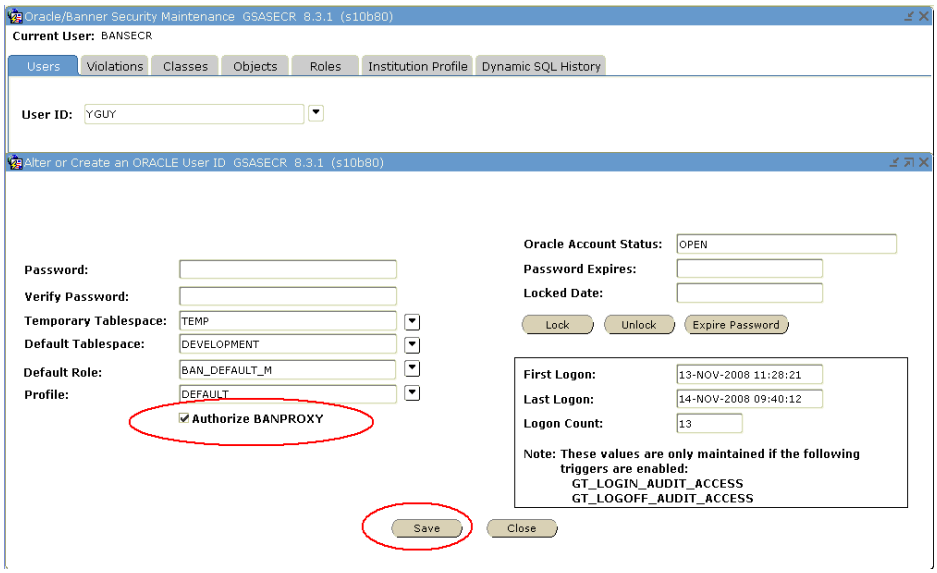
1. A proxy connection must be configured for every FINAID user who will be running Off-Premise FM Calculations. The INB banproxy user will be configured to act as the proxy user. Appropriate permission is required for each user to allow banproxy to connect to Oracle on that login user's behalf.

2. Open the Oracle/Banner Security Maintenance (GSASECR) form.

3. Select the User ID that will use the proxy connection and click the **Alter** button.



4. The User ID's credentials are displayed on the Authorize or Create an ORACLE ID window. Check the **Authorize BANPROXY** checkbox and click the **Save** button.



Step 3 Deploying finaidutils.jar

1. The `finaidutils.jar` file (…/`finaid/java/finaidutils.jar`) must be copied to the Oracle Application Server (OAS).

2. The file name `finaidutils.jar` must be appended to the end of the CLASSPATH in the file `OAS_HOME/forms/server/default.env`.

3. Once the above changes are made, INB must be restarted.

Step 4 Updating rnpfmxx.war

1. For the servlet, both the user name/password and the security seed numbers must be updated.

2. The war archive file `rnpfmxx.war` default seed numbers must be updated with the institution seed numbers.

Note

Two security seed numbers are required for Oracle forms, COBOL and Java processes to be allowed to connect to the Oracle database. ■

3. Login into the environment where Banner Financial Aid is installed. Navigate to `<BANNER_HOME>/finaid/java/`.

4. To extract the `nas.properties` file from `rnpfmxx.war` use the following command:

```
jar xf rnpfmxx.war WEB-INF/nas.properties
```

5. Open the `WEB-INF/nas.properties` file in a text editor.

6. Change the user name and password values, the text to the right of the equal sign, for these properties

   - `httpusername=`
   - `httppassword=`

Note

For Off-Premise applications, use the authentication code provided when the institution applied for the Off-Premise status. For On-Premise applications, use the authentication setup during the install of the Windows IIS, if basic authentication was configured (otherwise, make no changes).

7. Using the institution security seed number, change the values, the text to the right of the equal sign, for these properties

   - `seed1=`
   - `seed3=`

   7.1. Update the rnpfmxx.war with the edited nas.properties file with the following command:

```
jar uf rnpfmxx.war WEB-INF/nas.properties
```

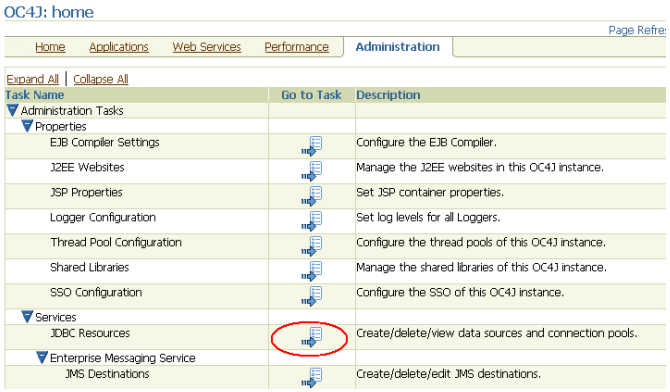
8. Cleanup by deleting the following temporary file and directory:

   - `nas.properties`
   - WEB-INF directory and its content.

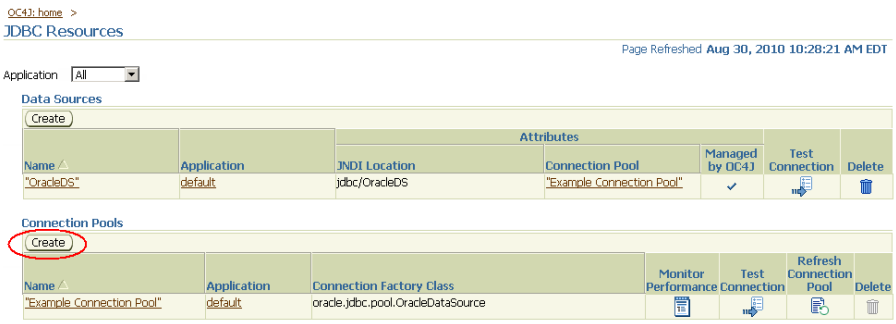
Step 5 Create Connection Pool and Data Source

1. Within the Oracle Enterprise Manager, navigate to the **Administration** tab.

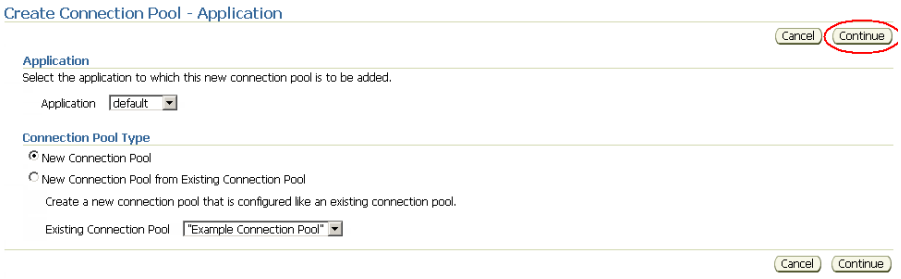
2. Click the **Services>JDBC Resources** Go to Task icon to display associated JDBC Resources.



3. Under the **Connection Pools** heading, click the **Create** button to create a connection pool.



4. From the **Create Connection Pool - Application** screen, click the **Continue** button to define the pool.



5. Define the following Connection Pool information:

   5.1. **Name**: `banner/finaid`

5.2. **JDBC URL**: use `jdbc:oracle:thin:@<HOST>:<PORT>:<SERVICE_NAME>`

This string will be used to connect to the Oracle database. For example, `jdbc:oracle:thin:@myserver:1521:mydatabase` where:

`<HOST>` is `myserver`

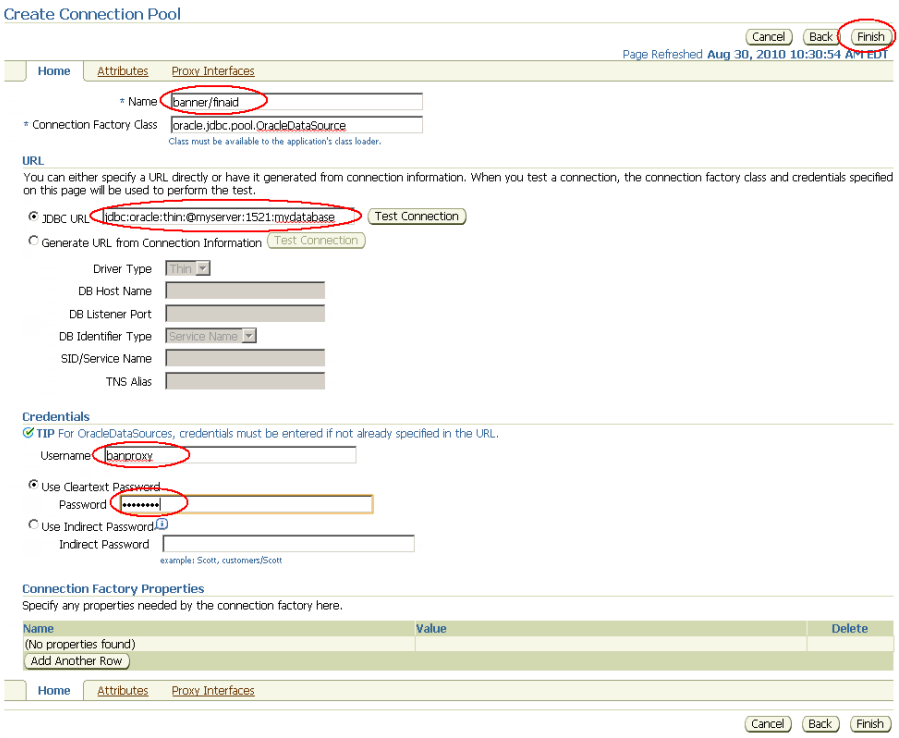
`<PORT>` is `1521`

`<SERVICE_NAME>` is `mydatabase`

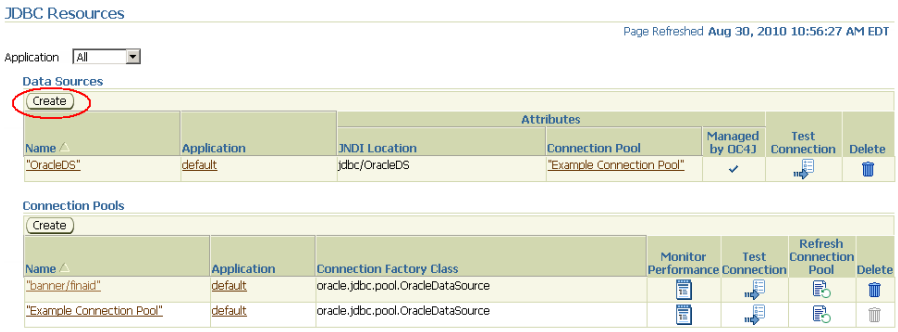
5.3. **Username**: `banproxy`

5.4. **Use Cleartext Password**: Use the password defined by the institution for user, `banproxy`.

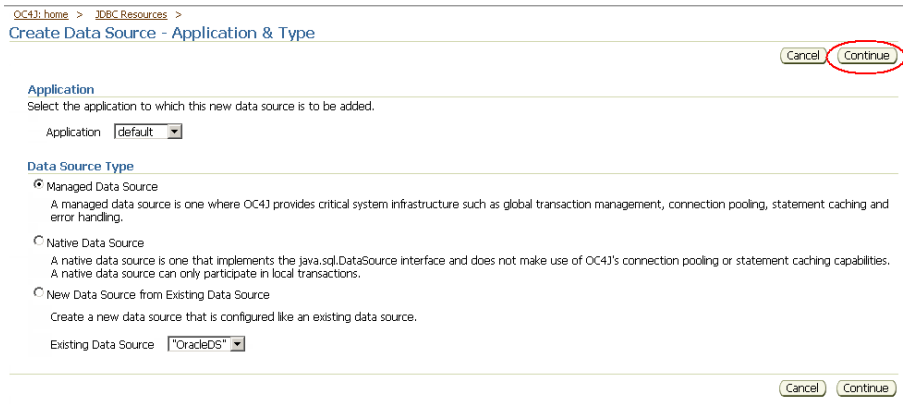
5.5. Click the **Finish** button to establish your Connection Pool configuration.



6. Under the **Data Sources** heading, click the **Create** button to create a Data Source.



7. From the **Create Data Source - Application & Type** screen, click the **Continue** button to define the data source.

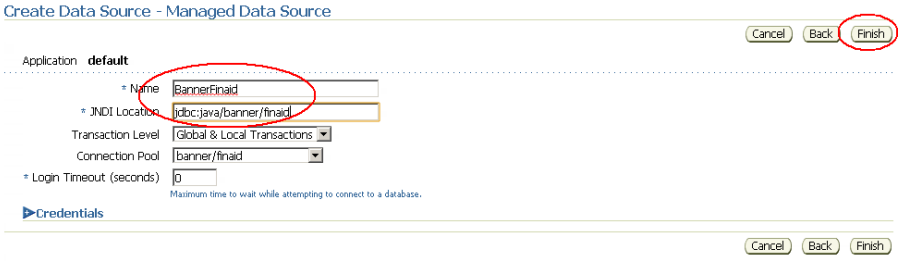


8. From the **Create Data Source - Managed Data Source** screen, define the following Data Source information:

   8.1. **Name**: `BannerFinaid`

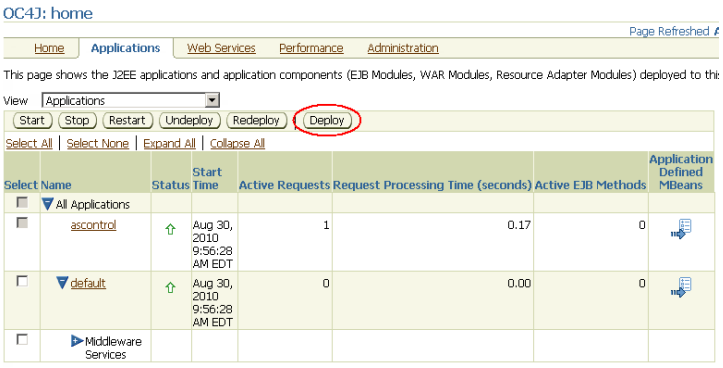
   8.2. **JNDI Location**: `jdbc:java/banner/finaid`

   8.3. Click the **Finish** button to establish your Data Source configuration.

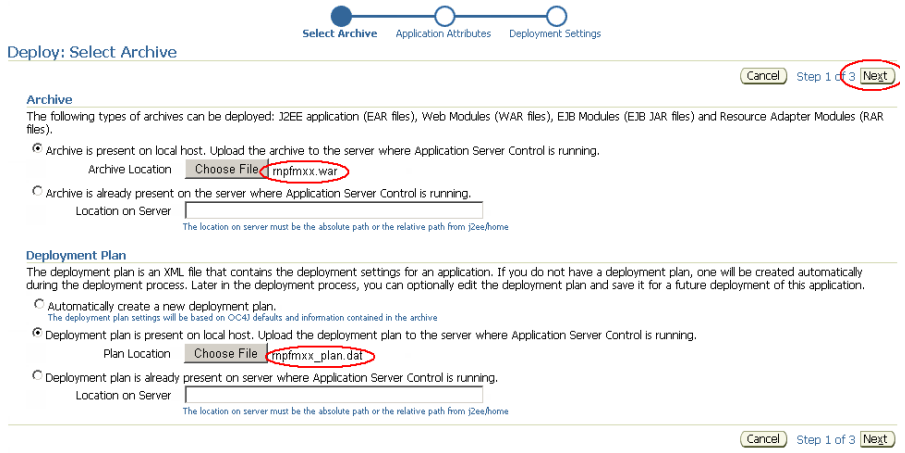


Step 6 Deploying rnpfmxx.war

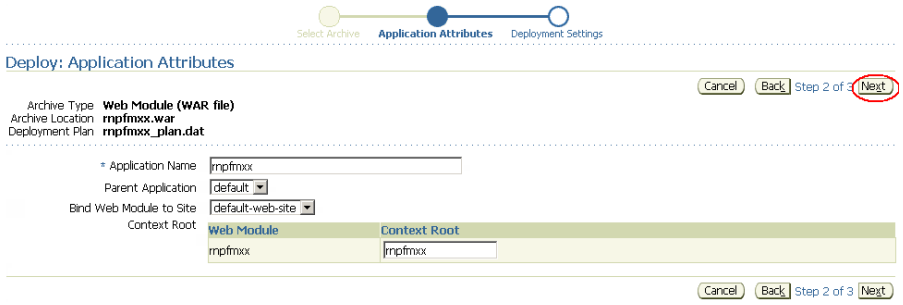
1. The following two files must be copied to a location where web browser access is possible:
   - …/finaid/java/rnpfmxx.war (archive)
   - rnpfmxx_plan.dat (deployment plan)
2. From a Web Browser, login into the Oracle Enterprise Manager.
3. Navigate to the **Applications** tab and click the **Deploy** button.



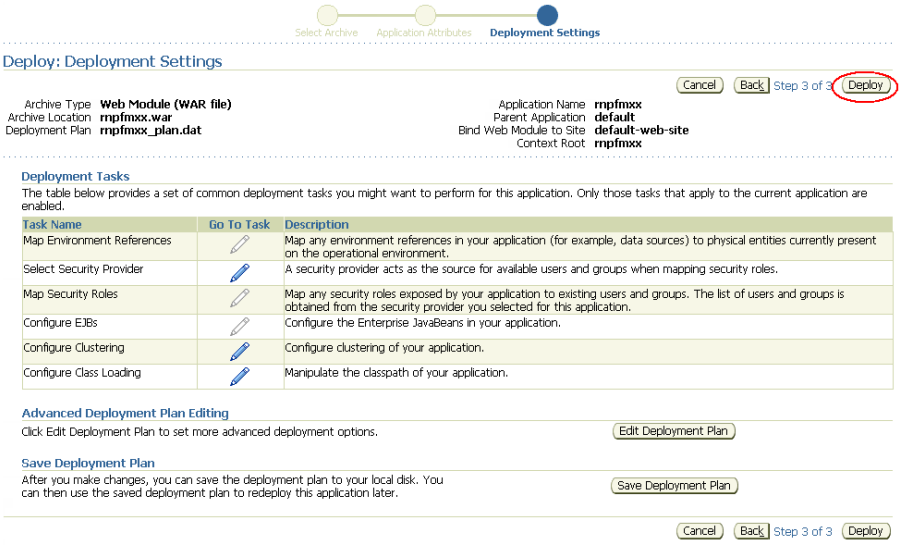
4. Click the **Choose File** button and navigate to the rnpfmxx.war archive file and click **Open**.
5. Click the **Choose File** button and navigate to the rnpfmxx_plan.dat deployment plan file and click **Open**.
6. From the **Deploy: Select Archive** screen (1 of 3), click the **Next** button to continue to the **Deploy: Application Attributes** screen (2 of 3).



7. From the **Deploy: Application Attributes** screen (2 of 3), wait for the selected files to be uploaded and click the **Next** button to continue to the **Deploy: Deployment Settings** screen (3 of 3).



8. From the **Deploy: Deployment Settings** screen (3 of 3), click the **Deploy** button to acknowledge your recently configured settings.



9. A confirmation page will be displayed to indicate successful deployment. From the Confirmation page, click the **Return** button to accept the configuration.

Step 7 Update INB to recognize the rnpfmxx.war Servlet

1. Login into INB and navigate to the Project Based Parameter Setup Form (RORPARM).

2. For parameter `SERVLET_URL`, replace the default value *<URL IS UNIQUE TO EACH INSTITUTION>* with the URL of the servlet.

   The servlet URL uses the following format:

   `http://<SERVER>:<PORT>/rnpfmxx/FmCalxx`

   `<SERVER>` the name of the Server where `rnpfmxx.war` is deployed

   `<PORT>` the port number

Online FM Calculations using job submission

Update INB to use job submission to process online (Off-Premise) FM Calculations.

1. Login into INB and navigate to the Project Based Parameter Setup Form (RORPARM).

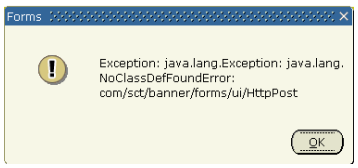
2. For parameter `USE_JOBSUB`, change the default value from “Y” to “N”.

Troubleshooting the FM Calculation

Missing finaidutils.jar.

Related Error Message

com/sct/banner/forms/ui/HttpPost



Description

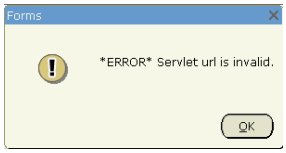
Symptom: INB cannot find finaidutils.jar.

Solution: Review: *Software Installation and Configuration,Off-Premise (SaaS) FM Calculations, Step 3 Deploying finaidutils.jar.*

Unable to connect to servlet rnpfmxx.war.

Related Error Message

ERROR Servlet url is invalid.



Description

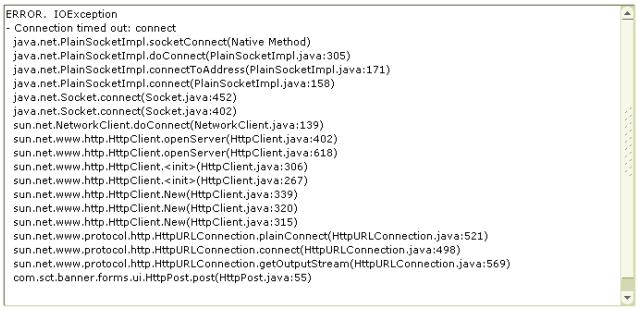
Symptom: The parameter `SERVLET_URL` defined on the Project Based Parameter Setup Form (RORPARM) has not been updated with the correct URL.

Solution: Review: *Software Installation and Configuration, Off-Premise (SaaS) FM Calculations, Step 4 Deploying rnpfmxx.war.*

Connection timed out

Related Error Message

Connection timed out



Description

Symptom: A parameter for `SERVLET_URL` was provided on the Project Based Parameter Setup Form (RORPARM). The URL provided is valid, but the service related to the URL is not responding.

Diagnostic: Double check the URL and make sure the value specifies the `rnpfmxx.war` servlet.

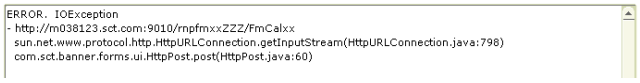
Note: The `www.sungardhe.com` web address is a valid URL. However, its value is not the correct URL for the servlet.

Solution: Review: *Software Installation and Configuration, Off-Premise (SaaS) FM Calculations, Step 6 Update INB to recognize the rnpfmxx.war Servlet.*

IOException

Related Error Message

IOException



Description

Symptom: A parameter for SERVLET_URL was provided on the Project Based Parameter Setup Form (RORPARM). The URL provided is valid, but the service related to the URL was not found.

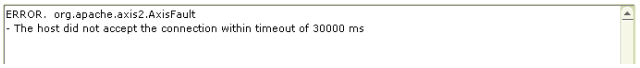
Solution: Review: *Software Installation and Configuration, Off-Premise (SaaS) FM Calculations, Step 6 Update INB to recognize the rnpfmxx.war Servlet.*

Invalid URL for FM Calculation Web Service

Related Error Message

org.apache.axis2.AxisFault

OR



OR

Description

Symptom: The parameter for WS_ENDPOINT in the Project Based Parameter Setup Form (RORPARM) should be:
https://fm1.sungardhe.com/EFCService/EFCCalculationService.asmx

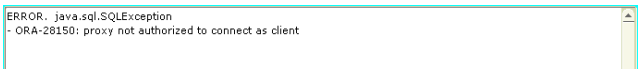
Diagnostic: For institutions implementing the on-premise FM Calculation Web Service, double check that the URL is correct.

Solution: Review: *Software Installation and Configuration, On-Premise (Local) FM Calculations.*

Unauthorized proxy login for Oracle

Related Error Message

proxy not authorized to connect as client



Description

Symptom: The user login into INB is not setup to allow proxy login.

Solution: Review: *Software Installation and Configuration, Off-Premise (SaaS) FM Calculations, Step 2 Configure banproxy.*

On-premise (Local) FM Calculations

Installing IIS 6.0 on Windows Server 2003

Overview

These instructions are used to configure your system, using IIS 6.0 on Windows 2003 to allow On-Premise (Local) FM Calculations.

Step 1 Minimum Requirements

1. The On-Premise install of the EFCService Web Service can only be done on a Windows Server:
   - 1.1. Windows Server 2003 running IIS 6.
   - 1.2. Microsoft.NET Framework 3.5.
   - 1.3. IIS configured to run ASP.net 2.0.

Step 2 Installing IIS 6.0 Windows Server 2003

1. Detailed instructions are available from Microsoft at the following URL:

   ```
   http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/
   Library/IIS/750d3137-462c-491d-b6c7-5f370d7f26cd.mspx?mfr=true
   ```

   1.1. Click **Start** > **Programs** > **Administrative Tools** > **Manage Your Server**.

   1.2. Click **Add or remove a role**.

   1.3. Use the Wizard to add the Application Server (IIS, ASP.NET).

Step 3 Install EFCService Web Service

1. Unzip the contents of the `SGHE_EFCService_Install_10292010.zip` file into the following folder:

   ```
   C:\inetpub\wwwroot\EFCService
   ```

The structure includes:

```
..\EFCService\App_Data\Config.xml
              \bin\EFC.dll
                  \EFCSoapExtension.dll
                  \ExceptionHandling.dll
                  \FederalEFC1011.dll
                  \FederalEFC.dll
                  \FinancialAidService.dll
              \efc_calculation_messages.xsd
              \EFCCalculationService.asmx
              \Web.config
```

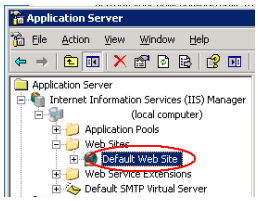
2. If the installed folder is the suggested folder `C:\inetpub\wwwroot\EFCService`, skip directly to item 4., below.

3. Open the files `Config.xml` and `Web.config` in a text editor and replace every instance of `C:\inetpub\wwwroot\EFCService` with the path of the actual installed folder. For example, if the installed folder is `C:\EFCService`, do a search and replace of `C:\inetpub\wwwroot\EFCService` with `C:\EFCService`.

4. Start the **Manage Your Server**.

5. Click **Manage this application server**.

6. From the Application Server, navigate to the **Default Web Site**.



7. Set the ASP.NET version.

   7.1. Right click on the Default Web Site and select **Properties**.

   7.2. On the ASP.NET tab, set the ASP.NET version to 2.0.x.

   7.3. Click OK.

8. Create a Virtual Directory.

   8.1. Right click on the Default Web Site and select New > **Virtual Directory**.

   8.2. Use the Virtual Directory Creation Wizard.

      8.2.1. Alias: **EFCService**.

      8.2.2. Path: Use the path specified in Step 3, item 3, above.

      8.2.3. On the Virtual Access Permissions, check both:

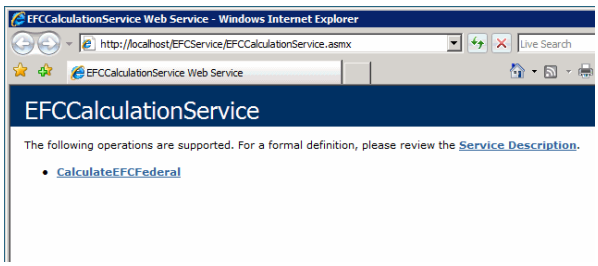
      - Read
      - Run Scripts (such as ASP)

Step 4 Web Service Test

1. From a Web Browser, go to the following URL:

   `http://localhost/EFCService/EFCCalculationService.asmx`

2. The EFCCalculationService page is displayed.

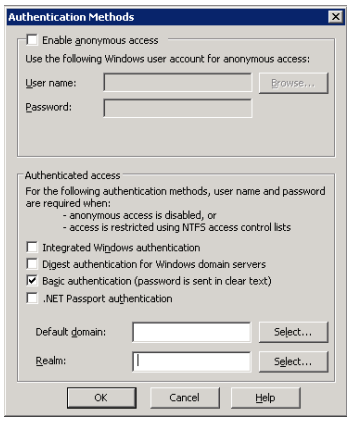


Step 5 Configuring the EFCService to use Basic Authentication (optional)

Note

This step is optional. If Step 5 is configured, Step 8 Updating the user name and password to connect to the EFCService (optional), below, must also be configured.

1. Navigate back to the Default Web Site on the Application Server (see Step 3 Install EFCService Web Service, items 4 and 5, above).
2. Right click on the Default Web Site. Select Properties.
3. On the **Directory Security** tab, edit the **Authentication and access control**.
4. Uncheck **Enable anonymous access**.
5. Check **Basic authentication** and fill in the **Default domain** and **Realm**.



Step 6 Setup EFCService for SSL (optional)

Note

This step is optional. If Step 6 is configured, Step 9 Updating the infinity.keystore (optional), below, must also be configured.

1. SSL requires obtaining and registering a certificate with the Web Server.
2. The bit length must be 1024.
3. Detailed instructions are available from Microsoft at the following URL:

```
http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/
Library/IIS/89c7ef2f-f7d6-483c-8b08-ae0c6584dd4d.mspx?mfr=true
```

Step 7 Updating INB to see the On-Premise FM Calculation Web Service

1. Login into INB and navigate to the Project Based Parameter Setup Form (RORPARM).

2. For parameter **WS_ENDPOINT**, replace the default value with the URL from Step 4 Web Service Test, above:

   ```
   http://localhost/EFCService/EFCCalculationService.asmx
   ```

3. Make sure to replace **localhost** with the correct address of the IIS.

Step 8 Updating the user name and password to connect to the EFCService (optional)

Note

If **Basic authentication** is configured in Step 5 Configuring the EFCService to use Basic Authentication (optional), above, this step must be completed.

1. Login into the environment where Banner Financial Aid is installed. Navigate to:

   ```
   <BANNER_HOME>/finaid/java/
   ```

2. Updating `rnpfm11.jar`

3. To extract the `nas.properties` file from `rnpfm11.jar`, use the following command:

   ```
   jar xf rnpfm11.jar nas.properties
   ```

4. Open the `nas.properties` file in a text editor.

5. Using a valid user name and password from the domain (configured in the Basic Authentication of Step 5 Configuring the EFCService to use Basic Authentication, item 5., above). Change the values, the text to the right of the equal sign, with the following properties:

   - `httpusername=`
   - `httppassword=`

   5.1. Update the `rnpfm11.jar` with the edited `nas.properties` file with the following command:

   ```
   jar uf rnpfm11.jar nas.properties
   ```

6. Updating `rnpfmxx.war`

7. To extract the `nas.properties` file from `rnpfmxx.war` use the following command:

   ```
   jar xf rnpfmxx.war WEB-INF/nas.properties
   ```

8. Open the `WEB-INF/nas.properties` file in a text editor.

9. Using a valid user name and password from the domain (configured in the Basic Authentication of Step 5 Configuring the EFCService to use Basic Authentication, item 5., above). Change the values, the text to the right of the equal sign, with the following properties:

   - `httpusername=`
   - `httppassword=`

   9.1. Update the `rnpfmxx.war` with the edited `nas.properties` file with the following command:

   ```
   jar uf rnpfmxx.war WEB-INF/nas.properties
   ```

10. Cleanup by deleting the following temporary file and directory.

    - `nas.properties`
    - WEB-INF (directory and its content)

Step 9 Updating the infinity.keystore (optional)

Note

If **Setup EFCService for SSL** is configured in Step 6 Setup EFCService for SSL (optional), above, this step must be completed.

1. There are two `infinity.keystore` files that must be updated with the certificate. One is located as a standalone file and is used by `rnpfm11.jar`. The other is embedded within the `rnpfmxx.war` file.

2. Login into the environment where Banner Financial Aid is installed.

   2.1. Navigate to the `<JAVA_HOME>/bin` directory.

   2.2. Copy the `<BANNER_HOME>/finaid/java/infinity.keystore` to this location.

   2.3. Copy the institution new certificate to this same location.

2.4. Use the following command to add the certificate to the `infinity.keystore`.

Note

Note, replace the filename `new_certificate.cer` with the correct file name of the certificate before running the command.

```
keytool -import -alias myschool -file new_certificate.cer -
keystore infinity.keystore -storepass u_pick_it -noprompt
```

2.5. Copy the newly update `infinity.keystore` back to:

`<BANNER_HOME>/finaid/java/infinity.keystore`.

3. Updating `rnpfmxx.war`

3.1. Navigate to `<BANNER_HOME>/finaid/java/`.

3.2. Create a sub directory `WEB-INF`.

3.3. Copy the updated `infinity.keystore`, from item 2.5., above, to:

`WEB-INF`.

3.4. Update the `rnpfmxx.war` with the updated infinity.keystore file with the following command:

```
jar uf rnpfmxx.war WEB-INF/infinity.keystore
```

4. Cleanup by deleting the temporary directory `WEB-INF` and its contents.

Installing IIS 7.0 on Windows Server 2008

Overview

These instructions are used to configure your system, using IIS 7.0 on Windows 2008 to allow On-Premise (Local) FM Calculations.

Step 1 Minimum Requirements

1. The On-Premise install of the EFCService Web Service can only be done on a Windows Server:

1.1. Windows 2008 running IIS 7.

1.2. Microsoft.NET Framework 3.5.

1.3. IIS configured to run ASP.net 2.0.

Step 2 Installing IIS 7.0 Windows Server 2008

1. Detailed instructions are available from Microsoft at the following URL:

   `http://technet.microsoft.com/en-us/library/cc771209(WS.10).aspx?ppud=4`

   1.1. Click Start, point to Administrative Tools, and click Server Manager.

   1.2. In Roles Summary, click Add Roles.

   1.3. Use the Add Roles Wizard to add the Web Server role.

   1.4. During the Select Role Services – check to include the following items:

   - ASP.NET
   - Basic Authentication

Step 3 Install EFCService Web Service

1. Unzip the contents of the `SGHE_EFCService_Install_10292010.zip` file into the following folder:

   `C:\inetpub\wwwroot\EFCService`

The structure includes:

```
..\EFCService\App_Data\Config.xml
              \bin\EFC.dll
                  \EFCSoapExtension.dll
                  \ExceptionHandling.dll
                  \FederalEFC1011.dll
                  \FederalEFC.dll
                  \FinancialAidService.dll
              \efc_calculation_messages.xsd
              \EFCCalculationService.asmx
              \Web.config
```

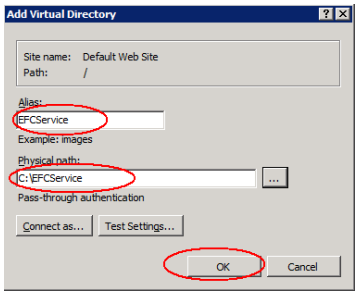
2. If the installed folder is the suggested folder, `C:\inetpub\wwwroot\EFCService`, skip directly to item 4., below.

3. Open the files `Config.xml` and `Web.config` in a text editor and replace every instance of `C:\inetpub\wwwroot\EFCService` with the path of the actual installed folder. For example, if the installed folder is `C:\EFCService`, do a search and replace of `C:\inetpub\wwwroot\EFCService` with `C:\EFCService`.

4. From the Server Manager, expand Web Server (IIS) and select Internet Information Service (IIS) Manager.

5. From the Connection screen, right click on the Default Web Site and select Add Virtual Directory.

5.1. Complete the **Add Virtual Directory** pop-up dialog. Make sure to use EFCService for the Alias. Click the Add Virtual Directory's **OK** button to save your settings.



6. From the Server Manager, expand Web Server (IIS) and select Internet Information Service (IIS) Manager.

   6.1. From the Connections screen, select Default Web Site and do a Refresh by pressing the F5 function key on the keyboard.

   6.2. The EFCService Folder will be displayed.

   6.3. Right click on the EFCService folder and select from the popup menu Convert to Application.

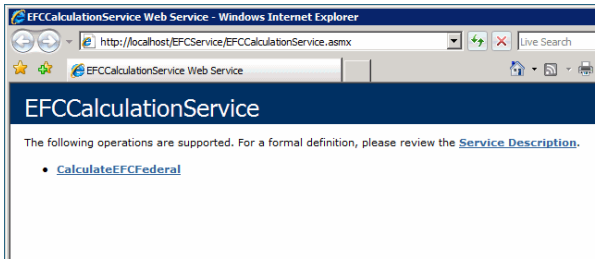
   6.4. Select OK.

Step 4 Web Service Test

1. From a Web Browser, go to the following URL

   ```
   :http://localhost/EFCService/EFCCalculationService.asmx
   ```

2. The EFCCalculationService page will be displayed:

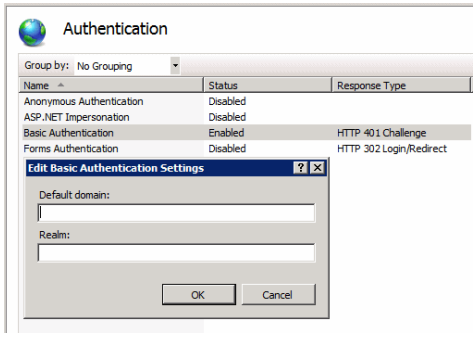


Step 5 Configuring the EFCService to use Basic Authentication (optional)

Note

This step is optional. If this step is configured, Step 8 Updating the user name and password to connect to the EFCService (optional), below, must also be configured.

1. Navigate back to the Default Web Site on the Internet Information Service (IIS) Manager.
2. With the Features View on, select **Authentication**.
3. Disable **Anonymous Authentication**.
4. Enable **Basic Authentication**.
5. Edit **Basic Authentication** and supply **Default domain** and **Realm** data.



Step 6 Setup EFCService for SSL (optional)

Note

This step is optional. If this step is configured, Step 9 Updating the infinity.keystore (optional), below, must also be configured.

1. SSL requires obtaining and registering a certificate with the Web Server.
2. The bit length must be set to a value of 1024.
3. Detailed instructions are available from Microsoft at the following URL:

```
http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/
Library/IIS/89c7ef2f-f7d6-483c-8b08-ae0c6584dd4d.mspx?mfr=true
```

Step 7 Updating INB to see the On-Premise FM Calculation Web Service

1. Login into INB and navigate to the Project Based Parameter Setup Form (RORPARM).

2. For parameter `WS_ENDPOINT`, replace the default value with the URL from Step 4 Web Service Test, above:

   ```
   http://localhost/EFCService/EFCCalculationService.asmx
   ```

3. Make sure to replace `localhost` with the correct address of the IIS.

Step 8 Updating the user name and password to connect to the EFCService (optional)

Note

If Basic Authentication is configured in Step 5 Configuring the EFCService to use Basic Authentication (optional), above, this step must be completed.

1. Login into the environment where Banner Financial Aid is installed. Navigate to:

   ```
   <BANNER_HOME>/finaid/java/
   ```

2. Updating `rnpfm11.jar`

3. To extract the `nas.properties` file from `rnpfm11.jar`, use the following command:

   ```
   jar xf rnpfm11.jar nas.properties
   ```

4. Open the `nas.properties` file in a text editor.

5. Using a valid user name and password from the domain (configured in the Basic Authentication of Step 5 Configuring the EFCService to use Basic Authentication, item 5., above). Change the values, the text to the right of the equal sign, with the following properties:

   - `httpusername=`
   - `httppassword=`

   5.1. Update the `rnpfm11.jar` with the edited `nas.properties` file with the following command:

   ```
   jar uf rnpfm11.jar nas.properties
   ```

6. Updating `rnpfmxx.war`

7. To extract the `nas.properties` file from `rnpfmxx.war` use the following command:

   ```
   jar xf rnpfmxx.war WEB-INF/nas.properties
   ```

8. Open the `WEB-INF/nas.properties` file in a text editor.

9. Using a valid user name and password from the domain (configured in the Basic Authentication of Step 5 Configuring the EFCService to use Basic Authentication, item 5., above). Change the values, the text to the right of the equal sign, with the following properties:

   - `httpusername=`
   - `httppassword=`

   9.1. Update the `rnpfmxx.war` with the edited `nas.properties` file with the following command:

   ```
   jar uf rnpfmxx.war WEB-INF/nas.properties
   ```

10. Cleanup by deleting the following temporary file and directory.

    - `nas.properties`
    - WEB-INF (directory and its content)

Step 9 Updating the infinity.keystore (optional)

Note

If **Setup EFCService for SSL** is configured in Step 6 Setup EFCService for SSL (optional), above, this step must be completed.

1. There are two `infinity.keystore` files that must be updated with the certificate. One is located as a standalone file and is used by `rnpfm11.jar`. The other is embedded within the `rnpfmxx.war` file.

2. Login into the environment where Banner Financial Aid is installed.

   2.1. Navigate to the `<JAVA_HOME>/bin` directory.

   2.2. Copy the `<BANNER_HOME>/finaid/java/infinity.keystore` to this location.

   2.3. Copy the institution new certificate to this same location.

2.4. Use the following command to add the certificate to the `infinity.keystore`.

Note

Note, replace the filename `new_certificate.cer` with the correct file name of the certificate before running the command.

```
keytool -import -alias myschool -file new_certificate.cer -
keystore infinity.keystore -storepass u_pick_it -noprompt
```

2.5. Copy the newly update `infinity.keystore` back to:

`<BANNER_HOME>/finaid/java/infinity.keystore`.

3. Updating `rnpfmxx.war`

3.1. Navigate to `<BANNER_HOME>/finaid/java/`.

3.2. Create a sub directory `WEB-INF`.

3.3. Copy the updated `infinity.keystore`, from item 2.5., above, to:

`WEB-INF`.

3.4. Update the `rnpfmxx.war` with the updated infinity.keystore file with the following command:

```
jar uf rnpfmxx.war WEB-INF/infinity.keystore
```

4. Cleanup by deleting the temporary directory `WEB-INF` and its contents.

FM Need Analysis Schema

The following code represents the current FM Need Analysis Schema for aid year 2010-2011:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- edited with XMLSpy v2005 rel. 3 U (http://www.altova.com) by
Martin Schwartz (Sungard SCT) -->
<xs:schema
xmlns:finaid="urn:sungardhe:enterprise:financial_aid:messages"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:sungardhe:enterprise:financial_aid:messages"
elementFormDefault="qualified" attributeFormDefault="unqualified"
version="1.0">
  <!-- Input XML - Root Element Structure -->
  <xs:element name="GetEFCCalculation">
    <xs:complexType>
      <xs:sequence>
```

```
        <xs:element name="ISIR" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="TransactionNumber">
                <xs:simpleType>
                  <xs:restriction base="xs:integer">
                    <xs:minInclusive value="00"/>
                    <xs:maxInclusive value="99"/>
                    <xs:fractionDigits value="0"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element ref="finaid:ExternalRecordID"/>
              <xs:element name="AwardYear"
type="finaid:AwardYearType"/>
              <xs:element ref="finaid:EFCDuration" minOccurs="0"/>
              <xs:element name="OverrideRejects"
type="finaid:YesNoType" minOccurs="0"/>
              <xs:element ref="finaid:Student"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- Output XML - Root Element Structure -->
  <xs:element name="ShowEFCCalculation">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="finaid:EFCEstimate" maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>1 per ISIR record passed to the
service</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- Input XML elements Definitions-->
  <xs:element name="ExternalRecordID" type="xs:string"/>
  <xs:element name="Student" type="finaid:ISIRStudentType"/>
  <xs:complexType name="ISIRStudentType">
    <xs:sequence>
      <xs:element name="BirthDate" type="finaid:DateBlankType"
minOccurs="0"/>
```

```
      <xs:element name="LastNameExists" type="finaid:NameExistsType"
minOccurs="0"/>
      <xs:element name="FirstNameExists" type="finaid:NameExistsType"
minOccurs="0"/>
      <xs:element name="AddressStateProvinceCode"
type="finaid:StateProvinceCodeType" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Contact Address</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="CitizenshipStatusCode"
type="finaid:CitizenshipStatusCodeType" minOccurs="0"/>
      <xs:element name="MaritalStatusCode"
type="finaid:StudentMaritalStatusCodeType" minOccurs="0"/>
     <xs:element name="MaritalStatusDate" type="finaid:DateBlankType"
minOccurs="0"/>
      <xs:element name="ResidencyStateProvinceCode"
type="finaid:StateProvinceCodeType" minOccurs="0"/>
     <xs:element name="HouseholdData" type="finaid:HouseholdDataType"
minOccurs="0"/>
      <xs:element name="SubmissionInformation"
type="finaid:SubmissionInformationType" minOccurs="0"/>
      <xs:element name="ProcessedApplicationInformation"
type="finaid:ProcessedApplicationInformationType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="HouseholdDataType">
    <xs:sequence>
      <xs:element name="MembersInFamily"
type="finaid:Integer2DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="NumberInCollege"
type="finaid:Integer2DigitsWithBlankType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="SubmissionInformationType">
    <xs:sequence>
      <xs:element name="AdditionalStudentData"
type="finaid:AdditionalStudentDataType" minOccurs="0"/>
      <xs:element name="ParentData" type="finaid:ParentDataType"
minOccurs="0"/>
      <xs:element name="AidAdministratorData"
type="finaid:AidAdministratorDataType" minOccurs="0"/>
      <xs:element name="DateApplicationCompleted"
type="finaid:DateBlankType" minOccurs="0"/>
      <xs:element name="SignatureSourceCode"
type="finaid:SignatureSourceType" minOccurs="0"/>
    </xs:sequence>
```

```
  </xs:complexType>
  <xs:complexType name="AdditionalStudentDataType">
    <xs:sequence>
     <xs:element name="FinancialData" type="finaid:FinancialDataType"
minOccurs="0"/>
      <xs:element name="QuestionnaireResponses"
type="finaid:QuestionnaireResponsesType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="FinancialDataType">
    <xs:sequence>
      <xs:element name="IncomeData" type="finaid:IncomeDataType"
minOccurs="0"/>
      <xs:element name="AssetData" type="finaid:AssetDataType"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="IncomeDataType">
    <xs:sequence>
      <xs:element name="TaxReturnStatusCode"
type="finaid:TaxReturnStatusCodeType" minOccurs="0"/>
      <xs:element name="TaxFormTypeCode"
type="finaid:TaxFormTypeCodeType" minOccurs="0"/>
      <xs:element name="Form1040AEZEligibleCode"
type="finaid:Form1040AEZEligibleCodeType" minOccurs="0"/>
      <xs:element name="AdjustedGrossIncome"
type="finaid:SignedInteger6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="IncomeTaxPaid"
type="finaid:Integer6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ExemptionsClaimed"
type="finaid:Integer2DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="StudentEarnedIncome"
type="finaid:SignedInteger6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="SpouseEarnedIncome"
type="finaid:SignedInteger6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="FathersEarnedIncome"
type="finaid:SignedInteger6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="MothersEarnedIncome"
type="finaid:SignedInteger6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="EducationCredits"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ChildSupportPaid"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="NeedBasedEmployment"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="GrantScholarshipAid"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
```

```
      <xs:element name="CombatPay"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="CoopEarnings"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="PensionPayments"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="IRAPayments"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ChildSupportReceived"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="InterestIncome"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="IRADistributions"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="UntaxedPensions"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="MilitaryClergyAllowances"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="VeteranNonEducationBenefits"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="OtherUntaxedIncome"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="OtherNonReportedMoneyReceived"
type="finaid:Integer5DigitsWithBlankType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AssetDataType">
    <xs:sequence>
      <xs:element name="CashSavingsChecking"
type="finaid:Integer6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="InvestmentNetWorth"
type="finaid:Integer6DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="BusinessFarmNetWorth"
type="finaid:Integer6DigitsWithBlankType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="QuestionnaireResponsesType">
    <xs:sequence>
      <xs:element name="HighSchoolDiplomaOrGED"
type="finaid:HighSchoolDiplomaOrGEDType" minOccurs="0"/>
      <xs:element name="FirstBachelorsByCutoff"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="SSIBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="FoodStamps" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="FreeReducedPriceLunch"
type="finaid:YesNoBlankType" minOccurs="0"/>
```

```
      <xs:element name="TANFBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="WICBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
     <xs:element name="DislocatedWorker" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="DependencyData"
type="finaid:DependencyQuestionnaireResponsesType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DependencyQuestionnaireResponsesType">
    <xs:sequence>
      <xs:element name="BornBeforeCutoffIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="MastersOrDoctorateIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
     <xs:element name="MarriedIndicator" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="SupportsChildrenIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="LegalDependentsIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="OrphanWardOfCourtIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="ActiveDutyMilitary"
type="finaid:YesNoBlankType" minOccurs="0"/>
     <xs:element name="EmancipatedMinor" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="InLegalGuardianship"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="HomelessYouthSchool"
type="finaid:YesNoBlankType" minOccurs="0"/>
     <xs:element name="HomelessYouthHUD" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="HomelessRisk" type="finaid:YesNoBlankType"
minOccurs="0"/>
     <xs:element name="VeteranIndicator" type="finaid:YesNoBlankType"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ParentDataType">
    <xs:sequence>
      <xs:element name="FatherStepfatherBirthDate"
type="finaid:DateBlankType" minOccurs="0"/>
      <xs:element name="FatherStepfatherSSNExists"
type="finaid:SSNExistsType" minOccurs="0"/>
      <xs:element name="FatherStepfatherLastNameExists"
type="finaid:NameExistsType" minOccurs="0"/>
```

```
      <xs:element name="MotherStepmotherBirthDate"
type="finaid:DateBlankType" minOccurs="0"/>
      <xs:element name="MotherStepmotherSSNExists"
type="finaid:SSNExistsType" minOccurs="0"/>
      <xs:element name="MotherStepmotherLastNameExists"
type="finaid:NameExistsType" minOccurs="0"/>
      <xs:element name="MaritalStatusCode"
type="finaid:MaritalStatusCodeType" minOccurs="0"/>
      <xs:element name="ResidencyStateProvinceCode"
type="finaid:StateProvinceCodeType" minOccurs="0"/>
      <xs:element name="SSIBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="FoodStamps" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="FreeReducedPriceLunch"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="TANFBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
      <xs:element name="WICBenefits" type="finaid:YesNoBlankType"
minOccurs="0"/>
     <xs:element name="DislocatedWorker" type="finaid:YesNoBlankType"
minOccurs="0"/>
     <xs:element name="HouseholdData" type="finaid:HouseholdDataType"
minOccurs="0"/>
      <xs:element name="ParentFinancialData"
type="finaid:ParentFinancialDataType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AidAdministratorDataType">
    <xs:sequence>
      <xs:element name="EFCAdjustmentCode"
type="finaid:EFCAdjustmentCodeType" minOccurs="0"/>
      <xs:element name="DependencyOverrideCode"
type="finaid:DependencyOverrideCodeType" minOccurs="0"/>
      <xs:element name="RejectOverrides"
type="finaid:RejectOverridesType" minOccurs="0"/>
      <xs:element name="AssumptionOverrides"
type="finaid:AssumptionOverridesType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ParentFinancialDataType">
    <xs:sequence>
      <xs:element name="IncomeData" type="finaid:IncomeDataType"
minOccurs="0"/>
      <xs:element name="AssetData" type="finaid:AssetDataType"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
```

```
  <xs:complexType name="RejectOverridesType">
    <xs:sequence>
      <xs:element name="RejectOverrideCode"
type="finaid:RejectOverrideCodeType" minOccurs="0" maxOccurs="18"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AssumptionOverridesType">
    <xs:sequence>
      <xs:element name="AssumptionOverrideCode"
type="finaid:AssumptionOverrideCodeType" minOccurs="0" maxOccurs="6"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ProcessedApplicationInformationType">
    <xs:sequence>
      <xs:element name="DependencyStatusCode"
type="finaid:DependencyStatusCodeType" minOccurs="0"/>
      <xs:element name="GraduateIndicator"
type="finaid:YesNoBlankType" minOccurs="0"/>
      <xs:element name="StudentTaxFilingStatusCode"
type="finaid:TaxFilingStatusCodeType" minOccurs="0"/>
      <xs:element name="ParentTaxFilingStatusCode"
type="finaid:TaxFilingStatusCodeType" minOccurs="0"/>
      <xs:element name="MotherStepmotherSSNMatch" type="xs:string"
minOccurs="0"/>
      <xs:element name="FatherStepfatherSSNMatch" type="xs:string"
minOccurs="0"/>
      <xs:element name="StudentSSNMatch" type="xs:string"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <!-- Output XML elements Definitions -->
  <xs:element name="EFCEstimate">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ResponseHeader">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="finaid:ExternalRecordID"/>
              <xs:element name="Response" type="xs:string"/>
              <xs:element name="EFC" type="xs:int" minOccurs="0"/>
              <xs:element name="EFCDate" type="xs:dateTime"
minOccurs="0"/>
              <xs:element ref="finaid:EFCDuration" minOccurs="0"/>
              <xs:element name="IsZeroEFC" type="xs:boolean"
minOccurs="0"/>
              <xs:element name="FormulaName" minOccurs="0">
```

```
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="Dependent"/>
                    <xs:enumeration value="Independent Without
Dependents"/>
                    <xs:enumeration value="Independent With
Dependents"/>
                    <xs:enumeration value="Simple Dependent"/>
                    <xs:enumeration value="Simple Independent Without
Dependents"/>
                    <xs:enumeration value="Simple Independent With
Dependents"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element name="IsFederal" type="xs:boolean"
minOccurs="0"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Assumptions" type="finaid:AssumptionsType"
minOccurs="0"/>
        <xs:element name="RejectReasons"
type="finaid:RejectReasonsType" minOccurs="0"/>
        <xs:element name="EFCCalculations"
type="finaid:EFCIntermediateValuesType" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="EFCIntermediateValuesType">
    <xs:sequence>
      <xs:element name="TotalIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="TotalIncomeAllowances"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="StateTaxAllowance"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="ParentSocialSecurityTax"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="EmploymentAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="IncomeProtectionAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="AvailableIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AvailableIncomeContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
```

```
      <xs:element name="DiscretionaryNetWorth"
type="finaid:SignedInteger9DigitsType" minOccurs="0"/>
      <xs:element name="NetWorth" type="finaid:Integer9DigitsType"
minOccurs="0"/>
      <xs:element name="AssetProtectionAllowance"
type="finaid:Integer9DigitsType" minOccurs="0"/>
      <xs:element name="ParentAssetContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AdjustedAvailableIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="TotalStudentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="TotalParentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeTotalParentContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="ParentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="StudentTotalIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentSocialSecurityTax"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentStateTaxAllowance"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="NegativeAdjustableIncomeOffset"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentIncomeAllowances"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentIncomeContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="StudentDiscretionaryNetWorth"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentAssetContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="FISAPTotalIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="NineMonthEFC"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="RejectReasonsType">
    <xs:sequence>
      <xs:element name="RejectCode" type="finaid:RejectCodeType"
minOccurs="0" maxOccurs="32"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AssumptionsType">
    <xs:sequence>
```

```
      <xs:element name="Assumption" type="finaid:AssumptionType"
minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AssumptionType">
    <xs:sequence>
      <xs:element name="FieldName" type="finaid:FieldNameType"/>
      <xs:element name="FieldValue" type="finaid:FieldValueType"/>
    </xs:sequence>
  </xs:complexType>
  <!-- Simple Types-->
  <xs:simpleType name="AwardYearType">
    <xs:restriction base="xs:int"/>
  </xs:simpleType>
  <xs:element name="EFCDuration">
    <xs:simpleType>
      <xs:restriction base="xs:byte">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="12"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:simpleType name="YesNoType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Y"/>
      <xs:enumeration value="N"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="CitizenshipStatusCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value=""/>
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="StudentMaritalStatusCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value=""/>
    </xs:restriction>
  </xs:simpleType>
```

```
<xs:simpleType name="MaritalStatusCodeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
    <xs:enumeration value=""/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="TaxReturnStatusCodeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value=""/>
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="TaxFormTypeCodeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value=""/>
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Form1040AEZEligibleCodeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value=""/>
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="HighSchoolDiplomaOrGEDType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
    <xs:enumeration value=""/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="DependencyOverrideCodeType">
  <xs:restriction base="xs:string">
```

```
      <xs:enumeration value="Requested"/>
      <xs:enumeration value="Cancelled"/>
      <xs:enumeration value="Failed"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="EFCAdjustmentCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Processed"/>
      <xs:enumeration value="Failed"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="RejectOverrideCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="A"/>
      <xs:enumeration value="B"/>
      <xs:enumeration value="C"/>
      <xs:enumeration value="G"/>
      <xs:enumeration value="N"/>
      <xs:enumeration value="W"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="12"/>
      <xs:enumeration value="20"/>
      <xs:enumeration value="J"/>
      <xs:enumeration value="K"/>
      <xs:enumeration value="D"/>
      <xs:enumeration value="E"/>
      <xs:enumeration value="F"/>
      <xs:enumeration value="R"/>
      <xs:enumeration value="S"/>
      <xs:enumeration value="T"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="AssumptionOverrideCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="5"/>
      <xs:enumeration value="6"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SignatureSourceType">
    <xs:restriction base="xs:string">
```

```
      <xs:enumeration value="ApplicantOnly"/>
      <xs:enumeration value="ApplicantAndParent"/>
      <xs:enumeration value="ParentOnly"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="DependencyStatusCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Dependent"/>
      <xs:enumeration value="Independent"/>
      <xs:enumeration value="DependentNoEFC"/>
      <xs:enumeration value="IndependentNoEFC"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SSNExistsType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="E"/>
      <xs:enumeration value="B"/>
      <xs:enumeration value="O"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="NameExistsType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="E"/>
      <xs:enumeration value="B"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="StateProvinceCodeType">
    <xs:restriction base="xs:token">
      <xs:enumeration value="AA">
        <xs:annotation>
          <xs:documentation>MILITARY-AMERICAS</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="AB">
        <xs:annotation>
          <xs:documentation>ALBERTA</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="AE">
        <xs:annotation>
          <xs:documentation>MILITARY-EUROPE</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="AK">
        <xs:annotation>
```

```
      <xs:documentation>ALASKA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="AL">
    <xs:annotation>
      <xs:documentation>ALABAMA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="AP">
    <xs:annotation>
      <xs:documentation>MILITARY-PACIFIC</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="AR">
    <xs:annotation>
      <xs:documentation>ARKANSAS</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="AS">
    <xs:annotation>
      <xs:documentation>AMERICAN SAMOA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="AZ">
    <xs:annotation>
      <xs:documentation>ARIZONA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="BC">
    <xs:annotation>
      <xs:documentation>BRITISH COLUMBIA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="CA">
    <xs:annotation>
      <xs:documentation>CALIFORNIA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="CO">
    <xs:annotation>
      <xs:documentation>COLORADO</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="CT">
    <xs:annotation>
```

```
            <xs:documentation>CONNECTICUT</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="CZ">
          <xs:annotation>
            <xs:documentation>CANAL ZONE</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="DC">
          <xs:annotation>
            <xs:documentation>DISTRICT OF COLUMBIA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="DE">
          <xs:annotation>
            <xs:documentation>DELAWARE</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="FL">
          <xs:annotation>
            <xs:documentation>FLORIDA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="FM">
          <xs:annotation>
            <xs:documentation>FEDERATED STATES OF MICRONESIA</
xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="GA">
          <xs:annotation>
            <xs:documentation>GEORGIA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="GU">
          <xs:annotation>
            <xs:documentation>GUAM</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="HI">
          <xs:annotation>
            <xs:documentation>HAWAII</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="IA">
```

```
    <xs:annotation>
      <xs:documentation>IOWA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="ID">
    <xs:annotation>
      <xs:documentation>IDAHO</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="IL">
    <xs:annotation>
      <xs:documentation>ILLINOIS</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="IN">
    <xs:annotation>
      <xs:documentation>INDIANA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="KS">
    <xs:annotation>
      <xs:documentation>KANSAS</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="KY">
    <xs:annotation>
      <xs:documentation>KENTUCKY</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="LA">
    <xs:annotation>
      <xs:documentation>LOUISIANA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="MA">
    <xs:annotation>
      <xs:documentation>MASSACHUSETTS</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="MB">
    <xs:annotation>
      <xs:documentation>MANITOBA</xs:documentation>
    </xs:annotation>
  </xs:enumeration>
  <xs:enumeration value="MD">
```

```
            <xs:annotation>
              <xs:documentation>MARYLAND</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="ME">
            <xs:annotation>
              <xs:documentation>MAINE</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MH">
            <xs:annotation>
              <xs:documentation>MARSHALL ISLANDS</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MI">
            <xs:annotation>
              <xs:documentation>MICHIGAN</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MN">
            <xs:annotation>
              <xs:documentation>MINNESOTA</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MO">
            <xs:annotation>
              <xs:documentation>MISSOURI</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MP">
            <xs:annotation>
              <xs:documentation>NORTHERN MARIANA ISLANDS</
xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MS">
            <xs:annotation>
              <xs:documentation>MISSISSIPPI</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
          <xs:enumeration value="MT">
            <xs:annotation>
              <xs:documentation>MONTANA</xs:documentation>
            </xs:annotation>
          </xs:enumeration>
```

```
        <xs:enumeration value="NB">
          <xs:annotation>
            <xs:documentation>NEW BRUNSWICK</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NC">
          <xs:annotation>
            <xs:documentation>NORTH CAROLINA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="ND">
          <xs:annotation>
            <xs:documentation>NORTH DAKOTA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NE">
          <xs:annotation>
            <xs:documentation>NEBRASKA</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NF">
          <xs:annotation>
            <xs:documentation>NEWFOUNDLAND</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NH">
          <xs:annotation>
            <xs:documentation>NEW HAMPSHIRE</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NJ">
          <xs:annotation>
            <xs:documentation>NEW JERSEY</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NL">
          <xs:annotation>
            <xs:documentation>NEWFOUNDLAND AND LABRADOR</
xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NM">
          <xs:annotation>
            <xs:documentation>NEW MEXICO</xs:documentation>
          </xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="NS">
  <xs:annotation>
    <xs:documentation>NOVA SCOTIA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="NT">
  <xs:annotation>
    <xs:documentation>NORTHWEST TERRITORIES</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="NU">
  <xs:annotation>
    <xs:documentation>NUNAVUT</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="NV">
  <xs:annotation>
    <xs:documentation>NEVADA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="NY">
  <xs:annotation>
    <xs:documentation>NEW YORK</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="OH">
  <xs:annotation>
    <xs:documentation>OHIO</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="OK">
  <xs:annotation>
    <xs:documentation>OKLAHOMA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="ON">
  <xs:annotation>
    <xs:documentation>ONTARIO</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="OR">
  <xs:annotation>
    <xs:documentation>OREGON</xs:documentation>
  </xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="PA">
  <xs:annotation>
    <xs:documentation>PENNSYLVANIA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="PE">
  <xs:annotation>
    <xs:documentation>PRINCE EDWARD ISLAND</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="PR">
  <xs:annotation>
    <xs:documentation>PUERTO RICO</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="PW">
  <xs:annotation>
    <xs:documentation>REPUBLIC OF PALAU</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="QC">
  <xs:annotation>
    <xs:documentation>QUEBEC</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="RI">
  <xs:annotation>
    <xs:documentation>RHODE ISLAND</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="SC">
  <xs:annotation>
    <xs:documentation>SOUTH CAROLINA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="SD">
  <xs:annotation>
    <xs:documentation>SOUTH DAKOTA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="SK">
  <xs:annotation>
    <xs:documentation>SASKATCHEWAN</xs:documentation>
  </xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="TN">
  <xs:annotation>
    <xs:documentation>TENNESSEE</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="TX">
  <xs:annotation>
    <xs:documentation>TEXAS</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="UT">
  <xs:annotation>
    <xs:documentation>UTAH</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="VA">
  <xs:annotation>
    <xs:documentation>VIRGINIA</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="VI">
  <xs:annotation>
    <xs:documentation>VIRGIN ISLANDS</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="VT">
  <xs:annotation>
    <xs:documentation>VERMONT</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="WA">
  <xs:annotation>
    <xs:documentation>WASHINGTON</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="WI">
  <xs:annotation>
    <xs:documentation>WISCONSIN</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="WV">
  <xs:annotation>
    <xs:documentation>WEST VIRGINIA</xs:documentation>
  </xs:annotation>
```

```
        </xs:enumeration>
        <xs:enumeration value="WY">
          <xs:annotation>
            <xs:documentation>WYOMING</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="YT">
          <xs:annotation>
            <xs:documentation>YUKON</xs:documentation>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="CN"/>
        <xs:enumeration value="FC"/>
        <xs:enumeration value="PQ"/>
        <xs:enumeration value="MX"/>
        <xs:enumeration value=""/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TaxFilingStatusCodeType">
      <xs:restriction base="xs:string">
        <xs:enumeration value="TaxFiler"/>
        <xs:enumeration value="NonTaxFiler"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="BlankType">
      <xs:restriction base="xs:string">
        <xs:enumeration value=""/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Integer1DigitType">
      <xs:restriction base="xs:integer">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="9"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Integer1DigitWithBlankType">
      <xs:union memberTypes="finaid:Integer1DigitType finaid:BlankType"/
>
    </xs:simpleType>
    <xs:simpleType name="Integer2DigitsType">
      <xs:restriction base="xs:integer">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="99"/>
      </xs:restriction>
    </xs:simpleType>
```

```
  <xs:simpleType name="Integer2DigitsWithBlankType">
    <xs:union memberTypes="finaid:Integer2DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="Integer5DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="99999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Integer5DigitsWithBlankType">
    <xs:union memberTypes="finaid:Integer5DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="Integer6DigitsType">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="999999"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Integer6DigitsWithBlankType">
    <xs:union memberTypes="finaid:Integer6DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger6DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="-999999"/>
      <xs:maxInclusive value="999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger6DigitsWithBlankType">
    <xs:union memberTypes="finaid:SignedInteger6DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="Integer7DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="9999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger7DigitsType">
    <xs:restriction base="xs:decimal">
```

```
      <xs:minInclusive value="-9999999"/>
      <xs:maxInclusive value="9999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Integer9DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="999999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger9DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="-999999999"/>
      <xs:maxInclusive value="999999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="YesNoBlankType">
    <xs:union memberTypes="xs:boolean finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="RejectCodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="5"/>
      <xs:enumeration value="6"/>
      <xs:enumeration value="7"/>
      <xs:enumeration value="8"/>
      <xs:enumeration value="9"/>
      <xs:enumeration value="10"/>
      <xs:enumeration value="11"/>
      <xs:enumeration value="12"/>
      <xs:enumeration value="13"/>
      <xs:enumeration value="14"/>
      <xs:enumeration value="15"/>
      <xs:enumeration value="17"/>
      <xs:enumeration value="18"/>
      <xs:enumeration value="20"/>
      <xs:enumeration value="A"/>
      <xs:enumeration value="B"/>
      <xs:enumeration value="C"/>
```

```
        <xs:enumeration value="D"/>
        <xs:enumeration value="E"/>
        <xs:enumeration value="F"/>
        <xs:enumeration value="G"/>
        <xs:enumeration value="J"/>
        <xs:enumeration value="K"/>
        <xs:enumeration value="N"/>
        <xs:enumeration value="R"/>
        <xs:enumeration value="S"/>
        <xs:enumeration value="T"/>
        <xs:enumeration value="W"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="FieldNameType">
      <xs:restriction base="xs:string">
        <xs:maxLength value="50"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="FieldValueType">
      <xs:restriction base="xs:string">
        <xs:maxLength value="50"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="DateBlankType">
      <xs:union memberTypes="xs:date finaid:BlankType"/>
    </xs:simpleType>
  </xs:schema>
```