ellucian

Banner Financial Aid
Hosted Services Handbook

January 2015

Ellucian
4375 Fair Lakes Court
Fairfax, VA 22033
United States of America

Contents

Introduction

Banner Financial Aid Hosted Services

This handbook describes the variety of external hosted services offered by Ellucian in support of the Banner Financial Aid product. Each chapter contains specific installation, configuration, and administrative materials.

Additionally, this handbook also includes other supplemental details in related Appendices.

Descriptions of the following hosted services are provided:

- FM Need Analysis Off-Premise Configuration
- FM Need Analysis On-Premise Configuration

Documentation corrections/clarifications

Any corrections/clarifications to this documentation that arise, subsequent to its posting, will be documented in the *Are there any corrections/clarifications to the documentation for Banner Financial Aid Hosted Services Handbook January 2015?*, Article # 000031802, and made available through the Ellucian Support Center (http://www.ellucian.com/Solutions/Ellucian-Client-Support/). Please refer to the Article periodically as you use to this document.

FM Need Analysis Off-Premise Configuration

Introduction

This chapter supplies technical details associated with installation and configuration of the hosted FM Need Analysis solution. Configurations, both required and optional, are included to achieve successful 2011-2012 FM Need Analysis calculations associated with release 8.10, or above. As such, this chapter is written with references specific to the RNPFM12 object. However, the steps associated with RNPFM12 can also be applied to subsequent aid year-specific versions of the RNPFMyy object.

To support the immediate FM Need Analysis calculation, the following two options have been made available:

- Job submission
- Advanced Queuing (AQ) - Available for year-specific RNPFMyy processes (beginning with RNPFM13).

AQ has been made available as an optional installation that can improve performance of online calculations. AQ runs on the job submission server, the same environment that runs batch processing. Refer to the Advanced Queuing section for additional details.

Note: Information found in this chapter supersedes all configuration requirements noted in prior release documentation.

Note: Issues regarding the hosted Banner Financial Aid FM Need Analysis solution should be submitted to the Ellucian Action Line under the Product Line: *Financial Aid* and the Product: *FM Need Analysis*.

When submitting a Case from the Ellucian Support Center, select the correct Product Line (`Financial Aid`) and Product (`FM Need Analysis`) options from the available drop down lists.

Object List

The following objects are associated with FM Need Analysis:

- `finaiddbseed.jar`

- `finaidfmlogin.jar`
- `finaidutils.jar`
- `fm_ellucian.cer`
- `rnpfmyy.jar *`
- `rnpfmyy.pl *`
- `rnpfmyy.shl *`
- `rnpfmxx_plan.dat`

***** - Where "yy" is the aid year.

Note: In INB, access to the RORPARM form will also be needed to update references to the servlet.

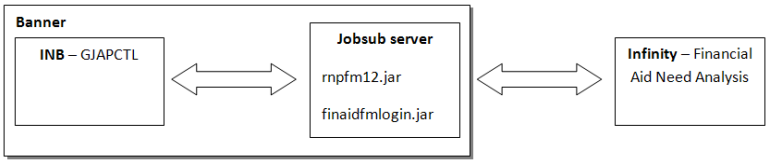
Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Configuration Options

Several options are available for running FM Need Analysis. The following diagrams illustrate the various FM Need Analysis configuration options:

Batch Need Analysis

The most efficient way to process large numbers of students for FM Need Analysis is by using job submission to run the RNPFMyy (where "yy" is the aid year) process. Therefore, this is a required configuration:

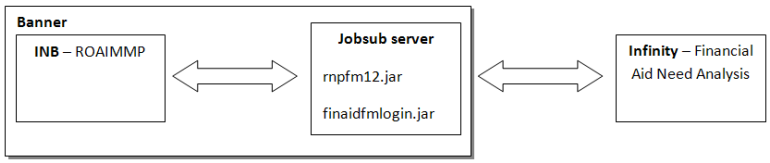


Immediate Processing (Job Submission Process vs. Advanced Queuing)

There are two configuration options available to perform immediate student processing for FM Need Analysis.

Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Running immediate FM Need Analysis using the job submission and AQ:



Running immediate FM Need Analysis using Advanced Queuing. Refer to section on Advanced Queuing for more information.

While better performance is noted using the immediate FM Need Analysis configuration using Advanced Queuing, this option will require the setup described in the Advanced Queuing section of this handbook.

	Configuration Times	Processing Time per Student
Job Submission	1 minute	6 - 30 seconds
Advanced Queuing	Approximately 15 minutes	2 - 6 seconds

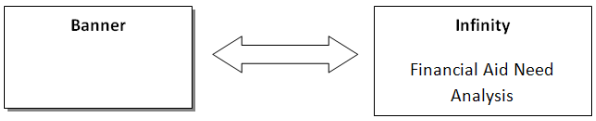
Off-Premise vs. On-Premise

Hosted solutions for Banner FM Need Analysis are separated into two major components, Banner and software-as-a-service (SaaS). The Banner element is required and must be implemented. However, two options exist for the SaaS implementation:

- Off-Premise
- On-Premise

Off-Premise

The preferred SaaS option is Off-Premise. This configuration requires no additional implementation and no maintenance. Off-Premise service offers uptime measured near 100% (based on current 2014 statistics). The service is secured by (Secure Socket Layer) SSL technology and requires a unique login for each institution. Software updates to the SaaS implementation are automatic and instantaneous (no update/upgrade actions are required by the institution).

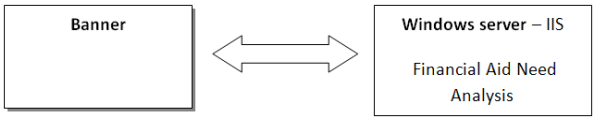


On-Premise

As an alternative, institutions may elect to use a locally hosted, On Premise, configuration. This optional service requires that the institution download, install, configure, and provide ongoing maintenance. With respect to On-Premise hosting maintenance, the institution is expected to monitor communication for a standard “update” message. In turn, the institution is expected to download, install, and configure any and all available software updates. Use of the On-Premise FM Need Analysis option also requires that either the Windows 2003 or 2008 server with IIS be installed.

Note: Two separate installation procedures are provided, one each for Windows Server 2003 and Windows Server 2008. Only one of the two Web Service options must be implemented if the On-Premise option is chosen.

Note: Support for Windows Server 2003 will end on July 14, 2015. After this date, Microsoft will no longer provide security updates, fixes or online support for Windows Server 2003.

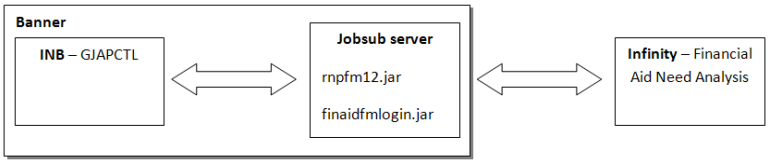


Please refer to *Chapter 3*, *FM Need Analysis On-Premise Configuration* for installation and deployment information for this option.

Configuration Checklists

This section includes a series of checklist items, based on the desired function that you want to use.

Batch Need Analysis (Required)



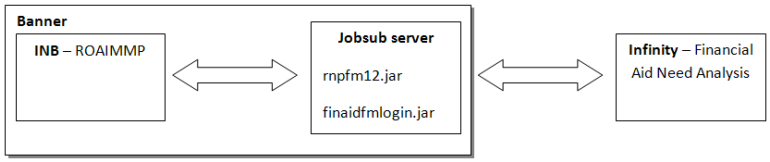
1. The `finaidfmlogin.jar` must be updated with the institution's login credentials. Refer to the *Updating finaidfmlogin.jar* procedure.
2. The job submission server must open outgoing port 443 to allow the batch process `rnpfm12.jar` to connect to server URLs:

   `https://fm1.ellucian.com`

   AND

   `https://fm2.ellucian.com`

Run immediate Need Analysis using the job submission option



1. In INB, navigate to the RORPARM form.
2. Change the value for parameter USE_JOBSUB to *Y* and save.

Run immediate Need Analysis using Advanced Queuing (refer to section on Advanced Queuing for more information)

Procedures

This section describes individual procedures associated with FM Need Analysis operations. Each procedure provides a stand-alone set of instructions associated with a particular FM Need Analysis function.

This is a reference section only and is not intended to be followed sequentially. Other sections in this chapter point back to this section to reinforce or instruct the user to perform certain necessary steps.

The following procedural information is listed in this section:

- Updating finaidfmlogin.jar
- Updating finaiddbseed.jar
- Internet Native Banner (INB)
- Loading Certificate to JVM for RNPFMyy
- Proxy Settings

Updating finaidfmlogin.jar

Embedded within the object `finaidfmlogin.jar`, is a `nas.properties` file. This file must be modified to allow each institution to connect to the Financial Aid Need Analysis system. The `finaidfmlogin.jar` file can be found:

- Under UNIX

  `$BANNER_HOME/finaid/java`

  and is linked to `$BANNER_LINKS`.

- Under Windows

  `${banner_home}\finaid\java`

Steps 1 through 4 are identical for both Windows and Unix. The additional step 5 is required for Unix, only.

1. Use the following command to extract the `nas.properties` file from `finaidfmlogin.jar`:

   ```
   jar xf finaidfmlogin.jar com/sungardhe/finaid/login/
   nas.properties
   ```

2. Open the `com/sungardhe/finaid/login/nas.properties` file in a text editor.
3. Change the user name and password values, the text to the right of the equal sign, for these properties. Be careful not to leave any trailing spaces after the values.

   ```
   httpusername=
   ```

   ```
   httppassword=
   ```

   Note: Use the authentication code provided when the institution applied for the off-premise status.

4. Use the following command to update the `finaidfmlogin.jar` with the edited `nas.properties`:

   ```
   jar uf finaidfmlogin.jar com/sungardhe/finaid/login/
   nas.properties
   ```

5. Unix only: Use the following command to relink `finaidfmlogin.jar` to `$BANNER_LINKS`:

   ```
   ls -f $BANNER_HOME/finaid/java/finaidfmlogin.jar
   $BANNER_LINKS
   ```

Updating finaiddbseed.jar

Embedded within the object `finaiddbseed.jar` is a `seed.properties` file. This file must be modified to allow each institution to connect to the Oracle database. The `finaiddbseed.jar` can be found:

- Under UNIX

  `$BANNER_HOME/finaid/java`

  and is linked to `$BANNER_LINKS`

- Under Windows

  `${banner_home}\finaid\java`

Steps 1 through 4 are identical for both Windows and Unix. The additional step 5 is required for Unix, only.

1. Use the following command to extract the `seed.properties` file from `finaiddbseed.jar`:

   ```
   jar xf finaiddbseed.jar com/sungardhe/finaid/dbseed/
   seed.properties
   ```

2. Open the `com/sungardhe/finaid/login/seed.properties` file in a text editor.

3. Using the institution security seed number, change the values, text to the right of the equal sign, for these properties. Be careful not to leave any trailing spaces after the values.

   ```
   seed1=
   ```

   ```
   seed3=
   ```

4. Use the following command to update the `finaiddbseed.jar` with the edited `seed.properties`:

   ```
   jar uf finaiddbseed.jar com/sungardhe/finaid/dbseed/
   seed.properties
   ```

5. Unix only: Use the following command to relink finaddbseed.jar to $BANNER_LINKS:

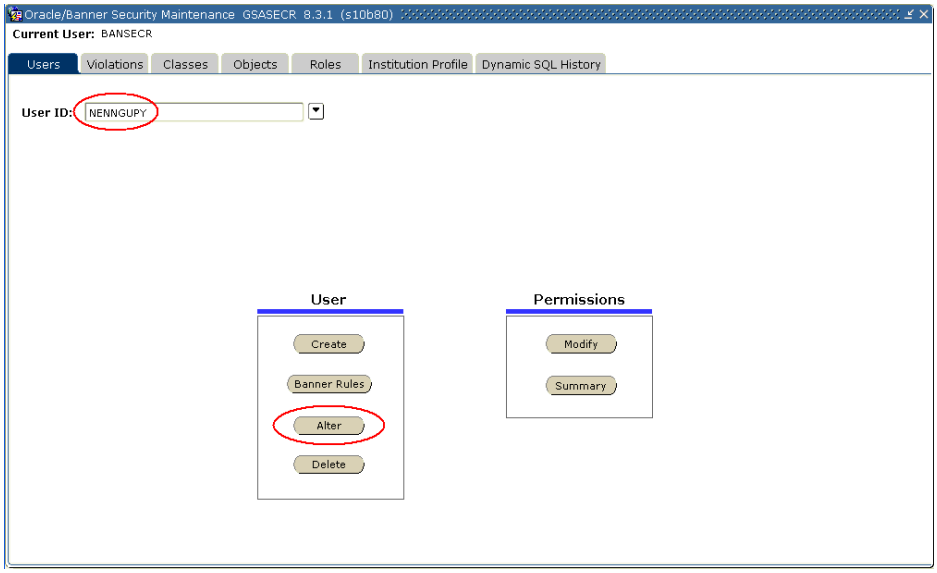
   ls `-f $BANNER_HOME/finaid/java/finaidfdbseed.jar $BANNER_LINKS`

Internet Native Banner (INB)

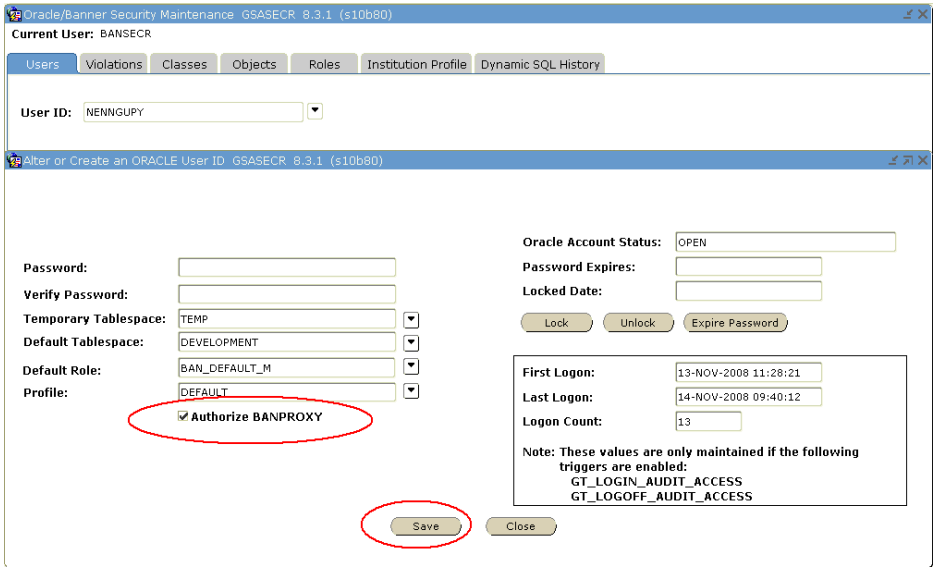
Configure banproxy

A proxy connection must be configured for each FINAID user who will be responsible for running a FM Need Analysis. 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.

1. As the bansecr userid, open the Oracle/Banner Security Maintenance (GSASECR) form.
2. Select the User ID that will use the proxy connection and click the **Alter** button.

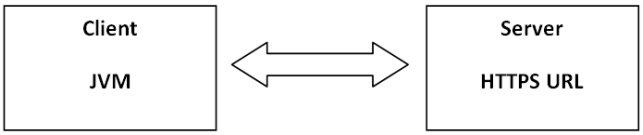


3. The User ID's credentials are displayed on the Authorize or Create an ORACLE ID window. Check the **Authorize BANPROXY** checkbox and click the **Save** button.



Loading Certificate to JVM for RNPFMyy

When connecting to a HTTPS URL (SSL is enabled on the server side) the connecting JVM may need to load the client side certificate.



Warning! *As of the Banner Financial Aid 8.13 release, the system enforces the Java Virtual Machine (JVM) 1.5 or higher requirement. With JVM 1.5 and higher, the certificates needed to connect to the server are pre-loaded.*

Use the following steps to load the client side certificate:

1. Navigate to the following directory of the job submission server:

   `<JAVA_HOME>/bin`

2. Copy the `fm_ellucian.cer` certificate to this same directory:

3. Execute the following command:

   Note: The command strings in this step must be applied in a single line format. If you choose to use these strings, 1) Highlight and copy the desired code string and paste the text into a text editor. 2) From the text editor, remove any line breaks (ultimately forcing the text onto a single line), replacing the line breaks with a single space character.

   The commands described here are intended for a Windows based environment. For UNIX and equivalent environments, replace the backslash character “\” with the forward slash character “/”.

   Note: The dash character (-), shown in the command strings below, is used as a switch within the command and should never be followed directly by a space character.

   - If the JVM is a JRE:

```
keytool -import -alias fm_ellucian -file fm_ellucian.cer -keystore
..\lib\security\cacerts -storepass changeit -noprompt
```

   - If the JVM is a JDK:

```
keytool -import -alias fm_ellucian -file fm_ellucian.cer -keystore
..\jre\lib\security\cacerts -storepass changeit -noprompt
```

Note: To load other certificates unique to the institution, use the following format:

```
keytool -import -alias <mycert> -file <mycertfile.cer>
-keystore ..\lib\security\cacerts -storepass changeit
-noprompt
```

4. Disable Loading of `Infinity.keystore`. The `Infinity.keystore` is automatically loaded by the year specific process, `rnpfm12.jar`. This must be disabled so the new certificate loaded to the JVM can be used.

   4.1. Open the appropriate script for the environment in a text editor.

   – For UNIX based system use script `rnpfm12.shl`.

   – For Windows based system use script `rnpfm12.pl`.

   4.2. Rename the property:

   – From:

   ```
   -Dcert=
   ```

   – To:

   ```
   -Dcert_HOLD=
   ```

Note: Scripts `rnpfm13.shl` and `rnpfm13.pl` and those for subsequent aid years are delivered with the *-Dcert* removed. No modifications are needed.

Proxy Settings

Warning! *Do not implement these proxy settings unless it is certain that the institution has a proxy server in place and is required for a connection to the internet. Unnecessarily implementing these steps will create connection problems where none existed.*

The process `rnpfmyy.jar` and its aid year equivalent require internet access to connect to the server. If the institution requires the connection be made through a proxy server, the following steps must be implemented.

Proxy settings for process rnpfmyy.jar

Process rnpfmyy comes with two properties, `http.proxyHost` and `http.proxyPort`, that can be added to allow the process to navigate through a proxy to connect to the internet.

1. To modify these properties, open the appropriate script for the environment in a text editor.
   - For UNIX based system use script `rnpfmyy.shl`.
   - For Windows based system use script `rnpfmyy.pl`.
2. Make the changes by adding the properties and replacing the default values with the appropriate values for your institution. Below the existing property `-Dcert`, the following properties must be added:
   - `-Dhttp.proxyHost=my_proxy_host`
   - `-Dhttp.proxyPort=my_proxy_port`

   Note: Replace the `my_proxy_host` and `my_proxy_port` with the appropriate values for your institution's proxy server.
3. For example, if the host is `www-proxy.sct.com` and the port is 8080:

For UNIX

```
. -Dhttp.proxyHost=www-proxy.sct.com \
. -Dhttp.proxyPort=8080 \
```

For Windows

```
"-Dhttp.proxyHost=www-proxy.sct.com "
"-Dhttp.proxyPort=8080 "
```

Redirecting INB to the Plus one (+1) URL

When there are changes made to the FM Need Analysis calculation, institutions will have the opportunity to direct their INB test environment's FM Calculator to this pre-production code. The pre-production URL is also called the *Plus one*, or simply *+1*. Whenever this URL is available, a separate communication will be sent to alert the Banner Financial Aid community of its status. The following configuration procedures describes the steps necessary to switch an INB test environment to use the *+1* URL.

After completing these steps, the institution will only be able to do FM Need Analysis calculations using the job submission option for online/immediate processing.

Note: The servlet option should not be used with the *+1* URL.

Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Batch processing of FM Need Analysis will continue to work as before, except it will reference the *+1* calculator.

Note: The *+1* URL is intended to be used for client testing only. It is not meant to be used for production. In addition, the purpose of the testing is to evaluate the latest changes to the functionality of the Need Analysis calculation that resides on the server. The *+1* URL is not intended to test the institutions installation of Need Analysis on Banner.

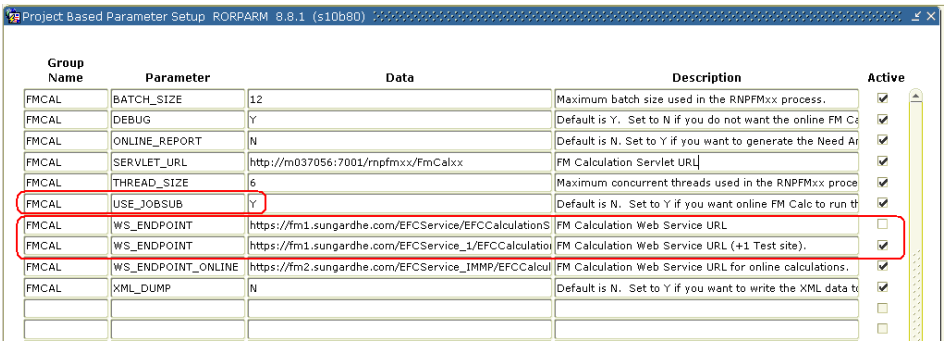
Switch the FM Need Analysis Calculator to use the pre-production (+1) URL:

1. Login into INB and navigate to the RORPARM form.
2. Make sure the value for USE_JOBSUB is set to Y.
3. For parameter WS_ENDPOINT:
   - Uncheck the **Active** indicator for URL:

     ```
     https://fm1.ellucian.com/EFCService/
     EFCCalculationService.asmx
     ```

   - Check the **Active** indicator for URL:

     ```
     https://fm1.ellucian.com/EFCService_1/
     EFCCalculationService.asmx
     ```



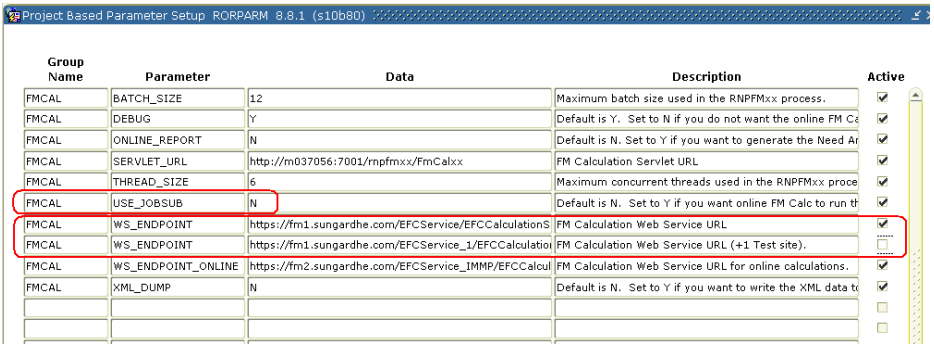
4. Take the steps necessary to save your RORPARM changes.

Switch the FM Need Analysis Calculator back to the production URL

1. Login into INB and navigate to the RORPARM form.
2. For parameter `WS_ENDPOINT`:
   - Check the Active indicator for URL:

     `https://fm1.ellucian.com/EFCService/EFCCalculationService.asmx`
   - Uncheck the Active indicator for URL:

     `https://fm1.ellucian.com/EFCService_1/EFCCalculationService.asmx`



3. Take the steps necessary to save your RORPARM changes.

Advanced Queuing (AQ) Configuration (Optional)

Introduction

To help with performance, Advanced Queuing (AQ) is available as an option, starting with the 8.16 release. AQ can replace or work in conjunction with the servlet installation. AQ does not require installation of the servlet or vice versa.

Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Note: AQ is only available starting with the Banner 8.16 release and the RNPFM13 process, and higher (RNPFMyy).

Requirements

None

Pre-Configuration

An Oracle user login that can run a Banner batch process from the host/command line is needed to start the AQ enabled processes.

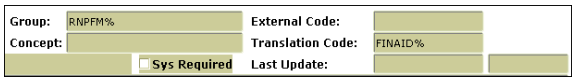
Note: For UNIX systems, because the QUEUE logs will be created in the home directory of the host user that starts RNRQINI at the host, it is recommended that the same userid that is used to start the GURJOBS process be used to start RNRQINI (for example, banjobs). For both Unix and Windows, because RNRQINI will start all Financial Aid queues (RNPFMyy as well as RPRSSBP), the userid/password specified in the command line must be a user with BAN_DEFAULT_M access to the RNPFMyy processes, RPRSSBP, and RNRQINI (for example faisusr).

Configuration

To start AQ

For FM Need Calculation

1. Login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:
   - **Group** - *RNPFM%*
   - **Translation Code** - *FINAID%*

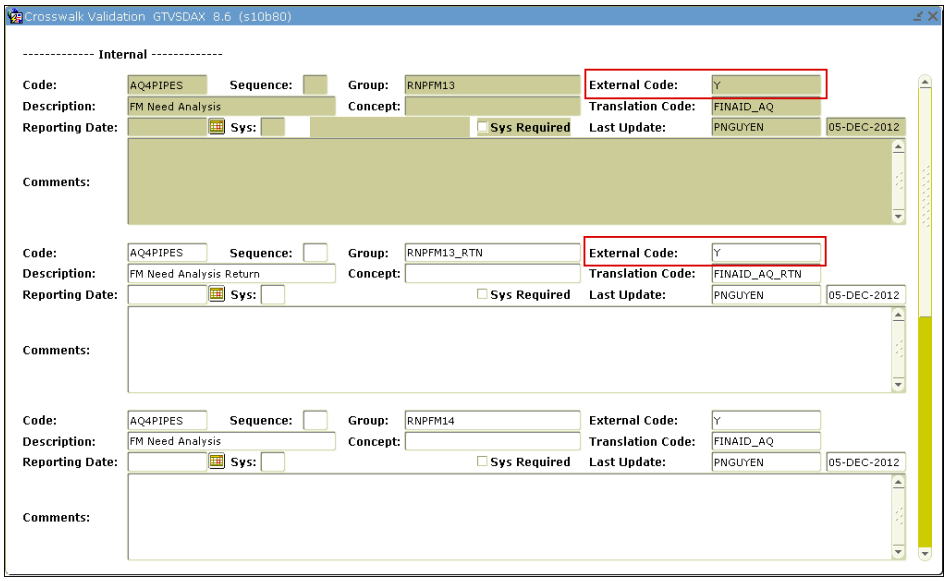


   Each process will include two records:

   – FINAID_AQ

   – FINAID_AQ_RTN

   Both records must have the **External Code** set to a value of Y.



2. In RORPARM, ensure that the USE_JOBSUB parameter is set to a value of *Y*.
3. From host, run the following command:

   ```
   rnrqini user/password START
   ```

4. If multiple Financial Aid process are setup in GTVSDAX to use AQ, a single run of RNRQINI will start any Financial Aid process using AQ.

To Stop AQ

1. From host, run the following command:

   ```
   rnrqini user/password STOP
   ```

2. To prevent the FM Need Calculation process from being restarted the next time RNRQINI is started, login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

   - **Group** - *RNPFM%*
   - T**ranslation Code** - *FINAID%*

   Each process will have two records:

   – `FINAID_AQ`

   – `FINAID_AQ_RTN`

   Both records must have the **External Code** set to a value of *N*.

3. To prevent other processes that use AQ from restarting the next time RNRQINI is started, login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

   - **Group** - *R%*
   - T**ranslation Code** - *FINAID%*

   Each process will have one record:

   – `FINAID_AQ`

   This record must have the **External Code** set to a value of *N*.

   Note: The AQ enabled processes (RNPFMyy) beginning with RNPFM13, behave very similar to GURJOBS, General's job submission process. Once you have setup GTVSDAX once, you can add the command to run RNRQINI with the start up script that starts Banner. This would be the same start-up script that starts GURJOBS.

   Likewise, to stop the processes, you can add the RNRQINI to the stop scripts that also stops GURJOBS.

Troubleshooting

Proxy not authorized

Related Error Message

java.sql.SQLException
ORA-28150: proxy not authorized to connect as client

Description

Symptom:	The user login into INB is not setup to allow proxy login.
Solution:	Review: Required installation (step 1).

Unauthorized

Related Error Message

org.apache.axis2.AxisFault: Transport error: 401 Error: Unauthorized

Description

Symptom:	The `nas.properties` file embedded within the `finaidfmlogin.jar` must be updated with the correct user name and password to allow a successful login to the server.
Solution:	Review: Required installation (step 3).

Certification path

Related Error Message

org.apache.axis2.AxisFault: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target.

OR

org.apache.axis2.AxisFault: com.ibm.jsse2.util.h: PKIX path building failed: java.security.cert.CertPathBuilderException: PKIXCertPathBuilderImpl could not build a valid CertPath.

Description

Symptom:	The JVM does not have the correct certificate loaded to connect to the server using HTTPS (SSL enabled).
Solution:	Review: Loading certificate to JVM. The `fm_ellucian.cer` certificate needs to be loaded. The `fm_ellucian.cer` certificate can be found in the `/finaid/java/` directory.

Note: If this is an on-premise solution and SSL is enabled on IIS, the institution client side certificate must be used rather than the one provided by Ellucian.

AQ Times out

As a default, the advanced queues for any Financial Aid process using AQ will timeout after four days of inactivity. A longer interval may be set using the GTVSDAX IDLEWAIT parameter. To change the IDLEWAIT setting from the default External Code value of *0* (hardcoded to mean 4 days), do the following:

1. Stop RNRQINI as follows:

   From host, run the following command:

   ```
   rnrqini user/password STOP
   ```

2. Login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

- **Code** - *IDLEWAIT*
- **Group** - *R%*

3. Change the **External Code** value from the default of *0* to be the number of seconds desired before timeout.

Note: The **External Code** value is in seconds. So, 60 sec * 60 min * 24 hours * 4 days = 345600 units.

If a *0* is used the logic in `rokadvq.f_get_idlewait` will return the default value of 345600 or 4 days. Therefore, for RNPFMXX and RPRSSBP, *0* = 4 days.

If you do not want the AQ process to timeout after 4 days of inactivity, please change this value from **External Code** = *0* to the desired number of seconds, for example 30 days: 60 sec * 60 min * 24 Hours * 30 days = 2592000 **External Code** = *2592000*

Please note it is recommended you set the IDLEWAIT value to less than 50 days (4,320,000 seconds)

4. Restart AQ as follows:

From host, run the following command:

```
rnrqini user/password START
```

FAQs

FM Need Analysis FAQs

Q1: What if the job submission server must use a proxy server to connect to the internet?
A: Refer to the *Proxy Settings* procedure (Proxy settings for process `rnpfm12.jar`).

Q2: What if the OC4J 10.1.3.x must use proxy server to connect to the internet?
A: Refer to the *Proxy Settings* procedure (Proxy settings for servlet `rnpfmxx.war`).

Note: Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Q3: Where do I get the seed number to update the seed.properties file within the `finaiddbseed.jar`?
A: If you have enough permission you can use the following SQL statement to retrieve the seed numbers:

```
SELECT GUBIPRF_SEED1,GUBIPRF_SEED3 FROM bansecr.gubiprf;
GUBIPRF_SEED1 GUBIPRF_SEED3
------------- -------------
12345678      87651234
```

Q4: There was an additional record introduced for the 7.19/8.10 Winter release in RORPARM called `WS_ENDDPOINT`. How is this additional record used?

RORPARM_GROUP_NAME	RORPARM_ACTIVE_IND	RORPARM_PARAMETER	RORPARM_DATA	RORPARM_DESCRIPTION
FMCAL	Y	ONLINE_REPORT	N	Default is N. Set to Y if you want to generate the Need Analysis Report when calculating need online.
FMCAL	N	WS_ENDPOINT	https://fm1.sungardhe.com/EFCService_1/EFCCalculationService.asmx	FM Calculation Web Service URL (+1 Test site).

A: This record with "/EFCService_1" in the URL can be made active instead of the production URL record for pre-production testing.

Q5: Do I need to open port 443 on the Oracle database server to allow connection to ?
A: No. For security reasons, the Need Analysis has been intentionally coded not to make a web service call directly from the Oracle database server and the INB server. However, the job submission server will need port 443 open to allow the batch process to connect to server. In addition, if the servlet option is chosen for immediate processing of Need Analysis, port 443 will also need to be open for the environment hosting the OC4J 10.1.3.x.

Note: Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Q6: If the Off-Premise option has been chosen, do we still need to change anything in RORPARM?
A: No. The default configuration is for Off-Premise.

Q7: For RORPARM settings, why does Ellucian have three sites?
A: For best performance when doing a single calculation from immediate processing or ONLINE processing, Advanced Queuing is the best option.

To help with performance, the online processing is separated from the batch processing so that the online processing is not competing for the same resources as the batch processing of thousands of student records.

- Batch processing (production version of the calculator):

```
https://fm1.ellucian.com/EFCService/
EFCCalculationService.asmx
```

- ONLINE processing (production version of the calculator):

```
https://fm2.ellucian.com/EFCService_IMMP/
EFCCalculationService.asmx
```

- Batch processing (BETA/pre-production version of the calculator):

```
https://fm1.ellucian.com/EFCService_1/
EFCCalculationService.asmx
```

Q8: When setting up `finaidfmlogin.jar`, where can I find my institution's authentication code, the values needed for in the `nas.properties` file?

```
httpusername=

httppassword=
```

A. The authentication code was provided to your institution when your institution applied for off-premise status. If the authentication code cannot be found, contact the Ellucian Action Line for assistance.

FM Need Analysis On-Premise Configuration

Introduction

This chapter supplies technical details associated with installation and configuration of the hosted FM Need Analysis solution. Configurations, both required and optional, are included to achieve successful 2011-2012 FM Need Analysis calculations associated with release 8.10, or above. As such, this chapter is written with references specific to the RNPFM12 object. However, the steps associated with RNPFM12 can also be applied to subsequent aid year-specific versions of the RNPFMyy object

To support the immediate FM Need Analysis calculation, the following two options have been made available:

- Job submission
- Advanced Queuing (AQ) - Available for year-specific RNPFMyy processes (beginning with RNPFM13).

AQ has been made available as an optional installation that can improve performance of online calculations. AQ runs on the job submission server, the same environment that runs batch processing. Refer to the *Advanced Queuing* section for additional details.

Note: Information found in this chapter supersedes all configuration requirements noted in prior release documentation.

Note: Issues regarding the hosted Banner Financial Aid FM Need Analysis solution should be submitted to the Ellucian ActionLine under the Product Line: *Financial Aid* and the Product: *FM Need Analysis*.

When submitting a Service Request from the Ellucian Support Center, select the correct Product Line (`Financial Aid`) and Product (`FM Need Analysis`) options from the available drop down lists.

Object List

The following objects are associated with FM Need Analysis:

- `finaiddbseed.jar`

- `finaidfmlogin.jar`
- `finaidutils.jar`
- `fm_ellucian.cer`
- `rnpfmyy.jar *`
- `rnpfmyy.pl *`
- `rnpfmyy.shl *`
- `rnpfmxx_plan.dat`

***** - Where "yy" is the aid year.

Note: In INB, access to the RORPARM form will also be needed to update references to the servlet.

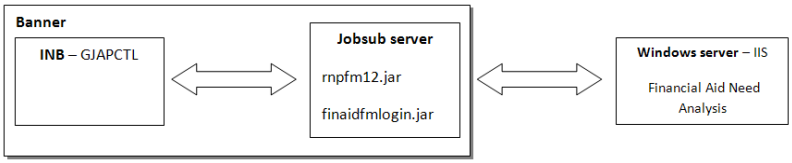
Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Configuration Options

Several options are available for running FM Need Analysis. The following diagrams illustrate the various FM Need Analysis configuration options:

Batch Need Analysis

The most efficient way to process large numbers of students for FM Need Analysis is by using job submission to run the RNPFMyy process. Therefore, this is a required configuration:

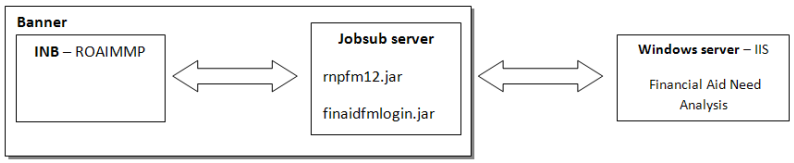


Immediate Processing (Job Submission Process vs. Advanced Queuing)

There are two configuration options available to perform immediate student processing for FM Need Analysis.

Note: The Servlet option is no longer supported (beginning with the 2014-2015 aid year - RNPFM15) and was replaced by Oracle Advanced Queuing (AQ). Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Because the RORPARM form is not aid year-specific, Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Running immediate FM Need Analysis using the traditional job submission:



Running immediate FM Need Analysis using Advanced Queuing. Refer to section on Advanced Queuing for more information.

While better performance is noted using the immediate FM Need Analysis configuration using Advanced Queuing, this option will require the setup described in the Advanced Queuing section of this handbook.

	Configuration Times	Processing Time per Student
Job Submission	1 minute	6 - 30 seconds
Advanced Queuing	Approximately 15 minutes	2 - 6 seconds

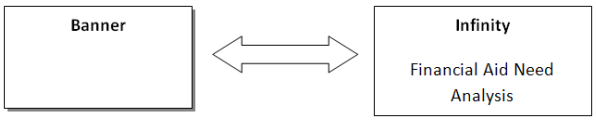
Off-Premise vs. On-Premise

Hosted solutions for Banner FM Need Analysis are separated into two major components, Banner and software-as-a-service (SaaS). The Banner element is required and must be implemented. However, two options exist for the SaaS implementation:

- Off-Premise
- On-Premise

Off-Premise

The preferred SaaS option is Off-Premise. This configuration requires no additional implementation and no maintenance. Off-Premise service offers uptime measured near 100% (based on current 2014 statistics). The service is secured by (Secure Socket Layer) SSL technology and requires a unique login for each institution. Software updates to the SaaS implementation are automatic and instantaneous (no update/upgrade actions are required by the institution).



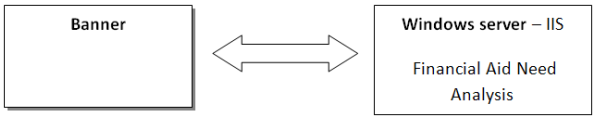
Please refer to *Chapter 2*, *FM Need Analysis Off-Premise Configuration* for installation and deployment information for this option.

On-Premise

As an alternative, institutions may elect to use a locally hosted, On Premise, configuration. This optional service requires that the institution download, install, configure, and provide ongoing maintenance. With respect to On-Premise hosting maintenance, the institution is expected to monitor communication for a standard “update” message. In turn, the institution is expected to download, install, and configure any and all available software updates. Use of the On-Premise FM Need Analysis option also requires that either the Windows 2003 or 2008 server with IIS be installed.

Note: Two separate installation procedures are provided, one each for Windows 2003 and Windows 2008. Only one of the two Web Service options must be implemented if the On-Premise option is chosen.

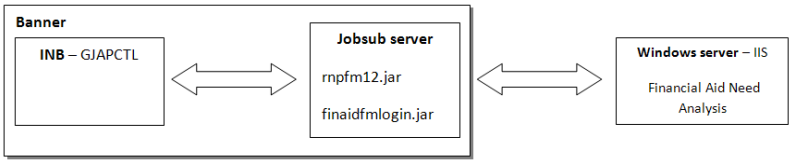
Note: Support for Windows Server 2003 will end on July 14, 2015. After this date, Microsoft will no longer provide security updates, fixes or online support for Windows Server 2003.



Configuration Checklists

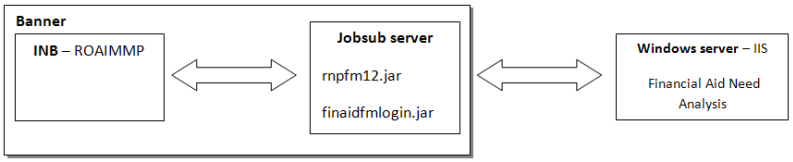
This section includes a series of checklist items, based on the desired function that you want to use.

Batch Need Analysis (Required)



- The `finaidfmlogin.jar` may be updated with the institution's login credentials. Refer to the *Updating finaidfmlogin.jar* procedure.

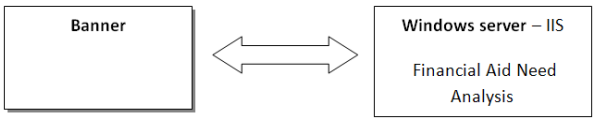
Run immediate Need Analysis using the job submission option



1. In INB, navigate to the RORPARM form.
2. Change the value for parameter USE_JOBSUB to *Y* and save.

Run immediate Need Analysis using Advanced Queuing (refer to section on Advanced Queuing for more information)

On-Premise Installation



Because the institution has selected the on-premise status, a locally hosted version of the Financial Aid Need Analysis must be installed.

1. Check for the latest version of `FinAidNeedAnalysis.xxxxxx.x.zip` and download the file if needed.

2. Install the web service on to an IIS server. There are two installation guides provided:
   - Windows 2003 running IIS 6
   - Windows 2008 running IIS 7
3. Choose the appropriate guide for your institution. Refer to the *On-Premise* procedure.

Procedures

This section describes individual procedures associated with FM Need Analysis operations. Each procedure provides a stand-alone set of instructions associated with a particular FM Need Analysis function.

This is a reference section only and is not intended to be followed sequentially. Other sections in this chapter point back to this section to reinforce or instruct the user to perform certain necessary steps.

The following procedural information is listed in this section:

- Updating finaidfmlogin.jar
- Updating finaiddbseed.jar
- Internet Native Banner (INB)
- On-Premise
- Proxy Settings

Updating finaidfmlogin.jar

If Basic Authentication was used during the Windows IIS installation, the following procedure must be done. Otherwise, no changes are needed.

Embedded within the object `finaidfmlogin.jar`, is a nas.properties file. This file can be modified to allow connection to Windows IIS with the credential created when Basic Authentication was turned on. The `finaidfmlogin.jar` file can be found:

- Under UNIX

  ```
  $BANNER_HOME/finaid/java
  ```

- Under Windows

  ```
  ${banner_home}\finaid\java
  ```

Steps 1 through 4 are identical for both Windows and Unix. The additional step 5 is required for Unix, only.

1. Use the following command to extract the `nas.properties` file from `finaidfmlogin.jar`:

   ```
   jar xf finaidfmlogin.jar com/sungardhe/finaid/login/
   nas.properties
   ```

2. Open the `com/sungardhe/finaid/login/nas.properties` file in a text editor.

3. Change the user name and password values, the text to the right of the equal sign, for these properties. Be careful not to leave any trailing spaces after the values.

   ```
   httpusername=
   ```

   ```
   httppassword=
   ```

   Note: Use the credentials created during the Windows IIS installation.

4. Use the following command to update the `finaidfmlogin.jar` with the edited `nas.properties`:

   ```
   jar uf finaidfmlogin.jar com/sungardhe/finaid/login/
   nas.properties
   ```

5. Unix only: Use the following command to relink `finaidfmlogin.jar` to `$BANNER_LINKS`:

   ```
   ln -f $BANNER_HOME/finaid/java/finaidfmlogin.jar
   $BANNER_LINKS
   ```

Updating finaiddbseed.jar

Embedded within the object `finaiddbseed.jar` is a `seed.properties` file. This file must be modified to allow each institution to connect to the Oracle database. The `finaiddbseed.jar` can be found:

- Under UNIX

  ```
  $BANNER_HOME/finaid/java
  ```

- Under Windows

  ```
  ${banner_home}\finaid\java
  ```

Steps 1 through 4 are identical for both Windows and Unix. The additional step 5 is required for Unix, only.

1. Use the following command to extract the `seed.properties` file from `finaiddbseed.jar`:

   ```
   jar xf finaiddbseed.jar com/sungardhe/finaid/dbseed/
   seed.properties
   ```

2. Open the `com/sungardhe/finaid/login/seed.properties` file in a text editor.
3. Using the institution security seed number, change the values, text to the right of the equal sign, for these properties. Be careful not to leave any trailing spaces after the values.

   ```
   seed1=

   seed3=
   ```

4. Use the following command to update the `finaiddbseed.jar` with the edited `seed.properties`:

   ```
   jar uf finaiddbseed.jar com/sungardhe/finaid/dbseed/
   seed.properties
   ```

5. Unix only: Use the following command to relink `finaddbseed.jar` to `$BANNER_LINKS`:

   ```
   ls -f $BANNER_HOME/finaid/java/finaidfdbseed.jar
   $BANNER_LINKS
   ```

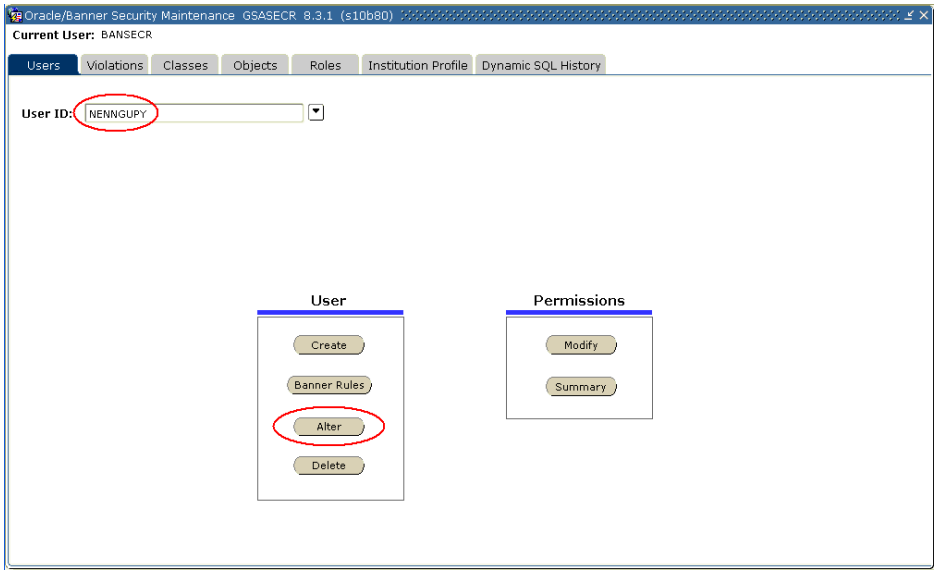
Internet Native Banner (INB)

Configure banproxy

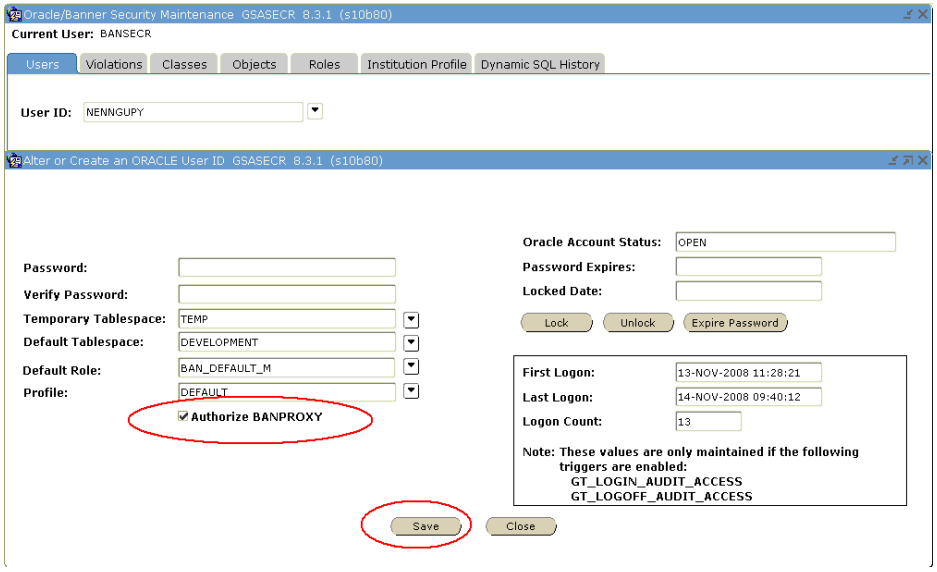
A proxy connection must be configured for each FINAID user who will be responsible for running a FM Need Analysis. The INB banproxy user will be configured to act as the proxy user. Appropriate permission is required for each user to allow banproxy to connect to Oracle on that login user's behalf.

1. As the bansecr userid, open the Oracle/Banner Security Maintenance (GSASECR) form.

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



3. 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.



On-Premise

IIS 6.0 on Windows Server 2003

This installation is only required if the institution applied for an on-premise install status.

Note: Support for Windows Server 2003 will end on July 14, 2015. After this date, Microsoft will no longer provide security updates, fixes or online support for Windows Server 2003.

Minimum requirements

1. The on-premise install of the Financial Aid Need Analysis is called the EFCService web service and it can only be installed on a Windows Server:
   - Windows Server 2003 running IIS 6
   - Microsoft.NET Framework 3.5
   - IIS configured to run ASP.net 2.0
2. Installing IIS 6.0 on Windows Server 2003
   2.1. A detailed set of 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`

   2.2. Click Start->Programs->Administrative Tools->Manage Your Server.

   2.3. Click Add or remove a role.

   2.4. Use the Wizard to add the Application server (IIS, ASP.NET).
3. Installing EFCService web service:

   Note: If you already have a working EFCService from a prior install, you must stop the service before proceeding.

   3.1. Unzip latest version of the `FinAidNeedAnalysis.xxxxxx.x.zip` into the following folder:

   `C:\inetpub\wwwroot\EFCService`.

   The structures should include:

```
..\EFCService\App_Data\Config.xml
              \bin\EFC.dll
                  \EFCSoapExtension.dll
                  \ExceptionHandling.dll
                  \FederalEFC1011.dll
```

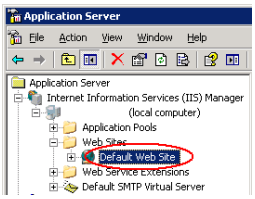
```
\FederalEFC.dll
\FinancialAidService.dll
\efc_calculation_messages.xsd
\EFCCalculationService.asmx
\Web.config
```

Note: If you already have a working EFCService from a prior install, you can now restart the service and skip to *step 4*, *Test the Web Service*.

3.2. Start the Manage Your Server.

3.3. Click Manage this application server.

3.4. From the Application Server, navigate to the Default Web Site.



3.5. Set the ASP.NET version.

– Right click on the Default Web Site and select Properties.

– On the ASP.NET tab, set the ASP.NET version to 2.0.x. Click Ok.

3.6. Create a Virtual Directory

– Right click on the Default Web Site and select New->Virtual Directory.

– Use the Virtual Directory Creation Wizard:

Alias use EFCService.

Path use the path from step 3.3 above.

On the Virtual Directory Access Permissions check both

– Read

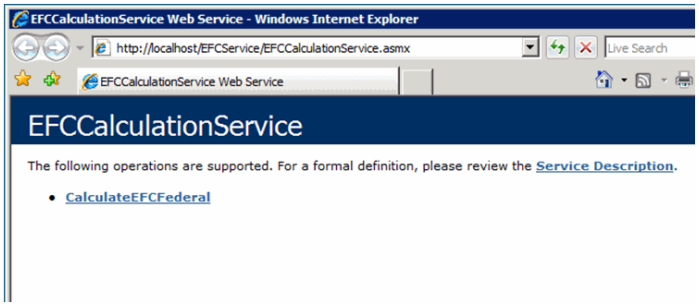
– Run scripts (such as ASP)

4. Test the Web Service:

   4.1. Enter and go to the following URL in a web browser:

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

   4.2. If the EFCCalculationService page is displayed, the Web Service test was successful.



5. Configuring the EFCService to use Basic Authentication (optional):

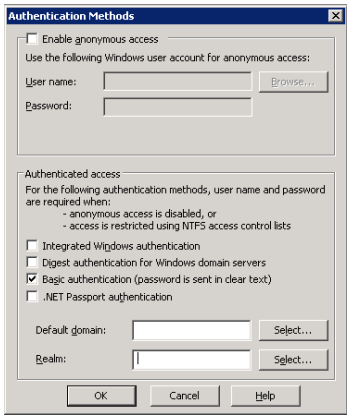
   5.1. Navigate back to the Default Web Site on the Application Server (refer to steps 3.4 – 3.5, earlier in this procedure).

   5.2. Right click on the Default Web Site and select Properties.

   5.3. On the Directory Security tab, edit the Authentication and access control.

   5.4. Uncheck Enable anonymous access.

   5.5. Check Basic authentication and enter the Default domain and Realm.



   5.6. Update the user name and password for Banner to make a connection. Refer to the *Updating finaidfmlogin.jar* procedure.

6. Setup EFCService for SSL (optional).

   6.1. SSL requires obtaining and registering a certificate with the Web Server.

   6.2. The bit length must be 1024 or 2048.

   6.3. A detailed set of 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`

   6.4. The client side certificate must be loaded to both JVM. Refer to the *Loading Certificate to JVM* procedure.

   – JVM running JAVA processes on the job submission server.

7. Update INB to recognize the on-premise FM Calculation Web Service.

   7.1. Login into INB and navigate to the RORPARM form.

   7.2. For both parameters, `WS_ENDPOINT` and `WS_ENDPOINT_ONLINE`, replace the default value with URL from step 4 – Test the Web Service.

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

IIS 7.0 on Windows Server 2008

This installation is only required if the institution applied for an on-premise install status.

Minimum requirements

1. The on-premise install of the Financial Aid Need Analysis is called the EFCService web service and it can only be installed on a Windows Server:

   - Windows 2008 running IIS 7
   - Microsoft.NET Framework 3.5
   - IIS configured to run ASP.net 2.0.

2. Installing IIS 7.0 on Windows Server 2008

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

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

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

   2.3. In Roles Summary, click Add Roles.

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

   2.5. During the Select Role Services, check to include the following items:

   – `ASP.NET`

   – Basic Authentication

3. Installing EFCService Web Service:

Note: If you already have a working EFCService from a prior install, you must stop the service before proceeding.

3.1. Unzip latest version of the `FinAidNeedAnalysis.xxxxxx.x.zip` into the following folder:

`C:\inetpub\wwwroot\EFCService`.

The structures should include:

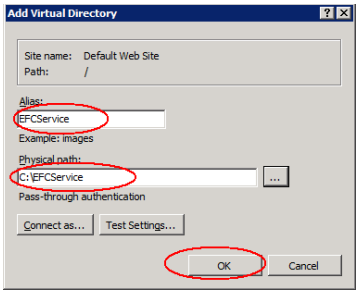
```
..\EFCService\App_Data\Config.xml
              \bin\EFC.dll
                  \EFCSoapExtension.dll
                  \ExceptionHandling.dll
                  \FederalEFC1011.dll
                  \FederalEFC.dll
                  \FinancialAidService.dll
              \efc_calculation_messages.xsd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

Note: If you already have a working EFCService from a prior install, you can now restart the service and skip to *step 4*, *Test the Web Service*.

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

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

– Fill out the Add Virtual Directory popup. Make sure to use EFCService for the Alias and click OK.



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

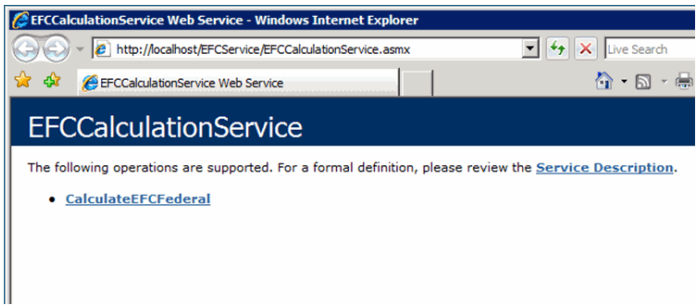
- From the Connections screen, select Default Web Site and do a Refresh by pressing the F5 function key.
- The EFCService Folder is displayed:
- Right click the EFCService folder, select Convert to Application from the popup menu and select OK.

4. Test the Web Service:

4.1. Enter and go to the following URL in a web browser:

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

4.2. If the EFCCalculationService page is displayed, the web service test was successful.



5. Configuring the EFCService to use Basic Authentication (optional):

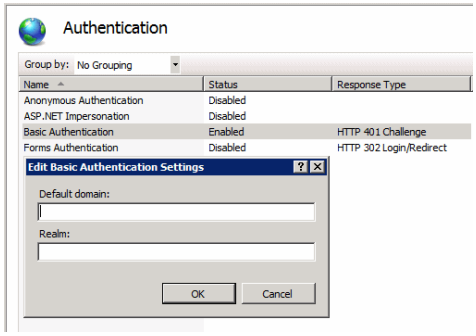
5.1. Navigate back to the Default Web Site on the Internet Information Service (IIS) Manager.

5.2. With the Features View on, select Authentication.

5.3. Disable Anonymous Authentication.

5.4. Enable Basic Authentication.

5.5. Edit Basic authentication and enter the Default domain and Realm.



5.6. The user name and password must be updated for Banner to connect. Refer to the *Updating finaidfmlogin.jar* procedure.

6. Setup EFCService for SSL (optional).

   6.1. SSL requires obtaining and registering a certificate with the Web Server.

   6.2. The bit length must be 1024 or 2048.

   6.3. A detailed set of 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
   ```

   6.4. The client side certificate must be loaded to both JVM. Refer to the *Loading Certificate to JVM* procedure.

   – JVM running JAVA processes on the job submission server.

7. Update INB to recognize the on-premise FM Calculation Web Service.

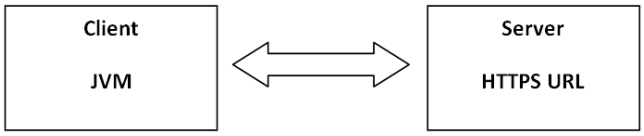
   7.1. Login into INB and navigate to the RORPARM form.

   7.2. For both parameters, `WS_ENDPOINT` and `WS_ENDPOINT_ONLINE`, replace the default value with URL from step 4 – Test the Web Service.

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

Loading Certificate to JVM for RNPFMyy

When connecting to a HTTPS URL (SSL is enabled on the server side) the connecting JVM may need to load the client side certificate.



Use the following steps to load the client side certificate:

1. Navigate to the following directory of the job submission server:

   `<JAVA_HOME>/bin`

2. Copy your client side SSL certificates to this same directory.

   Note: Do not use the `fm_ellucian.cer` Certificate.

3. Execute the following command:

Note: The command strings in this step must be applied in a single line format. If you choose to use these strings, 1) Highlight and copy the desired code string and paste the text into a text editor. 2) From the text editor, remove any line breaks (ultimately forcing the text onto a single line), replacing the line breaks with a single space character.

The commands described here are intended for a Windows based environment. For UNIX and equivalent environments, replace the backslash character "\" with the forward slash character "/". ?

Note: The dash character (-), shown in the command strings below, is used as a switch within the command and should never be followed directly by a space character. ?

- If the JVM is a JRE:

```
keytool -import -alias <mycert> -file <mycertfile.cer> -keystore
..\lib\security\cacerts -storepass changeit -noprompt
```

- If the JVM is a JDK:

```
keytool -import -alias <mycert> -file <mycertfile.cer> -keystore
..\lib\security\cacerts -storepass changeit -noprompt
```

4. Disable Loading of Infinity.keystore. The Infinity.keystore is automatically loaded by the year specific process, rnpfm12.jar. This must be disabled so the new certificate loaded to the JVM can be used.

4.1. Open the appropriate script for the environment in a text editor.

– For UNIX based system use script rnpfm12.shl.

– For Windows based system use script rnpfm12.pl.

4.2. Rename the property:

– From:

```
-Dcert=
```

– To:

```
-Dcert_HOLD=
```

Note: Scripts `rnpfm13.shl` and `rnpfm13.pl` and those for subsequent aid years are delivered with the *-Dcert* removed. No modifications are needed.

Proxy Settings

Warning! *Do not implement these proxy settings unless it is certain that the institution has a proxy server in place and is required for a connection to the internet. Unnecessarily implementing these steps will create connection problems where none existed.*

The process `rnpfmyy.jar` and its aid year equivalent require internet access to connect to the server. If the institution requires the connection be made through a proxy server, the following steps must be implemented.

Proxy settings for process rnpfmyy.jar

Process rnpfmyy comes with two properties, `http.proxyHost` and `http.proxyPort`, that can be added to allow the process to navigate through a proxy to connect to the internet.

1. To modify these properties, open the appropriate script for the environment in a text editor.
   - For UNIX based system use script `rnpfmyy.shl`.
   - For Windows based system use script `rnpfmyy.pl`.
2. Make the changes by adding the properties and replacing the default values with the appropriate values for your institution. Below the existing property `-Dcert`, the following properties must be added:
   - `-Dhttp.proxyHost=my_proxy_host`
   - `-Dhttp.proxyPort=my_proxy_port`

Note: Replace the `my_proxy_host` and `my_proxy_port` with the appropriate values for your institution's proxy server.

3. For example, if the host is `www-proxy.sct.com` and the port is 8080:

For UNIX

```
. -Dhttp.proxyHost=www-proxy.sct.com \
. -Dhttp.proxyPort=8080 \
```

For Windows

```
"-Dhttp.proxyHost=www-proxy.sct.com "
"-Dhttp.proxyPort=8080 "
```

Advanced Queuing (AQ) Configuration (Optional)

Introduction

To help with performance, Advanced Queuing (AQ) is available as an option, starting with the 8.16 release.

Note: AQ is only available starting with the Banner Financial Aid 8.16 release and the RNPFM13 process, and higher.

Requirements

None

Pre-Configuration

An Oracle user login that can run a Banner batch process from the host/command line is needed to start the AQ enabled processes.

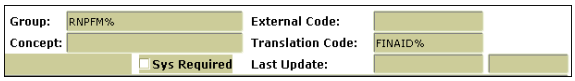
Note: For UNIX systems, because the QUEUE logs will be created in the home directory of the host user that starts RNRQINI at the host, it is recommended that the same userid that is used to start the GURJOBS process be used to start RNRQINI (for example, banjobs). For both Unix and Windows, because RNRQINI will start all Financial Aid queues (RNPFMyy as well as RPRSSBP), the userid/password specified in the command line must be a user with BAN_DEFAULT_M access to the RNPFMyy processes, RPRSSBP, and RNRQINI (for example faisusr).

Configuration

To start AQ

For FM Need Calculation

1. Login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:
   - **Group** - *RNPFM%*
   - **Translation Code** - *FINAID%*

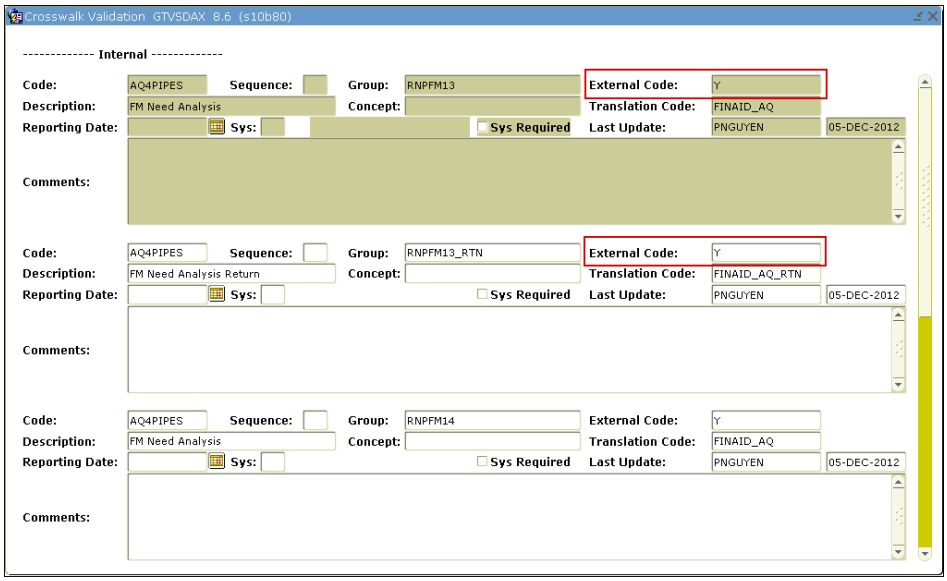


   Each process will include two records:

   – FINAID_AQ

   – FINAID_AQ_RTN

   Both records must have the **External Code** set to a value of Y.



2. In RORPARM, ensure that the USE_JOBSUB parameter is set to a value of *Y*.
3. From host, run the following command:

   ```
   rnrqini user/password START
   ```

4. If multiple Financial Aid process are setup in GTVSDAX to use AQ, a single run of RNRQINI will start any Financial Aid process using AQ.

To Stop AQ

1. From host, run the following command:

   ```
   rnrqini user/password STOP
   ```

2. To prevent the FM Need Calculation process from being restarted the next time RNRQINI is started, login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

   - **Group** - *RNPFM%*
   - T**ranslation Code** - *FINAID%*

   Each process will have two records:

   – `FINAID_AQ`

   – `FINAID_AQ_RTN`

   Both records must have the **External Code** set to a value of *N*.

3. To prevent other processes that use AQ from restarting the next time RNRQINI is started, login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

   - **Group** - *R%*
   - T**ranslation Code** - *FINAID%*

   The process will have one record:

   – `FINAID_AQ`

   This record must have the **External Code** set to a value of *N*.

   Note: The AQ enabled processes (RNPFMyy) beginning with RNPFM13, behave very similar to GURJOBS, General's job submission process. Once you have setup GTVSDAX once, you can add the command to run RNRQINI with the start up script that starts Banner. This would be the same start-up script that starts GURJOBS.

   Likewise, to stop the processes, you can add the RNRQINI to the stop scripts that also stops GURJOBS.

Windows Server Platform Certification (On-Premise Only)

Windows Server 2008 Platform

Introduction

Microsoft releases new operating systems every three to five years. This section contains FM Need Analysis Service certification details for Windows Server 2008 operating system platforms as follows:

- Windows 2008 (32 & 64 bit)
- Windows 2008 R2 (64 bit)

Windows Server 2008 (32 & 64 bit)

In the Windows Server 2008 operating system, Microsoft introduced many new features and technologies, which were not available in Windows Server 2003 with Service Pack 2 (SP2), that help to increase the security of computers running Windows Server 2008, increase productivity, and reduce administrative overhead. This OS is available on both the 32 and 64 bit platform.

Windows Server 2008 R2 (64 bit)

The Windows Server 2008 R2 operating system has been built on the foundation of Windows Server 2008, expanding existing technology and adding new features to enable organizations to increase the reliability and flexibility of their server infrastructures. This OS is only available on the 64 bit platform.

Microsoft has introduced powerful tools such as IIS version 7.5, updated Server Manager and Windows Power Shell version 2.0 to give customers greater control, increased efficiency, and the ability to react to front-line business needs faster than ever before. The features of Windows Server 2008 R2 are:

- Improved Web Application Platform
- Server and Desktop Virtualization
- Improved Power Management
- Scalability
- Reliability

Certification Objective

The objective is to certify the Banner Financial Aid FM Need Analysis service for compatibility against the following Windows operating system platforms:

- Windows 2008 (32 & 64 bit)
- Windows 2008 R2 (64 bit native)

Configuration Points Tested

Windows 2008 (32 bit)

The certification exercise determined the effect of installing Banner Financial Aid FM Need Analysis Service on the application server running Microsoft Windows Server 2008 (32-bit).

The following table lists the various Windows 2008 (32 bit) server side features used for validating the service:

Configuration Item	Option	Comments
Server Roles and Role services	Web role	Web role with Web server and Management tools as the role services is the minimal requirement.
Role services for Web role	Role services installed: • Web server • Management tools	
Supported versions of .NET	.NET 3.5 .NET 4.0	.NET 3.5 is the minimal requirement with the latest service pack SP1 applied.
Supported versions of IIS	IIS 7.0	IIS 7.0 version is installed with Web server role.
IIS Configuration • Application Authentication	• Anonymous • Basic	Virtual directory to be configured as an application. Retain the default setting of Anonymous authentication.
IIS Application Pool .NET Framework Version This is separate from the installed version of the dotNET Framework and refers to runtime version of .NET)	Application Pool: Managed pipeline mode: Integrated • If .NET Framework 3.5 is installed then .NET Framework version is v2.0.50727 • If .NET Framework 4.0 is installed then .NET Framework version of v4.0.30319 needs to be selected.	.NET Framework version 2.050727 and mode as Integrated pipeline is the recommended setting.

Windows 2008 (64 bit)

The certification exercise determined the effect of installing FM Need Analysis Service on the application server running Microsoft Windows Server 2008 (64-bit).

The following table lists the various Windows 2008 (64 bit) server side features used for validating the service:

Configuration Item	Option	Comments
Server Roles and Role services	Web role	Web role with Web server and Management tools as the role services is the minimal requirement.
Role services for Web role	Role services installed: • Web server • Management tools	
Supported versions of .NET	.NET 3.5 .NET 4.0	.NET 3.5 is the minimal requirement with the latest service pack SP1 applied.
Supported versions of IIS	IIS 7.0	IIS 7.0 version is installed with Web server role.
IIS Configuration • Application Authentication	• Anonymous • Basic	Virtual directory to be configured as an application. Retain the default setting of Anonymous authentication.
IIS Application Pool .NET Framework Version This is separate from the installed version of the dotNET Framework and refers to runtime version of .NET)	Application Pool: Managed pipeline mode: Integrated • If .NET Framework 3.5 is installed then .NET Framework version is v2.0.50727 • If .NET Framework 4.0 is installed then .NET Framework version of v4.0.30319 needs to be selected.	.NET Framework version 2.0 and mode as Integrated pipeline is the recommended setting.

Windows 2008 R2 (64 bit)

The certification exercise determined the effect of installing FM Need Analysis Service on the application server running Microsoft Windows Server 2008 R2 (64-bit).

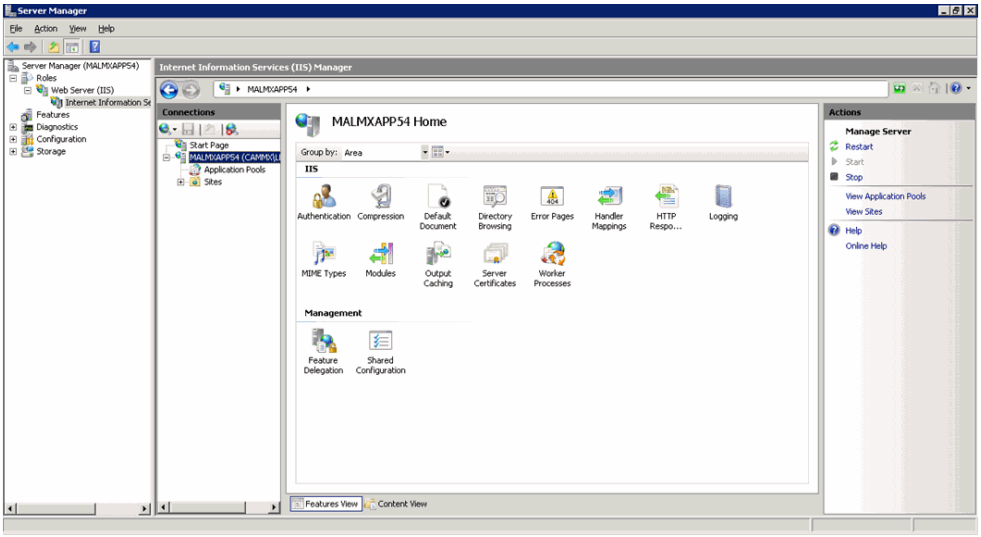
The following table lists the various Windows 2008 R2 (64 bit) server side features used for validating the service:

Configuration Item	Option	Comments
Server Roles and Role services	Web role	Web role with Web server and Management tools as the role services is the minimal requirement.
Role services for Web role	Role services installed: • Web server • Management tools	
Server Features	.NET Framework 3.5.1 features Remote server administration tools Windows process activation service	These features will need to be installed manually.
Supported versions of .NET	.NET 3.5 .NET 4.0	.NET 3.5 is the minimal requirement with the latest service pack SP1 applied. This is installed as a part of Server Features.
Supported versions of IIS	IIS 7.5	IIS 7.5 version is installed with Web server role.
IIS Configuration • Application Authentication	• Anonymous • Basic	Virtual directory to be configured as an application. Retain the default setting of Anonymous authentication.
IIS Application Pool .NET Framework Version This is separate from the installed version of the dotNET Framework and refers to runtime version of .NET)	Application Pool: Managed pipeline mode: Integrated • If .NET Framework 3.5 is installed then .NET Framework version is v2.0.50727 • If .NET Framework 4.0 is installed then .NET Framework version of v4.0.30319 needs to be selected.	.NET Framework version 2.0 and mode as Integrated pipeline is the recommended setting.

Helpful Screen Shots

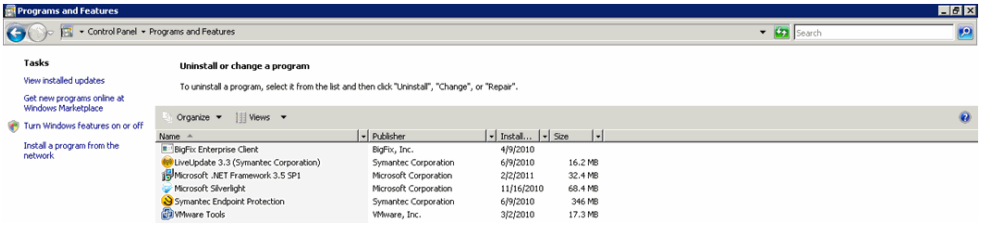
Windows Web Server Role & IIS Management Console

This is the IIS Manager console from which the IIS settings for various hosted web applications will be configured.



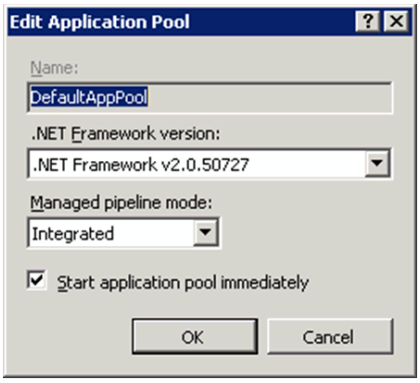
Programs and Features

This was formerly Add Remove Programs and will show which version(s) of dotNET are installed on the server. Programs and Features option is found in the Control Panel.



Application Pool Settings

This is the configuration settings for the application pool at the IIS level namely the .Net framework version and the pipeline mode which will be applied to the associated applications.



Certification Method

Certifying Banner Financial Aid FM Need Analysis Service involved the following tasks:

1. Identify the pre-requisites required for installing the Banner Financial Aid FM Need Analysis Service on Windows 2008 server. These features and patterns form the source for the certification testing of the service.
   - Identify the server-level configurations, different server roles, and features to be installed (i.e Application and Web server roles available).
   - Identify the supported .Net framework versions to be installed on the server for the service to be running.
   - Identify the configurations required at the IIS level for hosting the web service which includes the Application pool settings with respect to the .Net framework version and mode & the virtual directory settings.
2. The above step would also identify and analyze the behavioral changes in Windows 2008 which could affect the behavior of the Service.
3. Map the features and patterns identified to form combinations of server-level settings to be evaluated for validation of the service.
4. Installing Banner Financial Aid FM Need Analysis Service on the application server running Windows 2008.
5. Validating the invocation of the EFC Web service mapped to the chosen patterns and reporting issues.
6. Debugging and documenting the issues resolved.

Issues Identified

Issue I (All Platforms)

.Net Framework 4.0 related issues were identified as part of the certification exercise. Detailed description of each of the issues along with any known resolution is provided below:

Scenario	When .NET Framework 4.0 is installed and the application pool is configured to run in the .NET Framework version 4.0.
Problem	There is a duplicate `system.web.extensions/scripting/scriptResourceHandler` section defined.
Resolution	Option 1: • To set the .NET Framework version to v2.0.50727 for `DefaultAppPool` in the application pool at the IIS level. Option 2: • With the .NET framework version set to 4.0 in the application pool, the `system.web.extensions` section needs to be commented out in the `web.config` file.
Comments	Refer to the following links for additional information: • http://www.sitefinity.com/devnet/forums/sitefinity-3-x/suggestions/config-error-there-is-a-duplicate-system-web-extensions-scripting-scriptresourcehandler-section-defined.aspx • http://forums.asp.net/p/1571308/3939106.aspx

Issues II & III (All Platforms)

The following are two known issues that can occur if ASP.NET framework has been installed incorrectly:

Scenario	This error can occur when IIS is installed after installing .NET Framework 4, or if the 3.0 or 3.5 version of the WCF Http Activation module is installed after installing IIS and .NET Framework 4.
Problem	Could not load type `System.ServiceModel.Activation.HttpModule` from assembly `System.ServiceModel, Version=3.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089`.

Resolution	Use the ASP.NET IIS Registration Tool (`Aspnet_regiis.exe`) to register the correct version of ASP.NET: Run the following from command line: aspnet_regiis.exe /iru The aspnet_regiis.exe file can be found in either: • `%windir%\Microsoft.NET\Framework\v4.0.30319` • `%windir%\Microsoft.NET\Framework64\v4.0.30319` (on a 64-bit machine)
Comments	Refer to the following links for additional information: • http://devonenote.com/2010/06/could-not-load-type-system-servicemodel-activation-httpmodule/ • http://social.msdn.microsoft.com/Forums/eu/wcf/thread/39571e42-aca7-469d-8c68-aa59c2da4fcc

Summary

Windows 2008 (32 & 64 bit)

The Banner Financial Aid FM Need Analysis Service is fully compatible with Windows Server 2008 (32 & 64 bit) operating system version on the application server.

Windows 2008 R2 (64 bit)

The Banner Financial Aid FM Need Analysis Service is fully compatible with Windows Server 2008 R2 (64-bit) operating system version on the application server.

Additional Supporting Information

Download Location for .Net Framework 3.5 Service Pack 1

This page contains the link to download the installation package of .Net Framework version 3.5 Service Pack 1 which is a full cumulative update with new features build incrementally upon the .Net Framework 2.0, the .Net Framework 3.0, and the .Net Framework 3.5 and also contains issue fixes. This service pack is the latest version recommended for all operating systems.

http://www.microsoft.com/downloads/en/details.aspx?FamilyID=ab99342f-5d1a-413d-8319-81da479ab0d7&displaylang=en

Download Location for .NET Framework 4.0

This page contains the link to download the installation package of .Net Framework version 4.0 which introduces new features and improvements and is highly compatible with applications that are built with earlier .Net Framework versions, except for some changes that were made to improve security, standards compliance, correctness, reliability, and performance.

http://www.microsoft.com/downloads/en/details.aspx?FamilyID=9cfb2d51-5ff4-4491-b0e5-b386f32c0992&displaylang=en

Windows Server 2008 Product Overview Guide

This page contains the Product Overview documentation for Windows Server 2008.

http://www.greymatter.com/StaticPages/ProductGuideWindowsServer2008/tabid/212/Default.aspx

Windows Server 2008 R2 Product Overview Guide

This page contains the Product Overview documentation for Windows Server 2008 R2.

http://www.microsoft.com/windowsserver2008/en/us/overview.aspx

.Net Framework Support in Windows Server 2008

This page contains the available support provided by Windows Server 2008 for the various .Net Framework versions.

http://msdn.microsoft.com/en-us/library/cc531167.aspx

Microsoft .NET Framework 4 Readme

This page contains a list of known issues with .Net Framework 4.0 version.

http://download.microsoft.com/download/B/5/7/B57D25A2-B3FD-4668-91B9-DB43B6BD910D/NETFx4RTM.htm

ASP.NET IIS Registration Tool (Aspnet_regiis.exe)

This page contains the information about the ASP.NET IIS Registration tool (`Aspnet_regiis.exe`) which is used to register ASP.NET applications with Internet Information Services (IIS).

http://msdn.microsoft.com/en-us/library/k6h9cz8h.aspx

Troubleshooting

Proxy not authorized

Related Error Message

java.sql.SQLException
ORA-28150: proxy not authorized to connect as client

Description

Symptom:	The user login into INB is not setup to allow proxy login.
Solution:	Review: Required installation (step 1).

Unauthorized

Related Error Message

org.apache.axis2.AxisFault: Transport error: 401 Error: Unauthorized

Description

Symptom:	The `nas.properties` file embedded within the `finaidfmlogin.jar` must be updated with the correct user name and password to allow a successful login to the server.
Solution:	Review: Required installation (step 3).

AQ Times out

As a default, the advanced queues for any Financial Aid process using AQ will timeout after four days of inactivity. A longer interval may be set using the GTVSDAX IDLEWAIT parameter. To change the IDLEWAIT setting from the default External Code value of *0* (hardcoded to mean 4 days), do the following:

1. Stop RNRQINI as follows:

   From host, run the following command:

   ```
   rnrqini user/password STOP
   ```

2. Login into INB and open the Crosswalk Validation (GTVSDAX) form. Enter a query on:

- **Code** - *IDLEWAIT*

- **Group** - *R%*

3. Change the **External Code** value from the default of *0* to be the number of seconds desired before timeout.

   Note: The **External Code** value is in seconds. So, 60 sec * 60 min * 24 hours * 4 days = 345600 units.

   If a *0* is used the logic in `rokadvq.f_get_idlewait` will return the default value of 345600 or 4 days. Therefore, for RNPFMXX and RPRSSBP, *0* = 4 days.

If you do not want the AQ process to timeout after 4 days of inactivity, please change this value from **External Code** = *0* to the desired number of seconds, for example 30 days: 60 sec * 60 min * 24 Hours * 30 days = 2592000 **External Code** = *2592000*

Please note it is recommended you set the IDLEWAIT value to less than 50 days (4,320,000 seconds)

4. Restart AQ as follows:

   From host, run the following command:

   ```
   rnrqini user/password START
   ```

FAQs

FM Need Analysis FAQs

Q1: What if the job submission server must use a proxy server to connect to the internet?
A: Refer to the *Proxy Settings* procedure (Proxy settings for process `rnpfm12.jar`).

Q2: Where do I get the seed number to update the seed.properties file within the `finaiddbseed.jar`?
A: If you have enough permission you can use the following SQL statement to retrieve the seed numbers:

```
SELECT GUBIPRF_SEED1,GUBIPRF_SEED3 FROM bansecr.gubiprf;
GUBIPRF_SEED1 GUBIPRF_SEED3
------------- -------------
12345678      87651234
```

Q3: How do I test my login credential to the server for validity?
A: You can test the credential from a web browser.

1. From a web browser, go to the following URL:

   ```
   https://fm1.ellucian.com/EFCService/
   EFCCalculationService.asmx
   ```

2. In the login popup screen, type in the institution login user name and password.
3. If login is successful an EFCCalculationService web page is displayed.

Q4: There are two records in RORPARM called `WS_ENDDPOINT`. How is this additional record used?

RORPARM_GROUP_NAME	RORPARM_ACTIVE_IND	RORPARM_PARAMETER	RORPARM_DATA	RORPARM_DESCRIPTION
FMCAL	Y	ONLINE_REPORT	N	Default is N. Set to Y if you want to generate the Need Analysis Report when calculating need online.
FMCAL	N	WS_ENDPOINT	https://fm1.sungardhe.com/EFCService_1/EFCCalculationService.asmx	FM Calculation Web Service URL (+1 Test site).

A: This record with "`/EFCService_1`" in the URL can be made active instead of the production URL record for pre-production testing.

Q5: Do I need to open port 443 on the Oracle database server to allow connection to the server?
A: No. For security reasons, Ellucian has intentionally not coded the Need Analysis to make a web service call directly from the Oracle database server and the INB server. However, the job submission server will need port 443 open to allow the batch process to connect to the server.

Q6: If the Off-Premise option has been chosen, do we still need to change anything in RORPARM?
A: No. The default configuration is for Off-Premise.

Q7: We are an Off-Premise client and have just upgraded from 8.8.1 to 8.10. The servlet was working fine for us prior to the upgrade. With the recent upgrade to release 8.10, we now get the following error when doing a calculation from ROAIMMP:

Note: Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

ERROR. `org.apache.axis2.AxisFault` - Connection timed out: `connect`
`org.apache.axis2.AxisFault.makeFault(AxisFault.java:430)`

And this message when testing from a web browser:

Web Service connection Failed. ERROR. Invalid login credential or certificate.

ERROR. `org.apache.axis2.AxisFault` - Connection timed out: `connect`
`org.apache.axis2.AxisFault.makeFault(AxisFault.java:430)`

A: With the 8.10 release, a second URL was added for immediate Need Analysis processing. Open the firewall and/or VPN to allow connections to both URLs, using port 443:

```
https://fm1.ellucian.com

https://fm2.ellucian.com
```

Q8: For RORPARM settings, what is the difference between `WS_ENDPOINT` and `WS_ENDPOINT_ONLINE`?
A: `WS_ENDPOINT` is used by processes `rnpfm11/12.jar` whereas, `WS_ENDPOINT_ONLINE` is used by the servlet `rnpfmxx.war`.

Note: Even though the servlet remains available for aid year calculations prior to 2014-2015, its use is highly impractical and not recommended. Servlet use would require that you make RORPARM parameter changes (back and forth) depending upon the aid year being calculated.

Note: With the de-support of the servlet with the 8.19 release, `WS_ENDPOINT_ONLINE` is no longer used.

Q9: For RORPARM settings, why does Ellucian have three sites?
A: For best performance when doing a single calculation from immediate processing or ONLINE processing, Advanced Queuing is the best option.

To help with performance, a set of calculators have been separated from the batch processing. This way, the online processing is not competing for the same resources as the batch processing of thousands of student records.

- Batch processing (production version of the calculator):

```
https://fm1.ellucian.com/EFCService/
EFCCalculationService.asmx
```

- ONLINE processing (production version of the calculator):

```
https://fm2.ellucian.com/EFCService_IMMP/
EFCCalculationService.asmx
```

- Batch and online processing (BETA/pre-production version of the calculator):

```
https://fm1.ellucian.com/EFCService_1/
EFCCalculationService.asmx
```

Appendix A - FM Need Analysis Schema

The following code represents the current FM Need Analysis Schema:

```
<?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:message
s" xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:sungardhe:enterprise:financial_aid:mess
ages" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <!-- 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 name="TransactionReceiptDate"
type="finaid:DateBlankType" minOccurs="0"/>
              <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="CalcPCforIndependentStudent"
type="finaid:YesNoType" minOccurs="0"/>

      <xs:element name="StudentAlienRegistrationNoStatus"
type="finaid:YesNoWithBlankType" minOccurs="0"/>

      <xs:element name="SSACitizenshipStatusCode"
type="finaid:SSACitStatCodeWithBlankType"  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:element name="ISIRCommentCodes"
type="finaid:ISIRComments" 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:element name="ApplicationReceiptDate"
type="finaid:DateBlankType" 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="StudentTaxFilingCode"
type="finaid:TaxFilingCodeType" minOccurs="0"/>
      <xs:element name="ParentTaxFilingCode"
type="finaid:TaxFilingCodeType" minOccurs="0"/>
      <xs:element name="Form1040AEZEligibleCode"
type="finaid:Form1040AEZEligibleCodeType" minOccurs="0"/>
      <xs:element name="AdjustedGrossIncome"
type="finaid:SignedInteger7DigitsWithBlankType"
minOccurs="0"/>
      <xs:element name="IncomeTaxPaid"
type="finaid:Integer7DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ExemptionsClaimed"
type="finaid:Integer2DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="StudentEarnedIncome"
type="finaid:SignedInteger7DigitsWithBlankType"
minOccurs="0"/>
      <xs:element name="SpouseEarnedIncome"
type="finaid:SignedInteger7DigitsWithBlankType"
minOccurs="0"/>
      <xs:element name="FathersEarnedIncome"
type="finaid:SignedInteger7DigitsWithBlankType"
minOccurs="0"/>
      <xs:element name="MothersEarnedIncome"
type="finaid:SignedInteger7DigitsWithBlankType"
minOccurs="0"/>
```

```
      <xs:element name="EducationCredits"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ChildSupportPaid"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="NeedBasedEmployment"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="GrantScholarshipAid"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="CombatPay"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="CoopEarnings"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="PensionPayments"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="IRAPayments"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="ChildSupportReceived"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="InterestIncome"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="IRADistributions"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="UntaxedPensions"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="MilitaryClergyAllowances"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="VeteranNonEducationBenefits"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="OtherUntaxedIncome"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="OtherNonReportedMoneyReceived"
type="finaid:Integer8DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="AdjustedGrossIncomeFlag"
type="finaid:Integer1DigitWithBlankType" minOccurs="0"/>
      <xs:element name="IncomeTaxFlag"
type="finaid:Integer1DigitWithBlankType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AssetDataType">
    <xs:sequence>
      <xs:element name="CashSavingsChecking"
type="finaid:Integer7DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="InvestmentNetWorth"
type="finaid:Integer7DigitsWithBlankType" minOccurs="0"/>
      <xs:element name="BusinessFarmNetWorth"
type="finaid:Integer7DigitsWithBlankType" minOccurs="0"/>
```

```
      <xs:element name="AssetThresholdExceeded"
type="finaid:YesNoWithBlankType" 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:element name="NSLDSLoanFraudFlag"
type="finaid:YesNoType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ISIRComments">
    <xs:sequence>
      <xs:element name="ISIRCommentCode" type="xs:string"
minOccurs="0" maxOccurs="unbounded"/>
    </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:element name="Version" type="xs:string"
minOccurs="0"/>
                 <xs:element name="ProtectionAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>
                 <xs:element name="Age" type="xs:string"
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:element name="v4EFCCalculations"
type="finaid:v4EFCIntermediateValuesType" minOccurs="0"/>
         </xs:sequence>
       </xs:complexType>
     </xs:element>
     <xs:complexType name="v4EFCIntermediateValuesType">
       <xs:sequence>
         <xs:element name="Student"
type="finaid:StudentValuesType" minOccurs="0"/>
         <xs:element name="Parent"
type="finaid:ParentValuesType" minOccurs="0"/>
         <xs:element name="AlternativeEfcMonth1"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
         <xs:element name="AlternativeEfcMonth2"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
         <xs:element name="AlternativeEfcMonth3"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
         <xs:element name="AlternativeEfcMonth4"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
```

```
      <xs:element name="AlternativeEfcMonth5"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth6"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth7"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth8"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth9"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth10"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth11"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="AlternativeEfcMonth12"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="EFCIntermediateValuesType">
    <xs:sequence>
      <xs:element name="TotalIncome"
type="finaid:SignedInteger8DigitsType" 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:SignedInteger8DigitsType" 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:SignedInteger8DigitsType" 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:SignedInteger8DigitsType" 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:SignedInteger9DigitsType" minOccurs="0"/>

      <xs:element name="StudentAssetContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="FISAPTotalIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

      <xs:element name="NineMonthEFC"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

    </xs:sequence>

  </xs:complexType>

  <xs:complexType name="ParentValuesType">

    <xs:sequence>

      <xs:element name="TotalIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

      <xs:element name="StateTaxAllowance"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="IncomeProtectionAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="EmploymentAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="TotalIncomeAllowances"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="AvailableIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

      <xs:element name="NetWorth"
type="finaid:Integer9DigitsType" minOccurs="0"/>
```

```
      <xs:element name="AssetProtectionAllowance"
type="finaid:Integer9DigitsType" minOccurs="0"/>

      <xs:element name="DiscretionaryNetWorth"
type="finaid:SignedInteger9DigitsType" minOccurs="0"/>

      <xs:element name="ParentAssetContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="ParentContributionFromAssets"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="ParentContributionFromIncome"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="AdjustedAvailableIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

      <xs:element name="TotalParentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="ParentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="ParentSocialSecurityTax"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="NegativeAdjustableIncomeOffset"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="AlternativeTotalParentContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      </xs:sequence>

  </xs:complexType>

  <xs:complexType name="StudentValuesType">

    <xs:sequence>

      <xs:element name="TotalIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

      <xs:element name="StateTaxAllowance"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="IncomeProtectionAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="EmploymentAllowance"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="TotalIncomeAllowances"
type="finaid:Integer7DigitsType" minOccurs="0"/>

      <xs:element name="AvailableIncomeContribution"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>

       <xs:element name="AssetProtectionAllowance"
type="finaid:Integer9DigitsType" minOccurs="0"/>

      <xs:element name="DiscretionaryNetWorth"
type="finaid:SignedInteger9DigitsType" minOccurs="0"/>

      <xs:element name="StudentAssetContribution"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>

      <xs:element name="TotalStudentContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
```

```
      <xs:element name="FISAPTotalIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>
      <xs:element name="AvailableIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>
      <xs:element name="NetWorth"
type="finaid:Integer9DigitsType" minOccurs="0"/>
      <xs:element name="StudentSocialSecurityTax"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentTotalIncome"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>
      <xs:element name="StudentStateTaxAllowance"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentDiscretionaryNetWorth"
type="finaid:SignedInteger8DigitsType" minOccurs="0"/>
      <xs:element name="StudentIncomeAllowances"
type="finaid:SignedInteger7DigitsType" minOccurs="0"/>
      <xs:element name="StudentIncomeContribution"
type="finaid:Integer7DigitsType" minOccurs="0"/>
      <xs:element name="AdjustedAvailableIncome"
type="finaid:SignedInteger8DigitsType" 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="YesNoWithBlankType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Y"/>
    <xs:enumeration value="N"/>
    <xs:enumeration value=""/>
  </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: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="5"/>
    <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="TaxFilingCodeType">
  <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=""/>
  </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:enumeration value="Processed"/>
  </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="21"/>
      <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="0"/>
    </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="TaxFilingCodeType">
    <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=""/>
    </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="Integer7DigitsWithBlankType">
    <xs:union memberTypes="finaid:Integer7DigitsType
finaid:BlankType"/>
  </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="SignedInteger7DigitsWithBlankType">
     <xs:union memberTypes="finaid:SignedInteger7DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="Integer8DigitsType">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="99999999"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Integer8DigitsWithBlankType">
    <xs:union memberTypes="finaid:Integer8DigitsType
finaid:BlankType"/>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger8DigitsType">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="-99999999"/>
      <xs:maxInclusive value="99999999"/>
      <xs:fractionDigits value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="SignedInteger8DigitsWithBlankType">
```

```
    <xs:union memberTypes="finaid:SignedInteger8DigitsType
finaid:BlankType"/>
  </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="21"/>
      <xs:enumeration value="24"/>
      <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:simpleType name="SSACitStatCode">
      <xs:restriction base="xs:string">
        <xs:minLength value="0"/>
        <xs:maxLength value="1"/>
      </xs:restriction>
```

```
   </xs:simpleType>
  <xs:simpleType name="SSACitStatCodeWithBlankType">
    <xs:union memberTypes="finaid:SSACitStatCode
finaid:BlankType "/>
  </xs:simpleType>
</xs:schema>
```

Appendix B - Enabling Secure Sockets Layer (SSL) Communication

The Oracle *Application Server Administrator's Guide* provides recommendations and instructions for enabling secure communications throughout the Oracle Application Server (OAS) components. A typical application server deployment may consist of:

1. Network Load Balancer hardware which is used to route HTTP traffic to one or more Oracle Application Server installations.
2. The Oracle HTTP Server component which is used to serve static HTML resources and dynamic web applications using the `mod_plsql` module which is used by the Banner Self-Service web applications.
3. Java Web containers which support deployment of Ellucian Java web applications. Both the Oracle Containers for Java (OC4J) and Oracle Weblogic are web application servers supported by Oracle Application Server.

An institution may choose to enable Secure Sockets Layer (SSL) communication in one or more of the components listed above.

Please review the Oracle *Application Server Administrator's Guide*, *Part IV*, *Secure Sockets Layer* on how to enable SSL in one or more Oracle Application Server components.

Recommendations for securing applications in Oracle Application Server

Recommendation #1:

Enable SSL at the Network Load Balancer (NLB) hardware to support SSL communications between the web browser and the NLB. SSL is terminated at the NLB hardware and HTTP communication is continued to one or more Oracle Application Server instances. This configuration can securely support a network configuration where a Network Firewall resides between the NLB and OAS servers.

Recommendation #2:

Enable SSL in the Oracle HTTP Server component. This allows SSL to be configured once, at this component, and allows Oracle HTTP Server to route requests to one or more Weblogic or OC4J web containers. This configuration supports a secure environment for HTTPS communication between a web browser and the Oracle HTTP Server component.

SSL is terminated at the Oracle HTTP Server component and requests are either serviced by this component or routed to one of more Java web containers. Because the Oracle HTTP Server and Java web container are deployed on the same machine, SSL communication is not required to be enabled in the Java Web container.

Institutions may follow either recommendation or use a combination of both.

Note: It is not recommend that you enable SSL in the Weblogic or OC4J embedded HTTP servers because this requires additional maintenance to configure SSL and allows access to new HTTP ports. This configuration typically requires additional firewall rule exceptions to allow access. Ellucian Java web applications may be deployed in multiple Weblogic or OC4J servers and each would carry the burden of configuring SSL for each installation.

For additional information, please point your browser to the Ellucian Support Center page (http://www.ellucian.com/Solutions/Ellucian-Client-Support/) to review the following Article:

- 1-DTOVYK - How to configure SSL for INB & SSB under Oracle Fusion Middleware 11g

Recommendations for enabling SSL in a test environment

Ellucian recommends enabling SSL in the Oracle HTTP Server component to support a testing environment. For production environments, Oracle and Ellucian recommend customers obtain a Digital Certificate issued by a well known and reputable certification authority. This allows trusted communication via HTTPS by many programs that know and trust the chosen authority.

For test environments, customers may choose to create their own self-signed certificate which is used only for testing purposes. Web browsers and other programs will not trust this self-signed certificate and generate the appropriate warning messages, indicating that the certificate is not signed by a well known and reputable certification authority.

For additional information, please point your browser to the Ellucian Support Center page (http://www.ellucian.com/Solutions/Ellucian-Client-Support/) to review the following Article:

1-B8E7LJ - How To Generate A Wallet Containing A Self Signed Certificate Using ORAPKI

Warning! *This self-signed certificate should be used for testing purposes only with Oracle Application Server*

Configuring the Oracle Application Server to trust a customer generated self-signed certificate

When a customer creates a self-signed certificate and enables SSL in the Oracle HTTP Server component, web browsers will display a warning message that this website is not trusted. This is because the certificate was not signed by a well know and reputable certification authority.

Ellucian Java applications deployed in OAS can be configured to communicate to other Ellucian applications via HTTPS. The Java environment that is installed with OAS will generate the following error (as the certificate is not trusted):

javax.net.ssl.SSLException: SSL handshake failed

The institution can modify the OAS test Java environment by importing the self-signed certificate as a trusted certificate into the Java trusted certificate authority key store file.

This is located in:

```
$ORACLE_HOME/jdk/jre/lib/security/cacerts
```

The following command is used to import a self-signed certificate which will be trusted by the Java environment and allow HTTPS communication:

```
$keytool –import –trustcacerts –alias “pickAnAliasName” –file
“pathOfSelfSignedCert” –keystore $ORACLE_HOME/jdk/jre/lib/
security/cacerts
```

This will allow Ellucian Java applications, such as Cascade, to communicate to other Ellucian web applications via HTTPS and not encounter this error because the Java environment will trust this certificate to support HTTPS communication.